

UNIVERSIDAD DE GRANADA

Departamento de Lenguajes y Sistemas
de información



Programa Oficial de Doctorado en
Tecnologías de la Información y la Comunicación

Tesis Doctoral

Control y supervisión avanzada de sistemas basados en el uso de Interfaces Naturales de Usuario de Tipo Gestual (INU-G)

Realizada por:

Washington Gonzalo Pomboza Junez

Dirigida por:

Juan Antonio Holgado Terriza

Editor: Universidad de Granada. Tesis Doctorales

Autor: Washington Gonzalo Pomboza Junez

ISBN: 978-84-9163-538-3

URI: <http://hdl.handle.net/10481/48329>

El doctorando D. Washington Gonzalo Pomboza Junez y su director de la tesis D. Juan Antonio Holgado Terriza, garantiza, al firmar esta tesis doctoral, que el trabajo ha sido realizado por el doctorando bajo su dirección y hasta donde nuestro conocimiento alcanza, en la realización del trabajo, se han respetado los derechos de otros autores a ser citados, cuando se han utilizado sus resultados o publicaciones.

Granada, Septiembre del 2017

Doctorando

Washington Gonzalo Pomboza Junez

Director de la Tesis

Fdo: Juan Antonio Holgado Terriza

A mi familia.

Agradecimientos

Mientras la inquietud del hombre permanezca, las montañas serán valles y el infinito un pequeño salto... soñar implica añorar lo que se tiene, desear lo que la fe materializa y mantener en el corazón la nobleza del espíritu.

Gonzalo

No hay plazo que no se cumpla... dice un refrán, pues luego de haber recorrido el camino de la ciencia, donde familiares, amigos y compañeros han apostado por el proyecto iniciado, debo decir que han sido el gran apoyo para lograr el objetivo planteado.

Al llegar a culminar mi tesis, me conmueve el sentimiento de gratitud hacia todas las personas e instituciones que incondicionalmente me han brindado su apoyo para alcanzar este sueño. Por ello, en estas líneas expreso mis más sinceros agradecimientos

En primer lugar, agradezco a mi Director de tesis y Tutor Juan Antonio, por darme la oportunidad de trabajar a su lado, por guiarme en todo el camino del doctorado, por todas esas revisiones y correcciones tan acertadas a mis tan difusos manuscritos. Gracias por todo, pero principalmente, gracias por brindarme su amistad. Espero tener la suerte de seguir trabajando con él en nuevos proyectos. De igual manera, dejo constancia de agradecimiento a Nuria, por su apoyo incondicional y poner todo el esfuerzo para llegar al objetivo anhelado, por su calidad profesional y personal. Gracias por abrir la puerta del conocimiento.

Muchas gracias a Pablo, Jesús y todos los compañeros del grupo de trabajo, por su amistad y apoyo que me han dado durante todo este tiempo, y por haberme acogido como un miembro más de trabajo diario y constante. Mil gracias a todos los miembros de este gran grupo. El tiempo que he convivido con ustedes ha sido verdaderamente gratificante.

Agradezco a todos mis amigos y amigas que he conocido durante todo el trayecto del doctorado, y que al igual que yo, al estar lejos de su tierra y de los suyos, nos une el sentimiento de nostalgia que muchas veces nos invade, pero que es minimizado por la relación de amistad y por aquellos momentos de tapas que pasamos juntos. Muchas gracias a Andrés, Virgilio, Elio, Jorge....

Agradezco también a la Secretaria de Educación Superior, Ciencia, Tecnología e Innovación (SENESCYT) y al gobierno de la República del Ecuador por la beca que se me otorgó para poder realizar estos estudios.

Muchas gracias a mis amigos y compañeros de la Universidad Nacional de Chimborazo y a sus dignas autoridades, especialmente a su Rector, Ph. D. Nicolay Samaniego, por haberme brindado su apoyo para realizar estos estudios y a todos quienes en mayor o menor grado me han ofrecido su ayuda, expresamente a aquellos que permitieron lograr la Comisión de Servicios.

Gracias a la Universidad de Granada y al Centro de Investigaciones y Tecnologías Informáticas (CITIC) por haberme permitido formar parte de ellos.

Agradezco profundamente a toda mi familia por todas las atenciones, cariño y amor que siempre han sabido darme, lo cual ha mantenido firme mi decisión y me ha dado fuerzas para seguir adelante. Mil gracias a mis padres+ por sus consejos y apoyo incondicional que siempre me ofrecieron. Gracias por enseñarme a desafiar al mundo con sacrificio y empeño, para lograr un sueño. Gracias a mis hermanos, Bethy y Edwin por todo su afecto y amor que me han dado. Este trabajo también es por ustedes.

Y, he dejado para el final, el poder expresar mi profundo agradecimiento y gran amor que les tengo a mis cuatro soles. Muchas gracias a ti Lili quien, durante mucho tiempo, ha sido mi compañera de luchas y batallas las cuales hemos emprendido y librado juntos, que siempre me ha demostrado su amor, apoyo y dedicación hacia nuestros tres amores que iluminan y dan sentido a nuestras vidas y a los que amamos tanto, nuestros tres tres caballeros: RONNY, DAVID Y DANIEL, fuentes inagotables de inspiración y que han motivado en gran manera a concluir esta tesis. Espero que Dios, me permita compensar de alguna forma mi ausencia de todos ellos en estos años. Los amo!

MIL GRACIAS A TOD@S!

Índice general

Introducción	3
Antecedentes y motivación	3
Objetivos	7
Estructura de la tesis	7
1. El gesto y su interfaz	11
1.1. El gesto y sus características	12
1.1.1. Representación	13
1.1.2. Significado	13
1.1.3. Dificultad	14
1.2. Taxonomía	15
1.2.1. En la comunicación	15
1.2.2. Por su naturaleza	17
1.2.3. Gesto estático y dinámico	18
1.2.4. El gesto en interacción	20
1.3. Captura y reconocimiento gestual	20
1.3.1. Reconocimiento basado en dispositivos ópticos	23
1.3.2. Reconocimiento basado en sensores	26
1.3.3. Reconocimiento basado en una superficie de contacto	30
1.4. Procesamiento de la información	32
1.4.1. Algoritmos de aprendizaje supervisado	33
1.4.2. Algoritmos de aprendizaje no supervisado	33
1.4.3. Algoritmos de aprendizaje semi-supervisado	34
1.4.4. Conceptos de Aprendizaje de máquina (Machine Learning)	34
1.4.5. Flujo de la información	38
1.5. Resumen	42

2. Análisis de la Interfaz gestual usando medios ópticos (tracking)	45
2.1. Fundamentos de detección y seguimiento	46
2.1.1. Selección de características por tracking	49
2.1.2. Revisión científica	49
2.1.3. Enfoques para clasificación de gestos	56
2.2. El dispositivo Leap Motion	56
2.2.1. Gestos predefinidos	59
2.2.2. Gestos implementados	60
2.2.3. Características gestuales	62
2.3. Análisis del proceso de reconocimiento	65
2.3.1. Participantes	66
2.3.2. Biblioteca gestual propuesta	66
2.3.3. Protocolo experimental	66
2.3.4. Frecuencia de muestreo	71
2.3.5. Características: manos y componentes	72
2.3.6. Métricas	74
2.4. Resultados y discusión	77
2.4.1. Proceso de reconocimiento	77
2.4.2. Efectividad	80
2.4.3. Gesto de Control Efectivo	81
2.5. Propuesta experimental	83
2.5.1. Robot	83
2.5.2. Arquitectura del sistema	83
2.5.3. Gestos de Control	85
2.5.4. Interfaz gestual implementada en LEAPEV3	85
2.5.5. Experimentos con el prototipo	87
2.6. Conclusiones	88
3. Análisis de la Interfaz gestual usando sensores (señales EMG)	91
3.1. Las bioseñales	92
3.1.1. Señales EMG	94
3.2. Detección y acondicionamiento de la señal EMG	95
3.2.1. Capturar señales EMG usando electrodos	95
3.2.2. Amplificar la señal EMG	98
3.2.3. Medir las señales EMG	99
3.2.4. Problemas al medir señales EMG	101
3.3. El dispositivo de captura gestual MYO	106

3.4. Características y reconocimiento de patrones	109
3.4.1. Segmentación de datos	110
3.4.2. Selección de características	111
3.5. Métodos de clasificación	113
3.5.1. Revisión científica	113
3.5.2. Algoritmo Support Vector Machines	120
3.5.3. Algoritmo k-Nearest Neighbor	130
3.6. Proceso experimental	132
3.6.1. Librería gestual propuesta	135
3.6.2. Extracción de características	135
3.6.3. Participantes.	136
3.6.4. Experimentos off-line	139
3.6.5. Experimentos on-line	145
3.7. Resultados y discusión	145
3.7.1. Experimentos off-line	145
3.7.2. Experimentos on-line	148
3.8. Conclusiones	152
4. Análisis de la Interfaz gestual vs. la pantalla táctil	155
4.1. El gesto y su ámbito	157
4.1.1. El gesto en la interacción	157
4.1.2. El reconocimiento gestual	158
4.1.3. La aplicación del gesto	159
4.1.4. La accesibilidad gestual	160
4.1.5. Revisión científica	160
4.2. TUI, evolución y características	161
4.2.1. Ley de Fitts	164
4.2.2. Características de TUI	165
4.2.3. Android en los Smartphone	166
4.3. Características del Smartphone y su interfaz	168
4.3.1. Zonas de importancia	169
4.4. Definición del Test de interacción	172
4.4.1. Definición de los parámetros de evaluación	173
4.4.2. Software para test	173
4.5. Probando TUI vs. GBUI	176
4.5.1. Participantes	176
4.5.2. Protocolo experimental	176

4.5.3.	Pruebas con TUI	177
4.5.4.	Pruebas con GBUI	178
4.6.	Resultados	180
4.6.1.	TUI	180
4.6.2.	GBUI	182
4.6.3.	Comparación TUI vs. GBUI	184
4.7.	Propuesta experimental	185
4.7.1.	Arquitectura de interfaz GBUI	185
4.7.2.	Librería gestual y comandos	187
4.7.3.	Diseño de SICLLE	188
4.8.	Conclusiones	191
5.	El gesto como lenguaje de control	193
5.1.	Métricas gestuales	196
5.1.1.	Confort	196
5.1.2.	Exactitud del comando gestual	197
5.2.	Estudio de confortabilidad	198
5.3.	Fundamentos del Árbol de Control Gestual (GCT)	198
5.3.1.	Análisis y definiciones	199
5.3.2.	Estructura del GCT	201
5.3.3.	Sintaxis de la Actividad	204
5.4.	Materiales y métodos	205
5.4.1.	Dispositivo de captura gestual	205
5.4.2.	Biblioteca gestual	205
5.4.3.	Participantes	205
5.4.4.	Protocolo experimental	207
5.5.	Resultados	212
5.5.1.	Validando el GCT	217
5.6.	Conclusiones	218
6.	Conclusiones y trabajos futuros	221
Conclusiones	221
Publicaciones derivadas de la investigación	224
Trabajos futuros	226
A.	Señales EMG	229
B.	Análisis y conceptos complementarios	235

Índice de figuras

1.1. Taxonomía del gesto desde el punto de vista de la comunicación [147]	16
1.2. Taxonomía basada en el tipo de gesto	19
1.3. Formas de captura del gesto	23
1.4. Técnicas de reconocimiento del gesto basadas en visión	24
1.5. Usuario interactuando con un Smartphone.	31
1.6. Elementos de un sistema de aprendizaje [134].	36
1.7. Proceso usado en algoritmos supervisados de ML	39
1.8. Matriz de confusión de ML	41
2.1. Reconocimiento de un objeto por su forma y apariencia usando tracking en [199].	48
2.2. Leap Motion	57
2.3. Leap Motion y sus componentes	58
2.4. Gestos predefinidos de Leap Motion	59
2.5. Gestos implementados en esta investigación	61
2.6. Leap Motion y su área de interacción	62
2.7. Componentes para identificación del gesto.	64
2.8. Tasa de muestreo con y sin carga gestual	65
2.9. Enfoque dado al análisis del dispositivo.	66
2.10. Biblioteca gestual propuesta.	67
2.11. Condiciones experimentales	68
2.12. Escenarios de entrenamiento y prueba para la captura de datos.	69
2.13. Gestos y punto de referencia	70
2.14. Variación de frames para reconocimiento del gesto (a) y (e).	71
2.15. Datos de reconocimiento entregados por la API de LMC.	72
2.16. Librería gestual completa	78
2.17. Proceso de reconocimiento, estimado en frames y tiempo	79

2.18. Tasa media de muestreo vs tasa media de reconocimiento de LMC	80
2.19. Efectividad de los gestos propuestos en la biblioteca gestual de prueba	82
2.20. Sistema robótico EV3 GRIPP3R y sus componentes.	83
2.21. Arquitectura propuesta para la implementación del sistema de control gestual para dispositivos robóticos.	84
2.22. Descripción de LEAPEV3 y su operación gestual.	86
2.23. Sistema LEAPEV3 en plena operación (modo automático gestual).	87
3.1. Diagrama de una unidad motora [124].	94
3.2. Raw de la señal EMG. (a) Señal EMG medida en el antebrazo usando sEMG. (b) Señal rectificada.	97
3.3. Amplificador operacional de propósito general típico.	99
3.4. Raw de la señal EMG	100
3.5. Características y tipos de electrodos sEMG.	105
3.6. Pulsera gestual MYO y componentes externos. (a) Distribución de sensores. (b) Dimensiones de los sensores sEMG.	106
3.7. Sensor sEMG de MYO desde varias perspectivas.	107
3.8. Hiperplano óptimo en SVM para un caso separable linealmente.	121
3.9. Dos casos de separación en SVM. (a) Función de decisión lineal con errores. (b) Función de decisión no lineal.	123
3.10. Clasificación con SVM, casos <i>separables</i> , <i>no separables</i> y <i>mal clasificados</i>	124
3.11. Transformación a un espacio de alta dimensionalidad.	126
3.12. Diferentes tipos de valores de k	131
3.13. Flujo de la información para reconocimiento del gesto.	133
3.14. Gestos que conforman la librería de gestos para control.	135
3.15. Señales EMG al formar el gesto (i) y sus características.	137
3.16. Ubicación adecuada de MYO en el antebrazo del usuario.	139
3.17. Señales EMG, en los ocho canales, obtenidas al formar el gesto (i).	140
3.18. Lecturas recibidas por los ocho sensores sEMG.	142
3.19. Filtrado y selección de características aplicada a la señal EMG.	143
3.20. Matrices de confusión de los clasificadores usados.	146
3.21. Arquitectura para sistemas de reconocimiento gestual basada en clasificadores.	149
3.22. Sistema de reconocimiento en operación. (a) Ilustración de los músculos del antebrazo. (b) Pulsera MYO en brazo derecho de usuario.	151

4.1. Diagrama evolutivo de Android.	168
4.2. <i>Zonas de importancia</i> en un Smartphone con SO Android.	170
4.3. Secuencia de interacción a través de las <i>Zonas de Importancia</i>	172
4.4. Distribución de objetivos (iconos) para el test de interacción TUI.	174
4.5. Secuencia de gestos-objetivos por cada escenario en interacción GBUI usando GCT.	176
4.6. Escenarios de la interacción TUI (obtenida desde el test de interacción de la Sección 4.4).	177
4.7. Escenarios y niveles del test de interacción por GBUI (obtenida desde el test de interacción de la Sección 4.4).	179
4.8. Intentos positivos y negativos usando TUI.	181
4.9. Niveles de <i>ACC</i> alcanzados usando TUI.	182
4.10. Intentos positivos y negativos usando GBUI.	183
4.11. Niveles de <i>ACC</i> alcanzados usando GBUI.	184
4.12. Comparación de niveles de <i>ACC</i> alcanzados por las interfaces TUI y GBUI.	184
4.13. Arquitectura usada en el sistema SICLLE.	186
4.14. El Árbol de Control Gestual (GCT) definido en el sistema SICLLE.	189
4.15. Capturas de pantalla de la aplicación SICLLE.	190
5.1. Aspectos principales del gesto usado como un lenguaje.	193
5.2. Gestos propuestos para estudio de Confortabilidad y control.	198
5.3. Componentes de un grafo.	199
5.4. Estructura jerárquica basada en tres niveles.	201
5.5. Estructura jerárquica basada en tres niveles.	202
5.6. La <i>Activity</i> y componentes por nivel.	204
5.7. Biblioteca gestual propuesta para análisis de confortabilidad.	205
5.8. GCT implementado en el caso de estudio.	208
5.9. Imágenes del prototipo en operación.	209
5.10. Flujo del comando complejo para formación de una <i>Activity</i> usando el GCT.	210
5.11. Nivel de confort individual del gesto.	213
5.12. Nivel de confort de la biblioteca gestual.	214
5.13. <i>Activity</i> por usuario en posición sentado.	215
5.14. <i>Activity</i> por usuario en posición acostado.	215
5.15. Exactitud <i>ACC</i> de los usuarios en las posiciones analizadas.	216
5.16. Respuestas de los usuarios en usabilidad.	218

A.1. La totalidad del dataset de todos los gestos vistos por distribución y lineal representados.	232
A.2. Señales EMG con características de rectificación e integración.	233
B.1. Puntos de vista para análisis de MYO.	236
B.2. Variación de la exactitud usando el Raw de datos sin extracción de características.	237
B.3. Variación de la exactitud formando grupos de sensores sEMG de MYO.	239
B.4. La función de decisión es una línea recta.	241

Resumen

El uso del gesto en la interfaz es una forma de interacción que, va ganando espacio en la comunicación entre Hombre-Máquina. Esta tesis plantea el uso del gesto aplicado como una Interfaz natural de usuario en la interacción Hombre-Máquina, orientada al control de dispositivos eléctricos y electrónicos. En este trabajo se realizan varios enfoques, que tratan de descubrir la captura, reconocimiento y aplicación del gesto, sus características y su lenguaje.

Un análisis detallado del uso del gesto, especialmente de las manos, es realizado en primer término. Este análisis está encaminado a lograr una mejor comprensión de la riqueza natural que encierra el hombre en cada una de sus expresiones corporales y gestuales. Por ser tan amplia necesita ser definida y clasificada para determinar su alcance en la interacción, mediante interfaces gestuales, con dispositivos. Se definen varios aspectos, entre ellos se delimita la captura del gesto en tres ámbitos principales que son: óptico, por sensores y por contacto. Cada uno de estos aspectos es analizado e implementado, en casos de estudio experimentales. En cada uno de ellos, se contó con la participación activa de voluntarios que permitieron llegar a conclusiones válidas de aplicación.

Como consecuencia lógica de análisis inicial, el gesto, su captura y reconocimiento usando dispositivos ópticos, ha sido enfocado al control de dispositivos robóticos, en una segunda etapa. En ella se estudiaron las características de identificación y reconocimiento, llegando a establecer métricas que los definen y aplican en medios óptico-infrarrojos. Se logran definir nuevos conceptos, como el gesto de control y sus características, y además, la efectividad del gesto. Una propuesta experimental que aplica los resultados, ha sido realizada en un dispositivo robótico, implementando una interfaz natural de tipo gestual para control reactivo, usando una arquitectura abierta. Dicho experimento alcanzo los mejores resultados, y valido los conceptos de efectividad del gesto.

Luego, y continuando con nuestra investigación, hemos aplicado sensores para la captura del gesto por medio de señales electromiográficas (EMG), un tipo de

bioseñal generada en el músculo esquelético en respuesta a la actividad neuronal y que provoca la contracción y expansión de la masa muscular pertinente. Dichas señales, posibilitan el reconocimiento gestual basado en las lecturas obtenidas de aquellos músculos del brazo, antebrazo y mano que intervienen en la formación del gesto. Para establecer la validez de este tipo de reconocimiento, una amplia biblioteca gestual fue analizada e identificada usando algoritmos de Machine Learning, a saber SVM y k-NN. En el caso de SVM tres tipos de Kernel fueron probados (RBF, Polinomial y Sigmoidal). La mejor respuesta fue encontrada usando SVM con kernel Polinomial. Los resultados de este análisis fueron usados en la construcción e implementación de un sistema de interacción natural de usuario de tipo gestual basado en sensores sEMG y aplicado al control de dispositivos de casa.

Continuando con nuestra investigación, y a la luz de los resultados anteriores, se estudió y analizó un tipo de interacción ampliamente usada en nuestros días, la cual usa una pantalla táctil como medio de comunicación entre el usuario y la máquina. Definimos entonces a las interfaces por contacto con superficies táctiles (TUI). El caso más general de aplicación de una Touchscreen fue analizado, usando dispositivos Smartphone. En este sentido, se lograron resultados esenciales de evaluación, evaluando adicionalmente dos posiciones anatómicas muy usadas y posibles en el uso de una TUI sobre un Smartphone, a saber: Posición de pie o “normalz posición acostado o “especial”. Nuestros resultados indican que si bien la interacción por contacto es la más precisa, el uso del gesto es la alternativa natural más conveniente para gestión de un dispositivo, en este caso el Smartphone. Se comprobó que, la interfaz gestual es independiente de la posición anatómica del usuario y una alternativa indiscutible para casos especiales de uso como: enfermedad, impedimento físico o por comodidad. Una propuesta experimental fue desarrollada e implementada usando una arquitectura abierta, propuesta en esta investigación, y pensada para este tipo interfaz en dispositivos móviles. Los resultados fueron notable al lograr la gestión de llamadas telefónicas y el control de un Smartphone, a través del gesto del usuario y, con ayudas auditivas, evitando la supervisión visual del usuario. Finalmente, un test de usabilidad aplicado a todos los usuario, confirmó que el uso de una interfaz gestual en este tipo de dispositivos, es la opción más adecuada y quizá la ideal, en casos de emergencia, incapacidad física o visual.

Finalmente, logramos la definición de una estructura jerárquica de control, la cual usa el gesto como principal componente. Dicha estructura se fundamenta en la teoría de grafos y árboles, para llegar a definir una estructura a la cual hemos

denominado Árbol de Control Gestual (GCT). El GCT, está pensado para lograr el control de varios dispositivos y sus funcionalidades, a través de comandos complejos, formados y definidos en la Activity. Una Activity, ha sido definida como la agrupación de comandos atómicos simples, que logran su integración en base a los niveles del GCT. Para lograr un adecuado enfoque del lenguaje de control, se realizó un estudio de la confortabilidad del gesto, con la participación de voluntarios, logrando calificar al gesto en una escala de confort. Una experiencia práctica de implementación usando el concepto de GCT y de confortabilidad del gesto fue implementada, en donde se evaluó la facilidad de uso del GCT con usuarios reales y además, su aplicación al control de dispositivos de casa, tanto eléctricos y electrónicos. Los resultados indican que una estructura GCT logra muy buenos resultados tanto en facilidad de uso como en aceptabilidad de parte del usuario.

Por todo lo expuesto, esta tesis puede ser considerada un referente a la aplicación de Interfaces Naturales de Usuario de Tipo Gestual (INU-G) orientadas al control y supervisión de dispositivos eléctricos y electrónicos.

Introducción

Antecedentes y motivación

Las interfaces y su diseño han sido la principal preocupación de investigadores desde el nacimiento mismo de la informática. El afán de lograr una forma de comunicación sencilla y precisa con la máquina, al igual que sucede entre seres humanos, ha conllevado una evolución constante en la forma de comunicación con la máquina. El hombre, trata de lograr una comunicación basada en conceptos propios, concebidos y definidos en él, para adaptarlos a la interacción Humano-Máquina. Sin embargo, la forma de comunicación entre seres humanos ha sufrido también numerosas transformaciones.

Desde el lenguaje de señas, usado mucho antes que el lenguaje hablado, hasta los mensajes de correo electrónico de la actualidad, la forma de comunicación o transferencia de un concepto ha sido la clave para el desarrollo individual y colectivo. Obtener nuevos y mejores resultados, basados en la forma en la cual interactúa el hombre con la máquina, a conllevado ha experimentar numerosos cambios, todos ellos tratando de lograr un mejor aprovechamiento de los recursos disponibles (tanto humanos como técnicos), junto a una mejor comprensión de las necesidades. La ciencia del diseño y construcción de interfaces es la que ha dado soporte y ha fijado tendencias en la interacción hombre-máquina. Sus avances, son los que permitirán mejorar la interacción y superar barreras en la comunicación con la máquina.

La evolución de la interfaz, al igual que la comunicación, ha sido constante y siempre buscando la facilidad del usuario aunque ello involucre una amplia complejidad en la máquina. Desde las tarjetas perforadas de antaño, usadas como medio de entrada y salida de datos, hasta las interfaces multimodales de hoy en día (MMI, siglas en inglés), el salto tecnológico ha sido enorme. La idea de lograr una la interfaz omnipresente, siempre disponible, para un usuario cada vez más exigente, requiere de la integración de conceptos, formas, métodos y técnicas que se han ido sumando a la forma de interacción. El uso de la voz, movimientos corporales, gestos

(faciales, corporales, manuales), el toque de superficies, etc. para lograr la entrada y salida de datos es evidente en nuestros días. Así, el intercambio de información logra ser completamente dinámico y sujeto a los requisitos evolutivos del momento.

La gran ventaja alcanzada como consecuencia del avance tecnológico es lograr medir variables, usar características y componentes electrónicos insospechados hasta hace unos pocos años, logrando un intercambio de información entre el hombre y la máquina de un elevado nivel de complejidad. La razón, que el hardware y el software han mejorado y junto a ellos, nuevos conceptos que tratan de lograr un aprendizaje computacional insospechado, inaudito y tan amplio que rompe barreras de almacenamiento, capacidad de proceso y formas de aplicación. Lograr una comunicación natural ahora requiere del aprovechamiento de recursos, tanto físicos como lógicos, que procesen, identifiquen y respondan con la suficiente rapidez y exactitud que el usuario espera y que un sistema de interacción requiere, dos conceptos que juntos provocan que la interacción sea considerada válida.

Y es justamente, este punto de inflexión en el desarrollo técnico, aquel que marca la búsqueda y desarrollo de interfaces naturales de usuario, cada momento más “humanas”. Las Interfaces Naturales de Usuario (Natural user interfaces, NUI siglas en inglés) las cuales buscan aprovechar toda la riqueza, tanto física como sensorial, del usuario para interactuar con la máquina. Las INU, tratan de lograr una comunicación Hombre-Maquina de una manera “natural”, es decir, la interfaz es quien está sujeta ahora a la expresión espontánea corporal y sensorial del usuario. Lo contrario, limitaría la riqueza de la expresión, su interpretación e impondría condicionamientos. No deberían existir ataduras físicas (cables, botones, superficies) para la captura de datos, y, la expresión debería estar recogida directamente del medio en el cual se desarrolla, contextualizando su forma y sentido. De ahí que, en la búsqueda de una INU que logre su objetivo, se han empleado métodos científicos y recursos tecnológicos, tanto nuevos como los ya existentes, que logren definir técnicas y métodos de: captura, clasificación y reconocimiento basadas en imagen, sensores y superficies de contacto. La INU entonces, se presenta como la mejor opción para alcanzar la interacción basada en el usuario y en la riqueza gestual innata del ser humano.

El gesto, admitido como una forma de interacción natural y, usado como la expresión humana de una idea o concepto, logra un valor agregado que es justamente el concepto que define Palovic et al. [136] como el poder para reflejar el deseo, actitud u opinión en una acción. Por ejemplo, un gesto de “OK” (pulgar arriba, restantes dedos juntos en la palma de la mano), puede ser interpretado como “todo

esta bien” en varios países del mundo, sin necesidad de un lenguaje hablado. La postura de un dedo sobre la boca (dedo índice extendido y junto a los labios, los restantes juntos en la palma), dentro de este mismo contexto, significaría “hacer silencio”.

La idea de alcanzar un lenguaje gestual que universalmente represente un concepto sería lo idóneo, pero aquello, implicaría una difusión muy amplia y un entendimiento total, cosa que aún esta muy lejos de ser alcanzada. Sin embargo, en este trabajo planteamos los principios de diseño arquitectónico para la construcción de interfaces gestuales que incorporen un lenguaje basado en el gesto. Sistemas que logren un comportamiento único (comando) y/o personalizable, fundamentado en la optimización y la reutilización del gesto durante la interacción. Y, todo ello aplicado al control y supervisión de sistemas y dispositivos.

El gesto puede ser definido de varias maneras, tratado y conceptualizado de diferente forma. Por ello, es necesario profundizar en su clasificación, el concepto asociado y su aplicación en sistemas de interacción que lo implementen. Todo ello constituye la principal preocupación de este trabajo.

Sin embargo, podemos mencionar que el gesto consciente, ya sea corporal, facial y manual, es atributo propio e intrínseco del ser humano. Y quizá, de toda la riqueza gestual mencionada de aquel, los gestos manuales son quizá los de mayor alcance en la comunicación. Las manos poseen: la plasticidad y adaptabilidad que muchos conceptos requieren; la intensidad que la expresión necesita; y, la adaptación al contexto. Todos estos, componentes indiscutibles de la comunicación. Por ello, la riqueza del gesto manual abre nuevas y novedosas expectativas en la Interacción Humano-Ordenador (Human Computer Interaction, HCI por sus sigla en inglés) y más aún en la INU, al orientar su aplicación en la interfaz y conseguir con ella realizar tareas, tanto simples como complejas.

Dejar a un lado el uso de medios físicos (botones, teclado y mouse) es el nuevo reto y el gesto es quien acepta el desafío. No obstante, nacen inquietudes: ¿Será posible su aplicación en ambientes del hogar y la industria?, ¿Podrá adaptarse al control de dispositivos eléctricos y electrónicos?, ¿Será posible adaptar del gesto a nuevas arquitecturas en el diseño de la interfaz?, ¿Cuál son los dispositivos que mejor se ajustan a su captura?. Estas y otras interrogantes podrán encontrar respuesta en la presente investigación, en la cual tratamos de orientar aspectos de diseño, captura e implementación, además de validar nuevas propuestas tecnológicas.

Así pues, desde los primeros Guantes de Datos (DataGlobe, en inglés) de propósito general como: DataGlobeTM y Z-Glove[®] desarrollados por VPL investi-

gadores (California) [205] y presentado en la Conferencia sobre Factores Humanos en los Sistemas Informáticos y la Interfaz Gráfica en 1987. Hasta, aquellos que incorporan multitud de sensores como: el desarrollado por Asokan et al. [8] en la Conferencia sobre robótica y automatismos para aplicaciones humanas en 2016. Todos han tratado de capturar el gesto para que sea usado en la interfaz [141]. Muy pocos han logrado permanecer en el mercado, ya sea porque sus compañías han salido del negocio o, porque nunca fueron desarrollados comercialmente.

En este sentido, también debemos considerar a aquellos productos que habiendo sido lanzados inicialmente con un propósito específico, pueden ser adaptados y usados para otros objetivos. Entre ellos, aquellos dispositivos sacados al mercado para ser usados en juegos con el ordenador e impulsados por grandes compañías (Microsoft, Nintendo, Sony, etc). Entre estos productos tenemos a Microsoft Kinect, Nintendo Wii, DataGlobe, GoogleGlass, entre otros. Todos éstos productos, tratan de explotar el poder computacional del ordenador y establecer una comunicación transparente con el usuario a través de la interacción natural y por ende del gesto, en cualquiera de sus expresiones. El limitante, en muchos dispositivos, son las restricciones en el Kit de desarrollo de software (SDK) impuestas por el fabricante.

Desde otro punto de vista, también es necesario acudir a leyes o principios de diseño, pensados en contextos específicos. Algunas de estas leyes como Fitts [52] y sus variantes, han ayudado en la búsqueda de mejoras en las tradicionales Interfaces Gráficas de Usuario (GUI, por sus siglas en inglés), para hacerlas más eficientes, claras y sencillas. En la actualidad, es necesario crear nuevos conceptos que se adapten al gesto, a su ámbito, a sus expectativas y a sus limitaciones, hasta lograr los principios de diseño de Interfaces Naturales de Usuario basadas en Gestos (INU-G). Se hace necesario entonces, evaluar otras variables como: el confort, la fatiga en el usuario, el reconocimiento del gesto en la interfaz y la definición de un lenguaje gestual, entre otras, para ser usadas en la interacción. Medir el confort, proponer interfaces gestuales factibles y aplicarlas en varios ámbitos es también nuestro objetivo.

Finalmente, en esta investigación planteamos el uso del gesto en la Interacción Humano-Máquina (Human Machine Interaction, HMI en inglés) pasando desde su captura, clasificación y reconocimiento, hasta su aplicación en el hogar o, inclusive, proponiéndolo como una ayuda en interacción a usuarios que lo requieran. En nuestro recorrido, realizamos un estudio partiendo por aquellos dispositivos que fundamentan el reconocimiento basados en imagen (ópticos), pasando por aquellos que se basan en sensores (señales electrimográficas), hasta aquellos que usan su-

perfiles táctiles (touchscreen). Tratamos de lograr una propuesta al plantear una forma de sintaxis del lenguaje gestual, que tendrá amplias perspectivas en la industria y el hogar, planteado el uso del gesto, como un comando simple, para llegar al gesto, como un comando complejo.

Por todos los aspectos indicados, esta investigación pretende convertirse en un referente para lograr un mejor enfoque y expectativas, de uso de las INU-G, en el desarrollo y aplicación del gesto a la interfaz. Cada capítulo presenta un resultado experimental que deja testimonio de la investigación profunda realizada para lograr la implementación del gesto a la interacción humano-máquina.

Objetivos

Lograr el control y supervisión avanzada de sistemas basados en el uso del Interfaces Naturales de Usuario de tipo Gestual (INU-G).

El objetivo general, nos plantea los siguientes objetivos específicos:

- Establecer los componentes gestuales en la interfaz, su taxonomía y métodos de captura y reconocimiento gestual empleados.
- Analizar los métodos de captura y reconocimiento del gesto de la mano usando dispositivos basados en imagen.
- Investigar la captura y reconocimiento del gesto basado en sensores.
- Estudiar la captura y reconocimiento del gesto basado en superficies de contacto y proponer una forma de interacción basada en el gesto libre de la mano.
- Establecer los principios de diseño de un lenguaje gestual complejo basado en comandos gestuales simples. Como parte de esta etapa de la investigación se plantea determinar la confortabilidad del gesto, componente principal de las interfaces INU-G.

Estructura

Las tareas desarrolladas para cumplir con los objetivos planteados se encuentran organizadas en esta memoria en seis capítulos, los cuales se describen a continuación.

En el capítulo 1 se presentan los conceptos teóricos fundamentales que fueron estudiados a profundidad para llevar a cabo esta tesis. En él, se realiza un enfoque preliminar de la dificultad de la interfaz actual en contraposición a una interfaz más natural de usuario de tipo gestual. Se logra definir que el gesto es la expresión de un concepto mental y, como tal está sujeto a clasificación. Además, se examinan detenidamente las formas más comunes de reconocimiento gestual, logrando definir características y variables que permitirán su mejor aprovechamiento y aplicación. Por último, se hace un breve enfoque al uso de aquellos dispositivos, que han logrado los mejores resultados (ópticos, sensores, electrodos). Esta revisión nos permitió alcanzar las contribuciones principales de esta tesis y que se presentan en los capítulos siguientes.

logrando realizar un análisis del uso y aplicación del

El capítulo 2 esta orientado a describir el proceso de reconocimiento gestual, usando dispositivos ópticos. Se ha usado al controlador o también llamado sensor Leap Motion®, cuyo principio de funcionamiento se basa en tecnología óptica e infrarroja. Un profundo análisis de este dispositivo ha sido realizado, logrando establecer los fundamentos de reconocimiento gestual aplicados. En este sentido, se ha logrado establecer métricas de reconocimiento gestual y, además, implementar una biblioteca que logra la comparación entre gestos predefinidos por el sensor y aquellos implementados por nosotros. Los resultados obtenidos, han servido de fundamento para la publicación de un manuscrito, que versa sobre lo tratado en este capítulo y lo aplica exitosamente en el desarrollo e implementación de una interfaz gestual de control robótico llamada LEAPEV3. Esta interfaz implementa una INU-G sobre un sistema robótico.

En el capítulo 3 se analiza los principios de generación, captura y aplicación de señales electromiográficas (EMG) enfocadas a lograr el reconocimiento gestual. Hemos usado a la banda gestual MYO® para capturar las señales EMG a través de sus ocho sensores sEMG ubicados en el antebrazo del usuario. En este capítulo, hacemos uso de técnicas de Minería de Datos (Data Mining, en inglés) aplicando algoritmos de Aprendizaje Automático (Machine Learning en inglés) para lograr la clasificación y reconocimiento gestual. Dos populares algoritmos de ML, comúnmente aplicados en este tipo de señales, han sido comparados en exactitud de clasificación sobre una amplia biblioteca gestual analizada. Los resultados nos indican, que la exactitud alcanzada usando el algoritmo de Máquinas de Soporte Vectoriales (SVM, siglas en inglés) con kernel polinomial logra los mejores resultados. En este capítulo además, se realiza un análisis profundo del uso de señales

EMG, tipos de electrodos y selección de características basadas en el tiempo aplicadas en una señal EMG. Todo esto permitió lograr resultados de clasificación y reconocimiento gestual con una carga computacional mínima. Al igual que en el capítulo anterior, el análisis y sus resultados sentaron la base para una dos publicaciones científicas, en las cuales se vierte todo el estudio realizado y sus resultados, logrando establecer criterios de diseño para un uso óptimo del gesto, expandiendo sus posibilidades de aplicación al convertirlo en un control universal de dispositivos de casa.

El capítulo 4 se realiza un minucioso y amplio estudio del uso del gesto por toque en superficie táctiles conocidas normalmente como TUI (TouchScreen User Interface), en comparación con nuestra propuesta de Interfaces de Usuario basadas en Gestos (Gestural-based-on User Interface, GBUI). Para lograrlo se acude al smartphone como un dispositivo de alcance universal en tareas de comunicación, y sobre aquel se evaluaron los dos tipos de interface. Un estudio basado en la posición anatómica del usuario fue ejecutado, el cual determinó que las interfaces GBUI con una alternativa válida en interacción. Se acudió a principios de diseño y construcción de interfaces, como Fitts, para lograr una valoración del grado de dificultad (TUI) y exactitud (TUI-GBUI), demostrando que la interfaz GBUI es independiente de la posición del usuario. Los resultados de este capítulo, sirvieron para que una nueva publicación, en una revista de alto impacto, sea realizada. El manuscrito, logra plasmar todo lo investigado y, además, presenta una propuesta de GBUI para dispositivos móviles smartphone, llamada SICLLE y diseñada para la gestión de llamadas telefónicas. Por sus excelentes resultados, es recomendada especialmente para personas con necesidades puntuales ya sea por salud o enfermedad.

En el capítulo 5, se establecen los principios necesarios para alcanzar una estructura jerárquica que posibilite la construcción de comandos gestuales complejos, a partir de comandos gestuales sencillos, a la cual hemos denominado Árbol de Control Gestual (Gesture Control Tree, GCT). Esta estructura, ha sido probada en los capítulos anteriores y por los buenos resultados obtenidos en su implementación, en este capítulo se la define formalmente. Junto a ella, y como parte del estudio del gesto, hacemos un análisis del grado de confort del gesto, desde el punto de vista del usuario, de aquellos que conforman el catálogo gestual usado en esta tesis. Ambos estudios, GCT y confort, logran establecer los principios de un protocolo gestual orientado al control de dispositivos eléctricos y electrónicos. Estos resultados, fueron expuestos en un artículo de investigación que agrupa toda la propuesta

y valida los resultados aplicándolos en una interfaz de control gestual dedicada al control de dispositivos de casa.

Por último, en el Capítulo 6 se presentan las conclusiones a las que nos han conducido nuestros experimentos ya análisis en el desarrollo de esta tesis. Las conclusiones tratan de resumir de forma general todos los resultados parciales relevantes que se han obtenido a lo largo de nuestra investigación. Se incluyen además, las publicaciones realizadas durante el período de desarrollo e investigación de la misma. Así mismo, se exponen posibles líneas de investigación futuras, dentro de las cuales se puede advertir trabajos potenciales y de relevancia, lo cuales podrían desarrollarse para dar continuidad a esta investigación. Finalmente, se presentan dos apéndices que complementan las aportaciones realizadas. El primero, contiene un breve resumen de los principales documentos analizados y usados en esta investigación. El segundo, engloba todos los artículos publicados en congresos y revistas que han servido de fundamento en esta investigación.

Capítulo 1

El gesto y su interfaz

Para establecer la comunicación entre el usuario y el ordenador, la máquina o el sistema, es necesario establecer un vínculo de intercambio de información [147]. El vínculo de comunicación entonces, puede requerir de un espacio (el lugar), un instrumento (teclado, mouse, pantalla, cámara, sensor) o una superficie (objeto que aporta información) a los cuales comúnmente se les llama interfaz. La interfaz es la que determina la forma de intercambio de información (comunicación) o de interacción entre dos o más entidades, llámese usuario, sistema o la máquina. Un ejemplo de interfaz en informática, son las llamadas interfaces de usuario, que en muchos casos, pueden involucrar a dos ámbitos que son el hardware (por ejemplo, el teclado, el ratón, la pantalla, etc.) y el software (por ejemplo, un sector de la pantalla, un botón en la aplicación, o una opción de la ventana gráfica de un programa). Además, podemos encontrar también interfaces entre dispositivos que pueden abarcar tanto el nivel físico (puertos usb, ethernet, SCSI) como el lógico (programas, la API, controladores).

La Interacción Humano Ordenador (Human Computer Interaction, HCI), al momento, tiene un gran limitante en aquellas interfaces que no admiten o contemplan, en su diseño, la riqueza y la complejidad de la expresión gestual humana, la cual se manifiesta en aspectos tales como: el movimiento, el gesto, el pensamiento, la emoción, etc. La Interacción Multimodal (Multimodal Interaction, MMI) trata de combinar entonces varias técnicas y métodos de percepción (captura de información) y comunicación (transferencia entrada/salida de información) para sumarlos en la HCI [147]. En consecuencia, la búsqueda de la mejor forma de interacción se centra en lograr una forma de comunicación con la máquina (ordenador, maquinaria, robot, etc.) que no requiera: teclado, ni mouse, ni pantalla. Su objetivo, es alcanzar una forma de comunicación que no involucre tocar ninguna superficie du-

rante el manejo, operación o gestión de un dispositivo, y en su lugar, sea el usuario el principal vínculo de la interacción. A esta forma de comunicación, en la cual el usuario es el principal componente de la interfaz, se la ha definido como Interfaz Natural de Usuario (Natural User Interface, NUI) y es aquella en la cual la interfaz esta centrada en el usuario.

La NUI aprovecha cada componente individual del todo (usuario), para tratarlo como un mensaje, cuyo contenido podrá ser reconocido e interpretado por la máquina. La riqueza de la expresión humana incluye movimientos, gestos, expresiones y hasta pensamientos, para lograr la interacción con la máquina. Las NUI tratan, en lo posible, de integrar hardware y software, al amparo de las Nuevas Tecnologías de la Información y Comunicación (NTIC), en dispositivos que estén lo más cerca posible del usuario. Logran, se esta manera, un intercambio de información natural y eficiente, al minimizar errores, incrementar la satisfacción y disminuir la frustración; en pocas palabras, tratan de hacer más productivas las tareas que involucran a las personas y a las máquinas [147].

Para reducir las limitaciones en la comunicación con la máquina, en la actualidad, se consideran las ventajas de la MMI junto a los avances de la tecnología informática y electrónica. Esta combinación, ha logrado el desarrollado nuevos conceptos para la captura de información, diseño, implementación y control, plasmados en novedosos dispositivos llamados portables (wearables). Estos dispositivos logran integrar nuevas variables, insospechadas en muchos casos, a la interacción. Por ejemplo, dispositivos con Unidad de Medición Inercial (IMU) que permite conocer al mismo tiempo velocidad, orientación y fuerzas gravitacionales combinando acelerómetros y giroscopios; o el uso de bio-señales obtenidas del usuario e interpretadas por un sistema que logra medir movimiento, ritmo cardíaco, pulso, etc., por solo citar unos ejemplos. Todas, variables obtenidas en tiempo real, que logran instanciar la información, y que ahora muchos sistemas han incorporado a la interacción, lo cual era impensable hasta hace pocos años atrás.

1.1. El gesto y sus características

La primera forma de comunicación humana históricamente aceptada, es el gesto. Corballis [31] sostiene que, el lenguaje evolucionó a partir de los gestos manuales, e incluso menciona que, existe un fuerte predominio de la mano derecha como una característica propia de la especie humana. Por ello, la comunicación gestual tiene un contexto, en ocasiones tan amplio y complicado como la riqueza de pensamiento

del hombre. La expresión gestual del hombre puede ser analizada desde varios aspectos, pero básicamente, el gesto debe seguir los principios del lenguaje natural combinado con la necesidad de optimizar contrastes como: su representación, su significado, su dificultad y complejidad intrínseca. En este sentido, podemos decir que el gesto depende de su forma de representación, del posible significado y de la dificultad que imponga al ser humano tanto el poder realizarlo y el saber reconocerlo e interpretarlo.

1.1.1. Representación

La representación del gesto, ha sido enfocada en el mundo real desde la perspectiva de la dimensionalidad. El gesto, puede ser definido como una postura o gesto estático, a partir de una forma o posición de los dedos y de la mano. También se lo llama gesto simple, y básicamente se trata de adoptar una posición que exprese una idea [136]. Un ejemplo podría ser un gesto de aceptación “ok”. En cambio, un gesto complejo trata de adoptar una forma establecida que puede ser vista desde varios ángulos, a partir de los cuales, la postura se transforma. A este tipo de gesto se le llama dinámico y un ejemplo de aquel podría ser una la señal de “adios” [120, 136].

Mitra et al. [120] define al gesto dinámico como aquel que tiene tres componentes: movimiento de dedos, movimiento de la muñeca y los cambios en la posición y orientación. Un ejemplo, muy recurrente en este sentido, es el Lenguaje de Señas Americano (American Sign Language, ASL).

La representación también puede ser alcanzada usando modelos en 2D o 3D a partir de los datos capturados y en muchos casos usando una base de datos que almacena las características necesarias para representar la mano. Problemas tales como los Grados de Libertad (Degree Of Freedom, DOF) y la plasticidad de la mano como objeto deformable articulado, han dificultado el proceso de extracción de características y representación [15]. En aplicaciones que usan el gesto basados en la dinamicidad, como aquellas que aplican ASL, los gestos son tratados como una secuencia de posturas conectadas por movimientos continuos, en donde la forma de la mano y apariencia son fundamentales [151, 136].

1.1.2. Significado

El significado del gesto es múltiple, diverso y condicionado, en muchos casos, al contexto, ubicación geográfica, costumbres y conceptos preestablecidos que lo definen. El movimiento gestual ha sido parte del lenguaje usado por el hombre [31] y, si a este movimiento le agregamos un significado, entonces el lenguaje basado

en señas es un claro ejemplo. Pero también es cierto, que un mismo lenguaje puede tener significados diferentes a tal punto que existen varios lenguajes de señas como LSA (Lenguaje de señas americano), LSE (lenguaje de señas español), LSC (lenguaje de señas catalán), LSG (lenguaje de señas alemán), etc. Cada lenguaje tiene diferentes alfabetos y juegos de palabras, y las similitudes son creadas por movimientos de las manos y apoyadas por señas de cabeza, ojos y expresiones faciales. La necesidad de lograr un lenguaje gestual estandarizado en la interacción es evidente.

El significado va de la mano del reconocimiento, y también esta sujeto a parámetros de significancia previamente establecidos o configurados en un sistema informático. Desde la captura del gesto hasta su reconocimiento, llevan consigo mucho del significado dado a un gesto determinado. Esto nos hace pensar, que un lenguaje gestual estandarizado es necesario, el cual contenga reglas, cumpla condiciones y establezca restricciones, propias de un lenguaje.

1.1.3. Dificultad

La dificultad del gesto puede ser planteada desde dos puntos de vista: El humano y aquel determinado por la tecnología empleada para el reconocimiento.

Desde el punto de vista humano, la dificultad del gesto esta dada por variables propias a las posturas adoptadas por las manos o brazos, tales como: dolor, incomodidad, fatiga, etc. En este sentido, Rempel et al. [151] realizaron un estudio con veinticuatro (24) intérpretes profesionales del lenguaje de señas para lograr establecer un rango de confort para un muy amplio catalogo gestual sugerido por [58] y muy usado en el lenguaje de señas americano. En este trabajo, se determinan asociaciones claras de malestar con determinadas posturas de las manos. Además, concluyen que deben tomarse en consideración los parámetros de dificultad del gesto, para crear bibliotecas gestuales que puedan ser usadas en interacción hombre-máquina. Hacerlo, lograría optimizar la relación entre los procesos cognitivos y físicos del ser humano y sistemas de reconocimiento gestual.

Desde el punto de vista técnico y de reconocimiento, la dificultad se centra en los aspectos, a saber: la tecnología usada en la detección o captura del gestos y el tratamiento computacional necesario para lograr el reconocimiento. La primera, envuelve la detección o extracción de características más importantes y sus parámetros, lo cual implica la riqueza en captura de datos, los cambios en función del tiempo o frecuencia, entre aspectos externos como: características del medio, ruido, retardo, ect. La segunda, el tratamiento computacional de las característi-

cas tendientes a lograr un adecuado reconocimiento implica, el uso de técnicas estadísticas avanzadas, algoritmos de clasificación, amplio set de datos, parámetros óptimos de clasificación, ect. Al respecto, muchas y muy variadas técnicas se han propuesto usando procedimientos de minería de datos y algoritmos de aprendizaje automático.

1.2. Taxonomía

La literatura revisada nos indica, que el ser humano es portador de una amplia gama de información contextual, conocimientos generales, antecedentes culturales, capacidad lingüística y cognición. Esta información permite definir el mensaje y la interpretación de los gestos. Como ya se ha dicho en líneas anteriores, los gestos han sido usados desde hace mucho tiempo por el hombre como una forma de comunicación, y se vuelven casi inherentes a la naturaleza del ser humano, es decir, son deducibles por éste para entender o comprender aquello que se desea expresar. De allí, justamente, la importancia del uso del gesto en la comunicación. Un ejemplo práctico de lo dicho, es la forma en la cual los auxiliares de operación de vuelo, en un aeropuerto, ayudan al piloto a desplazarse en la pista hasta tomar su lugar de despegue o de amarre. También, se hace una consideración en relación a la naturaleza del gesto, es decir la forma como son concebidos los gestos. Al igual que el caso anterior, se analiza el gesto, pero tomando en consideración las variables que podrían llegar a identificarlo o como lograr su formación.

1.2.1. En la comunicación

Los mecanismos por los cuales el ser humano logra la comunicación, usando gestos, son muchos y variados. Al respecto, Francis Quek [147] y posteriormente Pavlovick et al. [136], dividen a la comunicación gestual en dos grupos: Gestos Actitudinales o de Acto y Gesto Simbólicos o de Símbolo. Cada uno de estos grupos describe a su vez los siguientes subconjuntos gestuales: Miméticos, Deícticos, Referenciales y Modales. La Figura 1.1 muestra la taxonomía en forma gráfica.

Gestos Actitudinales

Los **gestos actitudinales** o de acto, son aquellos en los cuales los movimientos gestuales realizados se relacionan directamente con la interpretación pretendida. En muchos casos, la forma de expresarlos o su contexto pueden poner la diferencia al momento de su interpretación. Un claro ejemplo de este tipo de gestos, es el uso

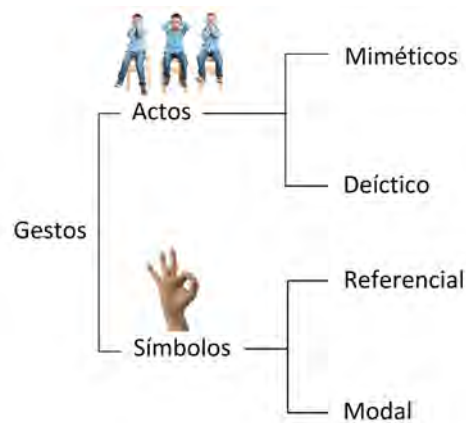


Figura 1.1: Taxonomía del gesto desde el punto de vista de la comunicación [147]

de los dedos y manos, logrando formas o movimientos en dirección a otra persona, dentro de un contexto de agrado o enfado.

- Miméticos, son gestos que tratan de imitar u ocultar su mensaje. En la naturaleza, un ejemplo práctico es el camaleón. Se puede decir que, hacen pantomima o reflejan su intención.
- Deícticos, son gestos que implican variables de tiempo, lugar y persona. Pueden ser Específicos, Genéricos y Metonímicos, diferenciados únicamente por el contexto.
 - Específico puede ser cuando indicamos algo en un mapa.
 - Genérico, aquel que identifica la clase de un objeto.
 - Metonímico, aquel que se realiza para dar una semejanza de aquello que deseamos comunicar. Por ejemplo, un gesto indicativo a la imagen de una granada podría usarse para indicar el nombre de nuestra ciudad.

Gestos Simbólicos

Los **gestos simbólicos** pueden considerarse como una representación gráfica de la intención o mensaje, que por lo general lleva intrínseco en su imagen o representación. El Lenguaje de Señas (LS) es un ejemplo de gesto simbólico.

- Referencial, aquel que opera en forma independiente para designar objetos o conceptos. Uno muy usado, es el gesto al frotar el dedo medio e índice al pulgar para referenciar al dinero.

- Modal, aquel que se sirve para en combinación con otros recursos llegar a describir algo, ya sea objeto o concepto. Un ejemplo, cuando estamos en una fiesta y no logramos escuchar a alguien que nos pregunta algo y hacemos gestos para indicar que hable más fuerte o se acerque al oído, en combinación con otros que indican que no escuchamos, como podría ser, el mover nuestras manos a la altura de las orejas.

1.2.2. Por su naturaleza

La literatura clasifica a los gestos en dos tipos: dinámicos o complejo y estáticos o simples. El gesto estático esta definido como la orientación y posición de la mano en el espacio durante un determinado tiempo y sin ningún movimiento. Si la última condición se rompe, y existe movimiento, el gesto pasa a ser dinámico. Un gesto dinámico puede ser considerado también como un camino entre una posición inicial y una posición final. En muchos casos un gesto dinámico involucra también movimiento de otras partes del cuerpo, como puede ser antebrazo, brazo, o el cuerpo en su conjunto.

También podemos decir, que si bien el gesto expresa información que depende del contexto, la naturaleza del gesto indica en cierta medida su alcance, agregando forma y movimiento. En cada caso, información y alcance de dicha información, pueden ser tratados y conceptualizados en forma diferente. La revisión bibliográfica nos permite clasificar por su naturaleza en:

El gesto corporal Este tipo de gesto está definido por el uso del cuerpo, en su conjunto o sus partes, vistas como un todo tendiente a lograr una expresión. Dicha expresión llevará un mensaje implícito, que emisor y receptor deberán interpretar de la manera en la cual se hayan puesto de acuerdo. Todas las partes del cuerpo expresarán un léxico que puede ser ampliado y mejorado, dependiendo del contexto y su objetivo.

El gesto manual El gesto manual es aquel que involucra a brazo, antebrazo, manos y dedos para alcanzar una pose o postura específica. Puede ser en forma separada, es decir usando una mano y sus dedos solamente, o usando las dos manos para formar un solo gesto. Su interpretación también esta sujeta a un contexto y objetivo, pero con la ventaja de que la plasticidad de la(s) mano(s) pueden llegar a determinar un lenguaje o expresión aún más extensa que la anterior.

El gesto facial El gesto facial, involucra a los componentes que conforman el rostro humano, tales como ojos, boca, nariz, cejas, párpados, orejas. Todos realizando una expresión facial o un movimiento, ya sea en conjunto o en forma separada. Su lenguaje puede ser definido e interpretado, al igual que en los otros casos, pero en ocasiones se ve condicionado al estado de ánimo, comportamiento o reflejo de aquello que lo provocó. Se dice entonces que el gesto facial, es una representación del estado de ánimo de una persona y nos permite conocer detalles intrínsecos de la naturaleza humana.

1.2.3. Gesto estático y dinámico

De los diferentes tipos de gestos, los gestos de la mano realizados conjuntamente con sus dedos, son los más expresivos y los más usados. Según Mitra & Ancharya [120], esto implica que: una postura es una configuración estática de los dedos sin movimiento de la mano, y un gesto sea un movimiento dinámico de la mano, con o sin movimiento de los dedos. Por lo que podemos definir al gesto estático como aquel en el cual el usuario asume una determinada pose o configuración. Mientras, un gesto dinámico es aquel que tiene varias fases (antes, durante y después). Por esto también se denomina a un gesto dinámicos como gesto complejo, y a un gesto estático como gesto simple.

Atendiendo a la clasificación del gesto realizada por [150], en base al movimiento, podemos obtener la Figura 1.1 en la cual se puede observar la taxonomía del gesto de la mano y sus categorías. Los gestos dinámicos son necesarios en casi todos los contextos y se los realiza en forma consciente o inconsciente, y durante la comunicación juegan un papel importante. De acuerdo a Rautaray & Agrawal [150] un 35 % de la comunicación humana se realiza en forma verbal y un 65 % en forma gestual o no verbal.

Los **gestos emblemáticos** son conversiones cortas directas de la comunicación verbal. Por ejemplo el gesto de “Adios”. Por lo general, este tipo de gesto varía de acuerdo a la cultura, lugar, región, entre otros factores. Aquellos gestos que tratan de transmitir una emoción o una intención son llamados **gesto afectivos**. Los gestos afectivos son, por lo general, conocidos mundialmente y están, en su mayoría, libres de la cultura. Por ejemplo, estrechar las manos entre dos personas, o juntar las manos en oración. Los gestos que controlan la interacción entre dos personas son llamados **gestos reguladores**. En ocasiones, estos gestos tratan de liberar la tensión del cuerpo (por ejemplo, alzar los brazos luego de una larga jornada de trabajo frente al computador). En muchos casos son gestos que se realizan en forma



Figura 1.2: Taxonomía basada en el tipo de gesto

no intencional durante la comunicación.

Los **gestos ilustrativos** enfatizan en puntos claves del discurso, para describir las declaraciones realizadas en la comunicación. Por ejemplo, el hecho de mover la mano durante un discurso o cerrar los puños para enfatizar fuerza, en el mismo contexto. Estos gestos son dependientes del comunicador, sus declaraciones en atención a su pensamiento y por supuesto del habla. Este tipo de gesticulaciones (hacer el gesto mientras habla) han sido clasificadas en cinco subcategorías: gestos de sentimiento, gestos deícticos, icónicos, metamórficos y cohesivos.

Los **gestos de sentimiento** son cortos y rápidos, rítmicos por lo general luego repetitivos. Expresan afecto. Por ejemplo, un abrazo seguido de palmadas en la espalda. Los **gestos deícticos**, son aquellos que apuntan a un objeto o persona, tratando de dar su localización real o tratando de resumir una localización abstracta o un período de tiempo. Por ejemplo, uno entre varios comunicadores apunta con el índice al reloj y luego acompaña un movimiento de la mano con los dedos extendidos, podría interpretarse como el hecho de que, quien realiza el gesto tiene el tiempo justo y ya tiene que marcharse. Los movimientos de las manos que tratan de representar o figurar una acción, por ejemplo, moviendo las manos tratando de simular el tomar una cuerda, podría representar la acción de escalada. A este tipo de gesto se lo llama **gestos icónicos**. Los **gestos metamórficos** tratan de representar abstracciones. Los **gestos cohesivos** son aquellos que temáticamente están relacionados, pero se encuentran temporalmente separados. La separación temporal de estos gestos temáticamente relacionados se debe a la interrupción del comunicador actual por cualquier otro comunicador.

1.2.4. El gesto en interacción

Conforme a lo expuesto anteriormente, la comunicación gestual presenta muchas ventajas, pero es necesario conocer también su alcance. En otras palabras, los gestos podrían tener limitantes de acuerdo a su ámbito, entendiéndose que éstos, son el alcance de un determinado grupo gestual (esto se trata más en detalle en la Sección 1.3). Sin embargo, debemos recordar que el o los gestos son la expresión de una idea, pero al mismo tiempo, pueden representar en si mismos todo el concepto de su entorno. Por ejemplo, un gesto corporal de frío, indicaría a otra persona que hace frío, que estamos con frío o que necesitamos abrigo. El concepto en esta idea es el “frío”.

En consecuencia, el único limitante del gesto y la comunicación gestual es la forma de captura, reconocimiento e interpretación que deberá establecerse en la interfaz gestual. La interfaz, en muchas ocasiones, está condicionada a la capacidad científica y tecnológica del momento. Sin embargo, es posible concluir, que la interacción gestual puede ser lograda en tres campos principales, concretamente al uso de Gestos Corporales (GC), Gestos Faciales (GF) y Gestuales Manuales (GM). La Tabla 1.1 muestra las limitaciones gestuales, entendidas como el límite en cada conjunto.

1.3. Captura y reconocimiento gestual

El uso del gesto en la interacción abre las puertas a nuevas formas de comunicación con la máquina y, su desarrollo y aplicación, va de la mano con la evolución de los dispositivos usados.

La captura y reconocimiento del gesto es un problema que ha sido abordado desde varios puntos de vista, tanto técnicos como metodológicos. Para resolverlo, se han empleado y al mismo tiempo desarrollado, varios dispositivos y métodos orientados a conseguir los objetivos de captura y reconocimiento gestual. En ambos casos, sometiendo a prueba la capacidad de dispositivos y la probando complejos y novedosos algoritmos de reconocimiento. Lo cierto es que, cada técnica establece límites de interacción.

En consecuencia, los dispositivos más recientes incorporan los últimos avances tecnológicos, pero, involucran también nuevas formas de captura y reconocimiento. Es un hecho entonces, que, al mismo tiempo, una creciente complejidad algorítmica se halle implícita en el proceso y, junto a ellos, variables que establecen una dependencia temporal y de exactitud. La captura y reconocimiento del gesto en tiempo

Tabla 1.1: Los gestos en interacción y su alcance

Tipo	Característica	Limitaciones
GC	Movimiento del cuerpo asociado a acciones. Aquí como ejemplo se podrían incluir los <i>mimos</i> . El gesto corporal incluye todas o casi todas las partes del cuerpo y sus posiciones, ya sea en conjunto en forma focalizada a una parte específica, pero principalmente involucra el cuerpo.	Implica el uso y expresión de todas las extremidades además de las expresiones gestuales faciales posibles.
GF	Movimientos de la cara, incluyendo ojos, nariz, boca e inclusive lengua. Algo comúnmente llamado mueca suele ser la expresión facial característica. Un gesto facial involucra la cara y todos los músculos que la controlan para lograr una expresión.	Por lo general expresa una idea o concepto, pero se limita únicamente a su contexto facial.
GM	Movimientos de las manos exclusivamente. Este tipo de gestos tratan de representar acciones. Permite realizar combinaciones de los gestos logrando un amplio conjunto que puede ser visto como un lenguaje. Un ejemplo, es el Lenguaje de Señas Americano (LSA) utilizado en la comunicación con personas con deficiencia auditiva-sonora.	Este tipo de gestos en muchos casos implica el uso de brazos y manos.

GC, Gesto corporal; GF, Gesto facial; GM, Gesto manual.

real es una muestra palpable de los logros alcanzados, sujetándose a restricciones temporales y sometándose a los límites de la técnica usada. Dichos límites, constituyen nuevos problemas, implícitos en la naturaleza del dispositivo y su técnica, tales como: en imagen la oclusión, en contacto la exactitud, y en sensores, la exactitud y precisión de la lectura.

Atendiendo a las características de reconocimiento empleadas por los dispositivos hemos creado la Tabla 1.2. En ella, se muestra los dispositivos más usados, algunos de ellos creados para usarse directamente en la captura gestual, componente o principio fundamental de captura, además de los problemas o limitaciones encontradas.

Las limitaciones existentes en la captura, como ya dijimos anteriormente, se encuentran relacionadas directamente con el dispositivo empleado. Mientras, que el método de reconocimiento, puede ser usado indistintamente y ajustado a la técnica empleada para su captura, dejando al investigador su aplicación, uso y posible mejora. Sin embargo, la complejidad algorítmica y de diseño, para lograr el reconocimiento, esta sujeta a fuertes requerimientos de tiempo de cálculo que

Tabla 1.2: Técnicas y sus características

Dispositivo	Componentes	Intrusivo	Principal restricción y limitantes
Guantes	Resistivos, magnéticos	Si	Conocimiento y adaptación al usuario. Debe ir junto al usuario y cerca de los sensores
Cámara	Óptico	No	Oclusión y condiciones lumínicas. La pérdida de enfoque es crítica
Touchscreen	Contacto	No	Precisión del usuario y área de toque. Debe realizarse directamente sobre el dispositivo
Kinect	Óptico, iR	No	Sujeto a distancia mínima de interacción. Oclusión. Límites Min. 20 cm y máx. 4 metros frente al dispositivo
Leap Motion	Óptico, iR	No	Área de interacción. Oclusión. Condiciones lumínicas. Límites Min. 5 cm sobre el dispositivo a máx. 60 cm
MYO	Electrodos EMG, IMU	Si	Ubicación crítica en el antebrazo. Alcance de la señal Bluetooth. Debe permanecer en contacto directo con la piel. Características físicas del usuario.

limitan su campo de acción y aplicación.

Junto a la complejidad algorítmica (relativa al software) se encuentran los datos y sus características. En consecuencia, las llamadas técnicas de Aprendizaje Automático (Machine Learning, en inglés) implementan algoritmos de predicción, los cuales hacen uso intensivo de los datos para resolver un problema del reconocimiento gestual. Principalmente, se basan en métodos de clasificación, los cuales procesan los datos obtenidos del dispositivo que previamente han sido organizados en un conjunto de datos, también llamado **dataset**. La gran ventaja, es que estos algoritmos pueden combinarse prácticamente con cualquiera de las técnicas expuestas para lograr el reconocimiento gestual, a partir del dataset enviado.

Por otro lado, la complejidad técnica (relativa al hardware) en ciertos dispositivos quedan totalmente fuera del alcance del usuario para su configuración, calibración y optimización. Esto, los convierte en dispositivos muy especializados y construidos para un propósito específico. En este punto, es necesario mencionar que en algunos casos, el uso continuo de ciertos dispositivos, pueden provocar problemas al usuario que se traducen en dolor y cansancio. Un ejemplo es Leap Motion, que requiere que se mantenga la mano o manos del usuario, realizando un gesto directamente sobre él, para de esta forma lograr su detección; o frente a él, en el caso de Kinect. En el caso de dispositivos que involucran contacto con alguna superficie

táctil, el ejemplo más idóneo es el dispositivo móvil o Smartphone. Y, en aquellos que requieren mantener posturas especiales de la mano podemos mencionar a la pulsera gestual MYO. Esta fenómeno de cansancio, fatiga o dolor ha dado lugar a nuevas investigaciones orientadas a medir el grado de confort del usuario, punto que también abordaremos en esta investigación.

Atendiendo a todo lo expuesto, y luego de una extensa revisión bibliográfica, hemos podido determinar que la captura del gesto puede ser realizada usando: dispositivos ópticos, dispositivos que incorporen sensores y dispositivos que requieran contacto con una superficie o pantalla táctil (Touchscreen). Estas tres formas de captura, agrupan a aquellas usadas para la captura del gesto. La Figura 1.3 muestra gráficamente esta clasificación, y además, trata de relacionar a cada técnica con los aquellos dispositivos que, al momento, el mercado tecnológico ofrece.



Figura 1.3: Formas de captura del gesto

Tomando como referencia los medios por los cuales se logra la captura del gesto, realizaremos un breve recorrido por cada uno de ellos, realizando un breve pero efectivo análisis de sus características, aplicabilidad y alcance.

1.3.1. Reconocimiento basado en dispositivos ópticos

El reconocimiento óptico o llamado también por imagen, es uno de los más extendidos, tanto en el ámbito científico como en el ámbito comercial. Pero, a pesar de ello, presenta ventajas y restricciones muy importantes que deben ser tomadas en consideración, pues su aplicación dependerá del sistema de reconocimiento gestual a implementarse. Este tipo de captura gestual, disponible para sistemas de reconocimiento distingue tres fases que son: detección, seguimiento y reconocimiento

propriadamente dicho [150]. Cada una de estas fases agrupa a otras etapas necesarias que se describen en cada una de ellas. La Figura 1.4 muestra las fases y los métodos más comunes empleados en cada una de ellas.

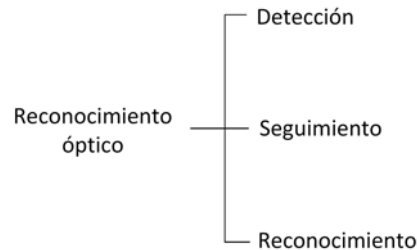


Figura 1.4: Técnicas de reconocimiento del gesto basadas en visión

Detección

El paso inicial en los sistemas de reconocimiento gestual es la detección de las manos y la segmentación de las correspondientes regiones de la imagen, lo cual es muy importante porque permite resaltar los datos relevantes o importantes a partir del fondo de la imagen, antes de pasar a las otras fases. Varios trabajos de investigación han usado: segmentación del color de la piel, características del contorno de la mano, valores de pixel (escala de grises), modelos 3D, y movimiento. Este último, el movimiento, es un enfoque bastante nuevo y requiere una configuración muy controlada, ya que asume que el único movimiento en la imagen se debe al movimiento de la mano [36]. Otros enfoques usan características como el color de la piel, sombra, movimiento y modelo anatómico de la mano. Un ejemplo de esto es el trabajo planteado por Simao et al.[170] y Alon et al. [5] para reconocimiento continuo del gesto basado en segmentación, y que puede ser aplicado a sistemas de reconocimiento de lenguaje de signos.

Seguimiento

El seguimiento depende directamente de la tasa de captura del frame (fotograma o instantánea) la cual debe ser lo suficientemente alta para lograr la captura de las imágenes para poder reflejar el movimiento observado de la mano. La idea es poder utilizar ciertas características que permitan identificar el movimiento como aquellas basadas en plantillas, estimación óptima (aplica filtro de Kalman) ¹,

¹El Filtro de Kalman convierte observaciones (detección de características) en estimaciones (trayectoria extraída). Alcanza un buen rendimiento en tiempo real, permite el tratamiento de la

filtrado de partículas ² (en desórdenes visuales densos), Camshift ³ (algoritmo de cambio medio).

Reconocimiento

El reconocimiento trata de lograr una interpretación de la semántica, que la mano transmite a través de su ubicación, postura y gesto. Toma como referencia, la clasificación de los gestos en estáticos y dinámicos. Para la detección de gestos estáticos (posturas), puede usarse un clasificador general o una plantilla (template-matcher). Mientras que, en gestos dinámicos, es necesario considerar el componente temporal, y requiere en su identificación técnicas que manejen esta dimensión (por ejemplo, Modelo Oculto de Markov), si la dimensión temporal puede ser modificada a través de la representación del gesto de la mano (por ejemplo, modelo basado en movimiento). Los gestos estáticos pueden ser clasificados también como: de aprendizaje lineal y no lineal [150]. Esta clasificación atiende al uso de técnicas de Machine Learning. El primero es usado en casos linealmente separables y el segundo para aquellos casos que no lo sean. Por otro lado, pero dentro del mismo ámbito, el reconocimiento acude a algoritmos de aprendizaje automático cuya aplicación y éxito depende del tamaño, la calidad y la naturaleza de los datos. También dependen de aquello que se desea hacer con la respuesta, de como se hayan traducido los cálculos del algoritmo para el equipo en el cual se ejecuta y, también dependen, del tiempo necesario y disponible para lograr un resultado válido.

Desde el punto de vista de los algoritmos de clasificación, el gesto es considerado un conjunto de estados y transiciones. Los estados representan un gesto estático (por ejemplo, una postura) y las transiciones representan cambios permitidos con restricciones temporales y/o probabilísticas [150]. Un gesto dinámico se considera entonces como un camino entre un estado inicial y un estado final. El principal inconveniente de este enfoque de algoritmos de clasificación es que el modelo usado para reconocimiento debe ser modificado cuando un nuevo gesto requiera ser incluido. Su complejidad computacional es alta, proporcional al número de gestos, lo cual no sucede en otras técnicas.

incertidumbre y logra la provisión de predicciones para los frames sucesivos [150].

²El filtrado de partículas permite modelar la ubicación de la mano en base a un conjunto de partículas [75].

³El algoritmo de cambio medio, es un métodos de seguimiento basado en núcleos que utiliza modelos de apariencia basados en densidad para representar objetivos[123]

1.3.2. Reconocimiento basado en sensores

Yang et al. [197] sostiene, que como la mano es un objeto deformable, no puede ser representada por un modelo simple y el rastreo y reconocimiento, de la mano humana, son fácilmente influenciados por factores ambientales como: la luminancia, color, etc., especialmente en guantes de datos [44].

El reconocimiento gestual basado en sensores ha tenido un gran apogeo en los últimos años. El sensor, es un dispositivo capaz de detectar magnitudes físicas o químicas llamadas variables de instrumentación (por ejemplo, temperatura, aceleración, inclinación, etc) y transformarlas a variables eléctricas (por ejemplo, resistencia, capacidad, tensión, etc). El sensor, siempre está en contacto con la variable de instrumentación y, en muchos casos, aprovecha las propiedades de la variable para adaptar su señal. Además, permite que aquella pueda ser interpretada por otro dispositivo (por ejemplo, termómetro) ⁴. También, se lo define como un dispositivo que convierte una forma de energía en otra. Los sensores más difundidos en diferentes áreas de análisis y reconocimiento de movimiento, posición, orientación, ubicación geográfica son por ejemplo, el acelerómetro, giroscopio, magnetómetro, GPS, etc.

Un sensor diseñado para medir o cuantificar las actividades humanas puede clasificarse como interno y externo [96].

Sensor interno

Un sensor interno, es aquel que toma la señal de la variable de instrumentación directamente del usuario. A este grupo pertenecen los denominados dispositivos portables (wearables) [193]. Dispositivos wearables han sido usados con amplia libertad y en la actualidad es posible encontrarlos en un gran número de sistemas especialmente de salud [23], bioseñales [193], control ambiental [144], control instrumental [168], etc. Sin embargo, a pesar de sus ventajas se lo considera como intrusivo, que sin afectar al cuerpo, es ajeno al usuario y su definición, en muchos casos, depende del contexto. Un claro ejemplo sería una pulsera diseñada para deportistas, en la cual se controla en el usuario ritmo cardíaco, pulso, temperatura, etc., pero exige que el usuario la lleve puesta durante el ejercicio.

⁴<https://es.wikipedia.org/wiki/Sensor>

Sensor externo

Un sensor externo, es aquel que captura la actividad del usuario pero sin estar en contacto directo con él. En otras palabras, no requiere que el usuario lo lleve consigo. Un ejemplo de este tipo de sensor, sería el uso de una cámara como parte de un sistema domótico inteligente [33]. El uso y mantenimiento de sensores externos por lo general es costoso y esta sujeta a restricciones de sistemas de su tipo, descritas anteriormente. Por estas razones, el usuario se ve motivado al uso de dispositivos wearables para medir movimientos del usuario (por ejemplo, acelerómetros, GPS), variables ambientales (por ejemplo, temperatura, humedad), señales físicas (por ejemplo, latidos del corazón, pulso). Los datos son ordenados en una dimensión temporal, permitiendo definir y encontrar problemas en la actividad humana.

Sensores en wearables

Un sensor wearable, no es más que un dispositivo portable que alberga a uno o varios sensores internos. Este nuevo grupo de sensores ha merecido especial atención en los últimos años, pues gracias a ellos se ha logrado medir, cuantificar, controlar, estimar, muchas y muy variadas variables físicas, fisiológicas, entre otras.

Como se expuso anteriormente, los dispositivos que tratan el reconocimiento gestual, incorporan cada uno sensores de diferente tipo (Sección 1.3) y puede ser empleados en detección, seguimiento y reconocimiento del gesto. Mientras que, un dispositivo wearable puede ser considerado en si mismos un sensor que alberga o contiene múltiples sensores.

La Tabla 1.3 pretende hacer un enfoque del sinnúmero de actividades que logran ser controladas, medidas o supervisadas por los dispositivos wearables, clasificados por grupos de acción.

El uso de sensores de múltiple naturaleza y su acoplamiento en infinidad de ambientes, ha llevado en la actualidad, concretamente a partir de los años 90, al desarrollo de una nueva disciplina llamada *computación pervasiva*. Esta disciplina, establece la capacidad de los sistemas para adquirir información (obtenida de sensores de muy variado tipo (por lo general de acuerdo a un contexto) del entorno de una persona, para ser usada en cualquier lugar y en cualquier momento (para una gran variedad de propósitos). Su complejidad dependerá de los objetivos a alcanzarse [73, 157]. Otras definiciones, la enfocan como la computación omnipresente, en la cual se asume que, una aplicación es un medio por el cual el usuario realiza una tarea, y no aquel software escrito para explorar las capacidades de un dispositivo [157]. Un claro ejemplo, de computación pervasiva u omnipresente (ubicua), son

Tabla 1.3: Actividades y uso de los wearables

Grupo	Actividad realizada
Parte superior del cuerpo	Hablar, respirar, mover la cabeza, masticar, actividad maxilar
Ambulatorio	Caminar, correr, cantidad de pasos, sentado, de pie, acostado, subir, bajar
Transporte	Tomar el autobús, conducir bicicleta, moto, coche, seguir una ruta
Uso del teléfono	Mensajes de texto, llamar, contestar, sonido, vídeos, fotos
Actividades diarias	Comer, beber, trabajar, leer, cepillarse los dientes, ver televisión
Militar	Rutas críticas, orientación, rastreo, prevención, evaluación de situación
Ejercicio y acondicionamiento físico	flexiones, nadar, correr, marcha, levantar pesas, spinning, marcha, saltar

los sistemas inteligentes de control domótico (para el hogar), en donde la ubicación del usuario es clave para intuir una orden y adaptarla al contexto.

Lara & Labrador [96], establecen cuatro grupos de atributos que son susceptibles de ser medidos usando sensores wearables, en el contexto de Reconocimiento de Actividades Humanas (Human Activity Recognition HAR, por sus siglas en inglés) y son: atributos ambientales, aceleración, localización y señales fisiológicas.

- **Atributos ambientales**, son aquellos tales como temperatura, humedad, nivel de audio, etc. Su principal intención, es proporcionar información contextual que describe el entorno del individuo [171]. Por ejemplo, si la iluminación es baja y el nivel de ruido es mínimo, la persona puede estar descansando. Este contexto es comúnmente acompañado de acelerómetros u otros sensores.
- **Aceleración**, es aquello concerniente al movimiento del individuo. En su mayor parte, los acelerómetros de tres ejes (x, y, z) son los más usados para reconocer actividades de deambulación (por ejemplo, caminar, correr, descansar, etc). En la actualidad, se incluyen en la mayoría de dispositivos móviles, pero su uso depende, en muy alto grado, de la aplicación y del tipo de actividad a reconocer [96]. Por ejemplo, para conocer el grado de movilidad de una persona, el lugar más recomendado para usar el dispositivo es en la cintura (al cinturón) [193]. En cambio, para determinar el movimiento del antebrazo, el lugar más adecuado es en la muñeca (pulsera) [104].

- **Localización**, básicamente viene dado por el Sistema de Posicionamiento Global (Global Positioning System, GPS por sus siglas en inglés) y trata de establecer la ubicación y todos tipos de servicios basados en localización [140]. La ubicación por GPS de una persona, es esencial para establecer un contexto para muchas aplicaciones [140]. Por ejemplo, si la persona camina por la calle, una aplicación podría ofrecerle lugares para tomar un café [28]. Existen servicios en la WEB, que permiten el uso del GPS del usuario, pero aún esta pendiente el tema de la privacidad, pues en muchos casos, el usuario es quien no desea ser rastreado.
- **Señales fisiológicas**, son en realidad señales cuya adquisición permite obtener información sobre la fuente que la generó [148]. También se las conoce como bioseñales, y las fuentes pueden ser los diferentes sistemas fisiológicos del organismo. Son parte inherente del usuario y captan datos tales como: pulso, ritmo cardíaco, temperatura corporal, ritmo respiratorio, etc. Este tipo de señales involucran al individuo en forma directa, pues las emite su sistema nervioso central, el cual trata de adaptarse a un contexto o responde a una necesidad. Por ejemplo, si el pulso es acelerado y la respiración es rápida podemos suponer que la persona esta pasando por momento de estrés. Por el contrario, si la actividad muscular es mínima y su actividad neurotransmisora estable, se puede afirmar que aquella persona esta muy relajada. Pueden existir señales Electrocardiográfica (ECG), Electroencefalográficas (EEG), Electromiográficas (EMG), entre otras. Esta últimas, son señales que afectan a un músculo unido al esqueleto y, básicamente se encargan, de ordenar la contracción o extensión de los músculos esqueléticos distribuidos en todo el cuerpo. Esta característica, permite que sensores basados en EMG hayan ganado especial interés, para medir y diagnosticar el comportamiento muscular por sus componentes de amplitud y frecuencia. Sin embargo, también son usadas en el reconocimiento gestual por sus características asociadas al movimiento muscular de manos y dedos. En este sentido, los resultados han sido alentadores, por ejemplo al aplicarlas al control de prótesis en personas con extremidades amputadas [3, 155] y en brazos robóticos [170, 110].

Para el reconocimiento de las las actividades humanas usando sensores wearables, es necesario que los datos en bruto (conocidos como raw) sean procesados [96]. El procesamiento de datos involucra la extracción de características de acuerdo al tipo de señal analizada. Luego, el modelo de reconocimiento, se construye a

partir del conjunto de instancias de características mediante técnicas de Machine Learning. Una vez encontrado el modelo adecuado (patrón), se lo evalúa usando instancias de prueba, obteniendo como resultado una predicción de la actividad realizada. Justamente en este punto, los algoritmos pueden ser evaluados para lograr una óptima clasificación y posterior implementación usando técnicas de Machine Learning [16].

Los sensores wearables, como hemos visto han “invadido.^{en} muchos sentidos el ambiente del usuario, al punto que se los ofrece integrados a otros de dispositivos de uso generalizado. Un ejemplo muy a lugar de este tipo de dispositivos, es el llamado teléfono inteligente o Smartphone. El Smartphone, en la actualidad esta dotado de una gran capacidad de procesamiento, posee un pequeño tamaño y, por lo general, tiene un mediano que lo hace asequible a todos los bolsillos. Muchos de ellos, o casi todos, incorporan sensores con el fin de ofrecer mejores opciones, de uso y aprovechamiento, al usuario. Entre otras cosas, miden variables provenientes del usuario (por ejemplo, pulso, ubicación, movimiento). Todos estos dispositivos, incorporan además, software de mensajería, asistencia y control, que ayudan al usuario, ya no solo en comunicación, sino también, en otras tareas de asistencia personal. Cuentan con nuevas y cada vez mejores herramientas, y usan como medio de interacción la superficie táctil. Por todas las características, en capítulo siguientes realizamos un análisis comparativo para determinar las características de interacción más acertadas entre el gesto y la superficie táctil, usando sensores de bioseñales.

1.3.3. Reconocimiento basado en una superficie de contacto

El uso de pantallas táctiles (Touchscreen, en inglés) en la actualidad goza de gran popularidad. Su integración masiva a dispositivos electrónicos, ha sido planteada como una forma fácil y efectiva de interacción con el usuario. En este sentido, los dispositivos móviles han sido los más beneficiados, en especial los llamados teléfonos inteligentes (Smartphone) como se puede observar en la Figura 1.5. Pero, junto al smartphone, el fabricante agrega sistemas operativos y aplicaciones, que tratan de aprovechar al máximo esta ventaja de interacción. Sin embargo, según Page [132] presentan contradicciones en su diseño para ser considerados inclusivos y ergonómicos. Su estudio concluye que si bien se adaptan a las necesidades de jóvenes y adultos, no se ha considerado el hecho de los estilos de uso. Pues, una persona joven usa las dos manos para escribir, mientras una persona adulta, sujeta con una mano el dispositivo y usa la otra para interactuar. Este es quizá, el problema principal que podría resolver el gesto y que nos ocupará en capítulos

siguientes.

Continuando, podemos decir, que otros trabajos de investigación la consideran ideal, a la interacción usando pantallas táctiles, por sus características simples, dan buena experiencia de usuario y son fáciles para navegar [10]. Otros mantienen, que no cumplen las expectativas de los usuarios y que en ningún momento llegan a ser tan cómodas como un teclado [139]. Mientras en contraste, otros presentan estudios que demuestran la gran versatilidad de la pantalla táctil frente al teclado [165].

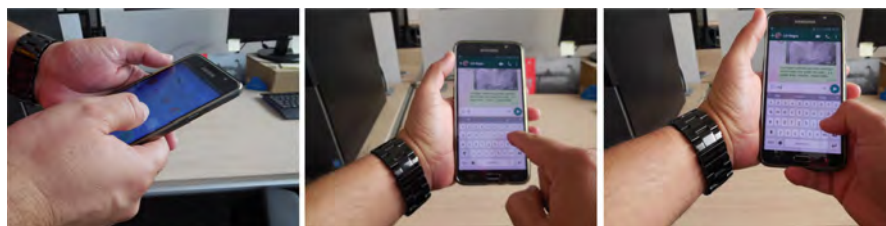


Figura 1.5: Usuario interactuando con un Smartphone.

En general, podemos decir que la pantalla táctil ha avanzado mucho en relación a las propuestas iniciales y junto a ella el software que las usan y aplican. Inconvenientes de tamaño, resolución, materiales y durabilidad han sido contemplados para su mejora. También podemos decir, que en la actualidad, la pantalla táctil, es una opción válida en la interacción, pero quedan algunos aspectos por resolver tales como: tamaño, flexibilidad, y adaptabilidad al usuario. El fabricante Samsung, es quien mejor a respondido en el desafío del desarrollo tecnológico para dispositivos móviles.

La pantalla táctil, permite realizar gestos como: arrastrar, toque, agrandar, achicar entre otros. Su ventaja, aunque con detractores, su alta exactitud. Su gran desventaja, a nivel gestual, es que requiere que el usuario toque un área específica (muy pequeña en ocasiones) para percibir un dato o una acción. Entonces, necesariamente debe contar con la ayuda de una aplicación que implemente a su vez, algún tipo de Interface Gráfica de Usuario (Graphical User Interface, GUI), para lograr una correcta interacción. Pero, por lo dicho por Page [132], con características de diseño y ergonomía que aún dificultan su uso.

Diseño

El diseño es un punto crítico en las Touchscreen y, su enfoque inclusivo, es uno de los desafíos más importantes. La idea, en este sentido, es lograr que todas las personas, indiferentemente de su edad o discapacidad, puedan hacer uso de esta

forma de interfaz. Esto ha inquietado a muchos investigadores, al punto de llegar a plantear la filosofía de “un diseño para todos” [125]. Esta propuesta plantea, que todos los productos convencionales atenderían a las necesidades y capacidades de todos los usuarios. Esto es sin duda, es uno de los puntos que más impulsan la investigación de nuevas y variadas formas de interacción, especialmente las interfaces naturales de usuario de tipo gestual.

Ergonomía

La ergonomía, es un punto neurálgico y más controversial dentro del diseño de las pantallas táctiles. El tamaño del dispositivo es quizá el problema más relevante en los dispositivos con Touchscreen. En ocasiones un diseño extremo, usando las dos manos, ha sido planteado por muchos fabricantes en sus dispositivos móviles, especialmente tabletas. En Smartphones, pocos se han aventurado por este camino y si lo han hecho, han creado modelos imposibles de manejar con una sola mano, aunque cumplen con características de comunicación. En este sentido, algunos estudios plantean que a mayor tamaño de las manos, menor será el esfuerzo [105]. Pero, en cambio otras investigaciones exponen que un usuario con manos más pequeñas tardará más en mantener la estabilidad del dispositivo. Todo esto, parece indicar que el debate continuará por más tiempo en este sentido. Inclusive, el tamaño del botón continúa en debate en relación al tamaño de los dedos. Destacan factores adicionales como edad [133] y grado de instrucción del usuario [95]. Nuevas alternativas, han sido presentadas las cuales mejoran la flexibilidad y dotan a la pantalla táctil de nuevas formas y diseños, creando Touchscreen flexibles [88]. Por lo expuesto, este es un campo muy interesante en la exploración en la interacción, y quizá la alternativa más prominente, es el uso de las manos sin tocar superficie alguna cuyo estudio proponemos en este tesis.

1.4. Procesamiento de la información

Quizá el punto más crítico, en el reconocimiento gestual, es el procesamiento de los datos que se obtienen de los dispositivos de captura ya sean ópticos, por sensores o por superficies de contacto. Las técnicas empleadas son muy diversas, y, tratan de ajustarse al dispositivo empleado y sus características. Sin embargo, más allá del dispositivo usado, esta justamente el hecho, de poder lograr un algoritmo que garantice el reconocimiento, el cual esta basado en muchos casos, en patrones ya definidos. Para lograr este objetivo, se han acudido a técnicas estadísticas y, dentro

de éstas a su más fino enfoque, el Aprendizaje de Máquina (Machine Learning, ML en inglés) o aprendizaje automático.

Las técnicas de ML, tienen su principal aplicación en la Minería de Datos (Data Mining, DM en inglés). Mientras, DM busca los patrones contenidos en los datos, ML trata de reproducir esos patrones y hacer predicciones basados en ellos. La primera explora los datos, mientras que la segunda, se centra en lograr predicciones basados en ellos. ML, trata de ayudar a las personas a resolver problemas, haciendo un análisis a los datos y tratando de establecer relaciones entre múltiples características [91]. Se dice entonces, que el aprendizaje de la máquina puede aplicarse con éxito a estos problemas, mejorando la eficiencia, tanto de máquinas como de sistemas. Cada instancia, de un conjunto de datos, se representa usando el mismo conjunto de características (continuas, categóricas o binarias). Desde este punto de vista, el aprendizaje depende de como se den las instancias, y pueden ser: el aprendizaje supervisado, no supervisado y semisupervisado. Todos ellos buscan lograr la predicción. A continuación describiremos brevemente cada uno de ellos.

1.4.1. Algoritmos de aprendizaje supervisado

Los algoritmos de aprendizaje supervisado, son aquellos en los cuales los datos o muestras son comparados con las etiquetas o clases. Busca patrones en las etiquetas y puede usar cualquier información que sea relevante. Una vez encontrado el mejor patrón, el algoritmo puede hacer predicciones sobre los datos procesados. En otras palabras, si las instancias están dadas usando etiquetas conocidas (que corresponden con las salidas correctas) entonces el aprendizaje es supervisado [91].

1.4.2. Algoritmos de aprendizaje no supervisado

Los algoritmos de aprendizaje no supervisado, son aquellos en los cuales no se etiquetan los datos y estos son tratados como un conjunto de variables aleatorias. En su lugar, el objetivo es organizar los datos de alguna manera o describir su estructura. En muchos casos, se utiliza la técnica de agruparlos en Grupos (Clusters) ⁵, la cual pretende organizar una colección de datos en grupos, de manera que los elementos dentro de un grupo sean lo más similares posible entre si, que los elementos de otros grupos o clousters [59]. El agrupamiento se realiza normalmente cuando no hay información disponible sobre la pertenencia de elementos de datos a clases predefinidas.

⁵El término Clustering (análisis de cluster), se aplica a la agrupación de datos en aprendizaje automático [194].

1.4.3. Algoritmos de aprendizaje semi-supervisado

Los algoritmos de aprendizaje semi-supervisado, son aquellos en los cuales se combina datos etiquetados y no etiquetados. Normalmente existe una pequeña cantidad de datos etiquetados junto a una gran cantidad de datos no etiquetados. Se lo considera parte del aprendizaje supervisado y no supervisado. En este algoritmo, además de la información de similitud utilizada por el agrupamiento no supervisado, en muchos casos se dispone de una pequeña cantidad de conocimientos sobre las restricciones entre pares (obligar must-link o no-enlazar cannot-link) entre elementos de datos o etiquetas de clase para algunos elementos [59]. Es útil cuando la adquisición de datos no etiquetados se impone al costo de mantener un agente humano capacitado, etiquetando los ejemplos de entrenamiento [24].

En la presente investigación, haremos uso de varios conceptos asociados a ML y que serán profundizados y explicados adecuadamente en forma oportuna.

A continuación, en la Tabla 1.4 tratamos de mostrar algunas de las ventajas de técnicas de ML más usadas en aprendizaje supervisado. Varias de ellas, ha sido usadas en problemas de clasificación y reconocimiento gestual, problema que trata esta investigación.

1.4.4. Conceptos de Aprendizaje de máquina (Machine Learning)

Bajo el concepto de Aprendizaje de Máquina (Machine Learning, ML en inglés), se engloban un conjunto de técnicas y algoritmos que permiten extraer información de los datos. Los sistemas de aprendizaje, tratan de encontrar una dependencia en los datos, utilizando un número (a veces pequeño) de observaciones [134]. Dicha dependencia viene generalmente representada por una función, que resume en cierto sentido, el conocimiento implícito que existe en las muestras tomadas. Así, por ejemplo en un problema de clasificación, la función puede representar la frontera de separación entre dos clases. En cambio, en un problema de regresión, la función proporcionará el mejor ajuste a los datos.

Según Pajares et al. [134], la construcción de un sistema de aprendizaje requiere de tres elementos, que se muestran en la Figura 1.6 y que se detallan a continuación.

Elemento fuente de datos

Este elemento, constituye los datos propiamente dichos y a los cuales se quiere caracterizar. Los datos, con los cuales trabajará el sistema de aprendizaje, deben satisfacer dos condiciones que, en la literatura especializada se conoce como i.i.d.

Tabla 1.4: Comparación entre diferentes técnicas de ML aplicadas a predicción

Algoritmo	Fundamento	Parámetros	Ventajas	Limitaciones
K-means [1, 82]	Determina k puntos llamados centros como la suma de las distancias de todos los puntos de datos respecto sus respectivos centros de cluster	Localización del cluster central	Rápido computacionalmente hablando. Produce clusters compactos.	El valor de K es difícil de fijar para el número de clusters. Particiones iniciales diferentes resulta en diferentes cluster finales.
K-nearest neighbor [196, 82, 88, 89, 27, 57, 82, 169]	Localiza las instancias k más cercanas a la instancia de consulta y determina su clase identificando la etiqueta de clase más frecuente (cite:kotsiantis)	Clase de vecino más cercano	Fácil de implementar. Baja complejidad. La adecuada elección de características da buenos resultados (routary)	Sensible a atributos arbitrarios. El valor de k afecta al rendimiento del algoritmo.
Árboles de Decisión [112, 43, 62, 70, 119]	Predice el valor de una variable de destino en función de diversas variables de entrada. La variable puede tomar un conjunto finito de valores (clasificación). Si la variable toma valores continuos es un árbol de regresión. Describe los datos no las decisiones.	Cada nodo interno (no hoja) esta etiquetado con una función de entrada. Valores de las características. Distribución de probabilidad sobre las clases.	Relativamente rápido aprendizaje con respecto al número de atributos de las instancias. Construye un árbol de decisión a partir de los datos de entrenamiento.	Solo es recomendable cuando tenemos un pequeño número de acciones y no todas las combinaciones son posibles. Mala generalización a partir de los datos de entrenamiento (sobreajuste). No se garantiza un árbol de decisión totalmente óptimo (NP-Completo).
Support Vector Machine [102, 164, 26, 98, 27, 57, 82, 169]	Establece un margen de separación entre hiperplanos que separan dos clases minimizando la norma cuadrada del hiperplano de separación. Puede llevarse a espacios de alta dimensionalidad para separación de muchas clases. (cite:kotsiantis)	Características del espacio de alta dimensionalidad. Dispone de varios núcleos o kernels.	Alta exactitud en predicción. Robusto para errores de entrenamiento. Rápida evaluación de la función objetivo aprendida.	Tiempo de entrenamiento alto, compleja función de aprendizaje (pesos). La selección del kernel para lograr clasificación multiclase es crítica.
Hidden Markov Model [204, 42, 156, 146, 84]	Generalización de una cadena de Markov sin restricciones. Conjunto de estados, las transiciones representan el conjunto de posibles posiciones de la mano.	Características de la señal de entrada	Fácilmente extensible. El adecuado en el algoritmo interno es posible y fácil de entender.	En necesario configurar un gran número de parámetros. Los datos de entrenamiento son grandes. (Routaray)
AdaBoost [108, 26]	Encontrar un clasificador fuerte a partir de una combinación lineal de clasificadores débiles simples. Propone entrenar una serie de clasificadores débiles de manera iterativa, de modo que cada nuevo clasificador se enfoque en los datos clasificados erróneamente por su predecessor, para lograr mejores resultados.	Número de iteraciones.	Se adapta y logra mejores resultados. Puede ser multiclase. Un clasificador débil no requiere conocimiento previo. Fácil, versátil y rápido de implementar.	Clasificadores débiles complejos pueden llevar a overfitting, demasiado débiles pueden producir bajo margen y overfitting. Muy sensible a ruido para lograr implementaciones en el mundo real (long,2010).
Redes Neuronales [175, 204, 57]				

Los Árboles de Decisión por lo general acompañan a otros clasificadores. El C4.5 es algoritmo de decisión más común de los árboles de Decisión.

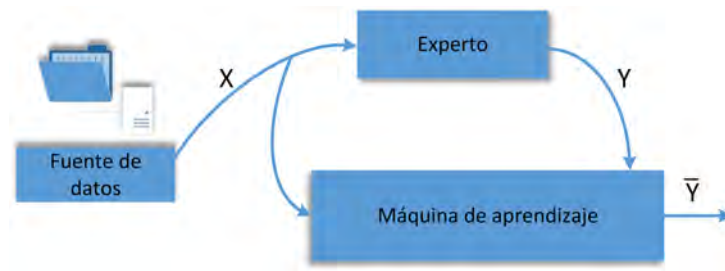


Figura 1.6: Elementos de un sistema de aprendizaje [134].

(“independent and identically distributed”, siglas en inglés) y son:

- **Los datos producidos por la fuente están idénticamente distribuidos**, lo cual significa que, todos los datos obedecen a una misma distribución de probabilidad, sea cual sea, pero única. Además, esta distribución es desconocida.
- **Los datos producidos por la fuente son independientes entre sí**, es decir, la generación de un dato, ni esta vinculada a datos generados con anterioridad, ni condiciona los datos futuros.

Normalmente, cada dato u observación, viene determinado por un vector x de nh componentes, donde cada componente refleja una característica o propiedad de los objetos bajo estudio. A cada vector, es el representante de un dato y se le denomina *vector de características*. Lo relacionado a la selección de características, se trata más en detalle en el Capítulo 3, Sección 3.4.2.

Elemento Experto

Por lo general es el ser humano, el cual a la vista de un dato generado, asigna un valor y a ese dato. El elemento experto deberá elegir entre la naturaleza de los datos y el algoritmo idóneo para dichos datos, pues no todos los algoritmos se acoplan o ajustan adecuadamente a todos los datos. En ocasiones es necesario un análisis estadísticos para tratar de determinar la forma de distribución de los datos y la mejor opción de clasificación.

Elemento Máquina de Aprendizaje

Es la máquina de aprendizaje, es el sistema de cómputo sobre la cual se entrenará o aprenderá el algoritmo. La máquina de aprendizaje, durante el proceso,

recibe un conjunto de pares (x,y) , que constituyen los datos de entrenamiento. Con los datos recibidos y con los valores asignados por el experto, la máquina de aprendizaje construye un modelo. El objetivo final, es prescindir del experto y, que ante la llegada de un dato x , el modelo construido durante el entrenamiento responda con un valor \bar{y} que esté cercano al valor real y .

Clasificación y Regresión

Como ya mencionamos anteriormente, el aprendizaje es una técnica para deducir una función a partir de los datos de entrenamiento. Por lo general son vectores, uno para los datos y otro para los resultados. Un sistema de clasificación predice una categoría, mientras que uno de regresión predice un número [194].

Clasificación y regresión son parte de los conceptos que hemos analizado y corresponden al aprendizaje supervisado. En la mayor parte de estudios asociados al gesto, se trata de etiquetar al gesto, usando para ello etiquetas de clase, convirtiéndose entonces en un problema de clasificación.

Tratando de lograr un mejor entendimiento en este sentido, proponemos este ejemplo, planteado por Moo et al. [121] quienes en su estudio aplicando SVM a un problema de clasificación documental exponen: “Se requiere agrupar un conjunto de 1.000 artículos tomando como referencia el tema que tratan, en 5 grupos posibles: arte, ciencia, tecnología, política y negocios. Los grupos pueden estar definidos previamente, o bien, pueden ser generados al momento mismo de agrupar los documentos, lo que haría nuestro trabajo aún más complicado. En el primer caso se habla de una tarea de **clasificación supervisada**, mientras que en el segundo se habla de **clasificación no supervisada** [121]. De alguna manera, en este ejemplo, se deberá encontrar un patrón de aquello que se pretende categorizar, para el caso del ejemplo, los artículos. Dicho patrón deberá responder principalmente a tres puntos [121]: **Primero**, ¿Qué hace que un artículo sobre avances en la medicina deba ser asignado al grupo de artículos sobre ciencia y no al de tecnología?. **Segundo**, se requiere encontrar la mejor forma de representar los elementos característicos de cada artículo, para no usar como entrada del algoritmo clasificador todo el texto que aquellos contienen, sino únicamente un vector x conteniendo aquellas características mínimas necesarias para realizar una clasificación efectiva. Y, **Tercero**, el tiempo que emplean los algoritmos para encontrar una respuesta, lo cual es en la mayoría de los casos, es largo y requiere de mucho procesamiento, inclusive procesamiento paralelo”.

Si este ejemplo, lo trasladamos a nuestro objeto de estudio, el gesto, encontra-

remos que al momento de llevar a cabo la tarea de clasificación gestual se deben determinar: el gesto, sus características principales y, finalmente, implementar (optimizar si es posible) el algoritmo en base a sus parámetros, para encontrar una respuesta adecuada con el menor costo computacional posible. Es aquí justamente, donde entran en juego los diferentes algoritmos de clasificación que ayudarán a resolver este problema.

En un algoritmo de ML se busca siempre obtener la mejor exactitud, pero en muchos casos el equilibrio con la precisión es lo más recomendable (dualidad precisión-exactitud). Estos parámetros pueden ser evaluados tomando en consideración la denominada Matriz de Confusión, la cual se estudia en la siguiente sección.

Los conceptos más comúnmente usados en Machine Learning han sido agrupados en el Apéndice B de esta tesis.

1.4.5. Flujo de la información

La Figura 1.7 trata de explicar gráficamente el flujo de la información al aplicar un algoritmo supervisado.

El primer paso, es la definición del problema a tratar con el algoritmo de ML. Por lo general, los datos están sujetos, como se había mencionado anteriormente, a la forma en que el dispositivo capte la información. En dispositivos como cámaras, la información puede estar contenido en frames ⁶ de datos que a su vez pueden contener información relacionada con dicho frame como: resolución, filtro aplicado, propiedades de color, etc. En sensores como EMG, la información puede ser una señal que codificada nos entrega un conjunto de dígitos, distribuidos entre valores máximos y mínimos, entre rangos positivos y negativos, etc. De ese gran conjunto de datos posible, es necesario poder discriminar, que datos son los más adecuados para lograr el objetivo de clasificación propuesto. Dicha discriminación nos podría ayudar a tomar en cuenta aquellos campos (atributos, características) que nos dan más información .

En muchos casos, se acude al llamado método de “fuerza bruta”, uno de los más simples, que significa medir todo lo disponible con la esperanza de que las características correctas (que aporten más información) puedan ser separadas. Sin embargo, debemos indicar que “fuerza bruta” es un método que no permite hacer una inducción de la información, pues puede contener ruido y valores faltantes de características, requiriendo un preprocesamiento significativo como lo muestra Buczak & Guven [22], en donde se analiza la complejidad de ML/DM en seguridad

⁶El frame puede considerarse una una instantánea de un objeto en un intervalo de tiempo.



Figura 1.7: Proceso usado en algoritmos supervisados de ML

cibernética.

El paso de **preparación de datos y preprocesado** tiene por objeto tratar de llenar ciertos “vacíos” que pueden producirse en los datos. Estos “vacíos” en muchos casos representan ruido o datos faltantes, que deberán ser minimizados en el primer caso y completados, en el segundo. Se realiza una selección de instancias para minimizar el ruido y también para afrontar la imposibilidad de aprendizaje a partir de datos muy grandes. Esto se considera un problema de optimización que intenta mantener la calidad de la minería mientras se minimiza el tamaño de la muestra [91](Liu y Motoda, 2001). El hecho, de que muchas características de-

penden unas de otras, influye en la exactitud de los algoritmos de clasificación de ML supervisados. Este problema se puede resolver según Markovitch & Rosenstein [113] **construyendo nuevas características del conjunto de características iniciales básicas**. Kotsiantis [91] indica que “*Esta técnica se denomina construcción/transformación de característica. Estas características generadas recientemente pueden conducir a la creación de clasificadores más concisos y precisos. Además, el descubrimiento de rasgos significativos contribuye a una mejor comprensión del clasificador producido, ya una mejor comprensión del concepto aprendido*”.

La elección del **algoritmo de aprendizaje** es un paso crítico. Varios investigadores aconsejan usar un algoritmo que mejor se acople a los datos, especialmente si son continuos o discretos. Mientras, otros proponen que se prueba varios algoritmos con los mismos datos y observar aquel que logre la mejor exactitud de predicción.

La evaluación de un algoritmo se basa en la exactitud de la predicción (el porcentaje de predicción correcta dividido por el número total de predicciones). Existe al menos tres técnicas de evaluación de un algoritmo, que las mencionamos a continuación:

1. **Dividir dos tercios del conjunto de entrenamiento** (dataset) para el entrenamiento propiamente dicho y otro tercio para estimar su desempeño. Por lo general se aplica el 70 % para entrenamiento y el 30 % para verificación.
2. **Validación cruzada** (cross-validation), donde el conjunto de entrenamiento se divide en conjuntos mutuamente excluyentes y de igual tamaño y, para cada subconjunto, el clasificador es entrenado sobre la unión de todos los otros subconjuntos.
3. **Validación de la exclusión** (Leave-one-out validation), es un caso especial de la validación cruzada. Todos los subconjuntos de prueba constan de una única instancia. Es mucho más costosa computacionalmente hablando, pero útil cuando se requiere la estimación más precisa de la tasa de error del clasificador.

Si, la evaluación de la tasa de error no es satisfactoria, debemos volver a una etapa previa del proceso de ML supervisado (como se muestra en la Figura 1.7). En este caso, lo más adecuado será verificar si: se están utilizando o no características realmente relevantes para el problema; se necesita un conjunto de entrenamiento más grande; la dimensión del problema es demasiado alta; el algoritmo seleccionado es inadecuado; o, si se requiera un ajuste (tuning) de parámetros, entre otros. Mayor información a este respecto se la puede encontrar en [194].

La **matriz de confusión** asociada a cada algoritmo, luego de cada test de prueba, puede ser usada para medir la exactitud de la predicción (ecuación 1.1), precisión (ecuación 1.2), recall (ecuación 1.3), falsa tasa de aceptación (FAR) (ecuación 1.4) y falsa tasa de rechazos (FRR) (ecuación 1.5). La figura 1.8 muestra los componentes de la matriz de confusión. Las operaciones que se describen a continuación y las ecuaciones siguientes, se realizan sobre la matriz de confusión y permiten mejorar el entrenamiento, además de lograr un mejor ajuste de los parámetros.

En esta investigación, el rendimiento del modelo fue medido usando el valor alcanzado por la exactitud. Esta forma de determinar el rendimiento, es el camino más comúnmente usado para los conjuntos de entrenamiento, pero debe ser obtenido a partir de la matriz de confusión. Además, entrega el modelo de clasificación creado, y, no debe ser usada como base para crearlo [106]. Los resultados obtenidos al ejecutar el modelo, se muestran en la matriz de confusión. Así pues, si bien la exactitud es la métrica más ampliamente extendida entre las medidas de rendimiento del Aprendizaje Automático, no es la única medida de rendimiento de un modelo. Pudiendo evaluarse al modelo, tanto por el número de casos negativos (TNR), cuanto por el número de casos positivos (TPR).

		Predecida	
		0	1
Actual	0	Verdaderos negativos (VN)	Falsos positivos (FP)
	1	Falsos negativos (FN)	Verdaderos positivos (VP)

Figura 1.8: Matriz de confusión de ML

- Positivos, el total de instancias positivas.
- Negativos, el total de instancias negativas.
- Verdaderos negativos, el total de instancias que siendo negativas han sido clasificadas como negativas.
- Verdaderos positivos, el total de instancias que siendo positivas han sido clasificadas como positivas.
- Falsos positivos, el total de instancias que siendo negativas han sido clasificadas como positivas.

- Falsos negativos, el total de instancias que siendo positivas han sido clasificadas como negativas.

$$Exactitud = \frac{VP + VN}{P + N} \quad (1.1)$$

$$Precision = \frac{VP}{VP + FP} \quad (1.2)$$

$$Recall = \frac{VP}{VP + FN} \quad (1.3)$$

$$FAR = \frac{FP}{VN + FP} \quad (1.4)$$

$$FRR = \frac{FN}{P} \quad (1.5)$$

1.5. Resumen

En este capítulo hemos realizado un enfoque sobre la riqueza gestual que el hombre encierra y que ha servido como medio de expresión y comunicación, incluso mucho antes de que existiese el lenguaje verbal. Hemos realizado un estudio del gesto, sus características y su clasificación. El gesto ha sido clasificado de acuerdo al uso en el lenguaje (Taxonomía) y de acuerdo a su naturaleza (forma-tipo). Hemos realizado un estudio de la complejidad del captura y técnicas de reconocimiento gestual. También, hemos logrado identificar las formas de captura del gesto y, hemos definido cada una de ellas. Hemos realizado un recorrido sobre los tipos de sensores, su clasificación y potencial uso, llegando a definir los llamados sensores wearables. Se ha expuesto el uso de pantallas táctiles en la interacción logrando establecer que el dispositivo móvil es el dispositivo que mayor ventaja ha sacado de su aplicación. Se ha revisado las técnicas de procesamiento de información, que de acuerdo al dispositivo, se producen para la captura del gesto en base a los datos generados. Nos hemos adentrado en el aprendizaje de máquina y hemos analizado varios algoritmos de ML utilizados para la clasificación y reconocimiento gestual.

Por último, hemos propuesto el flujo de información que deberá tenerse en cuenta al trabajar con técnicas de ML aplicadas a reconocimiento gestual.

Ahora, debemos continuar adentrándonos en cada una de las diferentes formas de captura gestual y su integración a sistemas de reconocimiento y control. Nos resta el desafío de validar la interacción gestual y descubrir su real potencial, aplicándola al control de dispositivos e incorporando nueva teoría del lenguaje gestual a la interacción. Podremos evaluar al gesto libre de la mano del usuario con el competidor más directo cual es la pantalla táctil, probar su independencia en relación a la posición corporal, creando software, proponiendo arquitecturas y lenguajes. Todo esto, se muestra como un nuevo y novedoso perfil de comunicación. Dicho perfil, podrá ser usado en muchos y variados ámbitos, como por ejemplo la supervisión, control y operación de dispositivos de naturaleza heterogénea (domésticos, robóticos, industriales, etc.).

Capítulo 2

Análisis de la Interfaz gestual usando medios ópticos (tracking)

El uso de medios ópticos o basados en visión, para lograr la interpretación de los gestos, es quizá uno de los medios más extendidos para tratar de lograr la facilidad y naturalidad deseada en la HCI. El término seguimiento o tracking, hace referencia al hecho de poder rastrear un objeto en base al control de parámetros (posición, orientación, zoom) con la ayuda de algún algoritmo diseñado para este propósito, en función de las características extraídas de la imagen. Esto es posible gracias a la gran capacidad de procesamiento de los ordenadores y a la disponibilidad de cámaras de vídeo de alta calidad y bajo costo. Ahora se habla de formas automatizadas de vídeo, que implementan algoritmos de seguimiento de objetos. El tracking para ser efectivo, debe mantener un determinado objeto, en movimiento, dentro del campo de visión. Por ello, debe enfrentar varios y serios problemas, que pueden surgir debido al movimiento rápido, al cambio de los patrones de apariencia, tanto del objeto como de la escena, las estructuras de objetos no rígidos, la oclusión de un objeto a otro y de un objeto a una escena, cambios de luz y color, y además, el movimiento de la cámara [199, 120, 179]. El seguimiento o tracking, se realiza normalmente en el contexto de aplicaciones de alto nivel, que requieren conocer la ubicación y/o la forma del objeto en cada frame. En muchos casos, se hacen suposiciones para restringir el problema de seguimiento en el contexto de una aplicación particular.

El reconocimiento del gesto basado en medios ópticos, proporciona libertad de

movimiento del usuario, al no requerir aditamento alguno, dentro de su campo de visión. Además, logra su captura y reconocimiento, pero solo luego de superar varios de los problemas descritos anteriormente. A pesar de dichos problemas, los medios ópticos han sido ampliamente usados en el reconocimiento gestual, logrando un mayor impacto que aquellos que acuden al uso de guantes de datos, pues a estos últimos, se los considera demasiado intrusivos. Además, los guantes requieren una calibración frecuente, restringen la expresión natural del gesto del usuario y lo atan, a procesos de calibración y de alcance de transmisión de datos (rango de alcance de cables, Bluetooth, WiFi). Pero, se espera que en los próximos años, estas restricciones sean superadas y además, los guantes incorporen nuevos y mejores sensores, como las Unidades de Medición Inercial (Inertial Measurement Unit, IMU) o sensores de presión de aire. Todo ello, para alcanzar una mejor experiencia de uso en el usuario.

En este capítulo, analizamos el gesto aplicado a control sobre una interfaz gestual basada en el controlador Leap Motion. Dicho dispositivo, combina tecnología óptica e infrarroja en el reconocimiento gestual por tracking. Para probar este dispositivo, hemos diseñado y aplicado completo análisis, usando una amplia biblioteca gestual compuesta tanto por gestos dinámicos como estáticos. En primer lugar, se realiza un estudio minucioso del dispositivo y sus características de reconocimiento. Aquí, se plantean algunas definiciones resultantes de la experimentación y la observación. Se propone un conjunto de métricas aplicables a un dispositivo de esas características y sobre las cuales, se han evaluado los gestos del catálogo propuesto. Esto ha permitido, determinar su comportamiento y su validez en control reactivo. Finalmente, se establece un caso experimental, aplicado sobre un sistema robótico LEGO EV3, el cual valida los hallazgos encontrados.

2.1. Fundamentos de detección y seguimiento

La interacción usando medios ópticos, puede ser considerado un campo de investigación interdisciplinario, el cual involucra a la visión por computadora, procesamiento de imagen, machine learning, bio-informática y psicología [122]. Por esto, se definen tres pasos importantes en tracking que son: la detección de los objetos; el seguimiento de dichos objetos, de un fotograma (frame) a otro; y, el análisis de las posibles pistas dejadas por los objetos para conocer su comportamiento. Dicho de forma más simple, el tracking trata de estimar la trayectoria de un objeto en el plano de la imagen, mientras que se mueve alrededor de una escena. Por lo gene-

ral, el algoritmo trata de proporcionar información centrada en características del objeto, tales como: orientación, área o forma del objeto [199].

El interés por aplicar el gesto a tareas comunes, ha permitido el desarrollo de nuevos dispositivos tecnológicos, que incorporan las definiciones anteriores de reconocimiento, al gesto; pasando de dispositivos que debe usar el usuario (como guantes) a dispositivos que no requieren que el usuario los lleva consigo (medios ópticos) y, que a la vez fusionan hardware y software en un solo aparato [162, 37, 154]. El seguimiento de objetos es un problema complejo y sujeto a múltiples dificultades como: movimiento abrupto de objetos, cambio de patrones de apariencia (objeto y escena), la estructuras de objetos no rígidos, movimiento de la cámara, pérdida de información al pasar de 3D a 2D, ruido, complejidad del objeto, forma y grados de libertad del objeto, oclusiones en forma parcial o total, cambios en la iluminación y requerimientos de procesamiento en tiempo real [199]. Estos problemas, han sido enfrentados de varias maneras tales como: simplificar el movimiento imponiendo restricciones (movimiento, apariencia), suponiendo la forma del movimiento (sin cambios bruscos de velocidad o aceleración), asumiendo información previa (numero, tamaño, apariencia, forma) para simplificar el problema.

En un escenario de tracking, un objeto puede definirse como aquello que es de interés en un dominio específico [199]. Es decir, se toma en consideración su forma y apariencia, por ejemplo, un avión en el cielo, los peces en una pecera, etc. Por su forma pueden ser: por puntos, forma geométrica, silueta y contorno, modelos articulados y modelos esqueléticos. La Figura 2.1 muestra la clasificación de tracking por su forma y su apariencia.

Por su forma:

- Puntos. El objeto es representado por un punto, que es un centroide, o por un conjunto de puntos. Figura 2.1.a y b.
- Forma geométrica. La forma del objeto es representada por rectángulo, elipse, círculo, etc. El movimiento del objeto por cada representación es usualmente representada por traslación, afín o proyectiva. Se puede usar tanto con objetos rígidos como no rígidos. Figura 2.1.c y d, .
- Contorno y silueta del objeto. El contorno define el límite de un objeto. La silueta y el contorno, son usadas para tracking, de formas no rígidas. Figura 2.1.g, h e i.
- Modelos articulados. Son las partes unidas por articulaciones. La relación entre las partes se rige por modelos de movimiento cinemático. Figura 2.1.e.

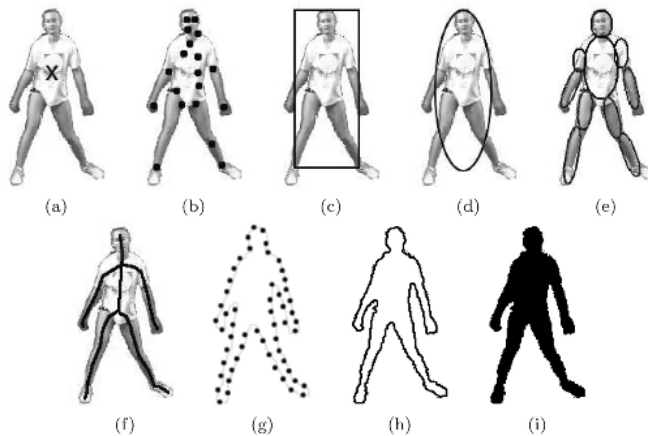


Figura 2.1: Reconocimiento de un objeto por su forma y apariencia usando tracking en [199].

- Modelos esqueléticos. Es una extracción del eje medial a la silueta del objeto. Puede usarse para modelar objetos articulados. Figura 2.1.f.

Por su apariencia pueden ser: densidades de probabilidad de la apariencia de un objeto, plantillas, modelos de apariencia activa y modelos de apariencia multivista. Las formas y las apariencias pueden combinarse para realizar el tracking.

- Densidad de probabilidad de apariencia, la cual pueden ser paramétrica (gausiana) o no paramétrica (Ventanas Parzen) e histogramas. Se pueden calcular a partir de las regiones de imagen especificadas por los modelos de forma (región interior de una elipse o su contorno). Figura 2.1.d.
- Plantillas. Se forman usando formas geométricas simples o siluetas. Solo son adecuadas para realizar tracking de objetos cuyas poses no varían considerablemente durante el proceso. Figura 2.1.g y h.
- Modelos de apariencia activa. Son aquellos que se generan modelando simultáneamente la forma y la apariencia del objeto. Por lo general requieren de un entrenamiento para que la forma y la apariencia aprendan del conjunto de muestras del componente principal. Figura 2.1.h.
- Modelos de apariencia multivista. Son aquellos que codifican diferentes vistas de un objeto. Figura 2.1.i.

2.1.1. Selección de características por tracking

Iniciaremos indicando que la selección de características es fundamental para el reconocimiento de gestos, pues el gesto, concretamente la mano puede variar en su forma, movimiento y textura. Las características en muchos casos depende del tipo de gesto a ser detectado, por ejemplo en gestos estáticos la extracción de algunas características geométricas como la punta de los dedos, las direcciones y contornos de las manos. Así también, existen otras características no geométricas como color, silueta y texturas, pero han sido muy poco eficaces en el reconocimiento. En todos ellos, toda la imagen es tomada como entrada.

2.1.2. Revisión científica

A continuación presentamos la Tabla 2.1 en la cual se resumen algunos trabajos científicos más relevantes, en los cuales se exponen el reconocimiento gestual logrado realizando tracking del gesto. En estos trabajos, se puede apreciar tanto la diversidad de los modelos y técnicas de segmentación usados, además del algoritmo de clasificación propuesto.

Brown et al [19], plantean el uso del reconocimiento del gesto usando un sensor óptico wearable, conocido como Leap Motion, junto a una unidad de medición inercial para combinar los beneficios del wearable y del seguimiento basado en medios ópticos. Su análisis le permitió desarrollar un soporte llamado LEIMU el cual logró dar mayor flexibilidad a su enfoque de reconocimiento gestual, al incorporar mayor movilidad en la mano sin tener que restringirse a estar directamente sobre el dispositivo. Con LEIMU, es el dispositivo el que acompañaba al usuario. En este estudio se usó un enfoque estadístico para la evaluación de los resultados y los usuarios que participaron en él, consideraron que LEIMU es una interfaz eficaz, además de ofrecer mejoras en la precisión del tracking que si se ubiese usado al wearable sobre una mesa.

Yuruk [200], trata el problema del reconocimiento desde un punto de vista basado en verificación de imágenes de la mano (derecha). Las imágenes usadas fueron capturadas por escáner y, en una etapa de preprocesamiento del algoritmo, las siluetas, de las imágenes de la mano, se registraron en una postura fija que involucraba traslación y rotación y, posteriormente, en forma separada de los dedos. Fueron evaluadas dos conjuntos de características a saber la distancia de Hausdorff de los contornos de las manos y características de los componentes independientes de las siluetas de las imágenes de las manos. Este método alcanzó buenos resultados demostrando que el reconocimiento manual es viable en ambientes de seguridad de

Tabla 2.1: Reconocimiento del gesto usando características de reconocimiento óptico

Primer autor	Tipo entrada	Tipo segmentación/modelo	Características	Clasificador	Us	Librería	D/E	ACC
Brown [19]	Leap Motion	Modelo 3D, esqueleto	Geometría	Propietario	4	SI (12 gestos)	D/E	-
Yuruk [200]	Cámara digital	Contorno y Normalización. Distancia de Hausdorff	Geometría	K-means	4	NO	E	98,21
Dinh [43]	Cámara web	Silneta de profundidad	NO	Bosques aleatorios (Random Forest)	5	SI (6 gestos)	D/E	98,50
Premaratne [145]	Wabe Controller	Segmentación de la piel, YCbCr, plantillas	NO	Lucas-Kanade, Momentos Invariantes	-	SI (8 gestos)	D/E	98,75
Ding [42]	Kinect	Modelos esqueleticos y articulados	Geometría	HMM-DTW-PCA	7	SI (10 gestos)	D/E	87,86
Sánchez-Nielsen [160]	Cámara Web	Plantillas, color de piel, Normalización	Geometría	Distancia de Hausdorff		SI (25 gestos)	E	90
Su [178]	Kinect	Esqueleto, articulado	Geometría	DTW-Fuzzy	4	SI (5 gestos)	D/E	80,28
Setiawan [167]	Cámara Web	Modelos de apariencia multivista	Geometría	NN Shape Fitting	1	SI (5 gestos)	E	84,53
Sarkar [163]	Leap Motion	Modelo 3D, esqueleto	Geometría	Propietario	1	SI (4 gestos)	D/E	-
Mehler [114]	Kinect	Enfoque de apariencia activa.	Geometría	DTW		SI (19 gestos)	D/E	87
Stergiopoulou [175]	Cámara digital	Segmentación de la piel, YCbCr, plantillas	NO	Self-Organized Gas, SGONG	1	SI (5 gestos)	E	89
Chen [26]	Cámara Web	Segmentación de la piel, YCbCr, plantillas	NO	SVM		SI (9 gestos)	E	95,88
Qian [146]	Kinect	Seguimiento por Camshift	NO	HMM	1	SI (4 gestos)	D/E	85
Itkharik [75]	Cámara Web	Silneta, Contorno, Envolverte Convexa	Geometría	Correlación de Harris	1	SI (10 gestos)	E	81,66
Hsieh [69]	Cámara Web	Formas geométricas, puntos, Modelos de apariencia multivista	Geometría	SVM	5	SI (6 gestos)	E/D	95,37
Ohm-Bar [130]	Cámara Web	Segmentación de la piel, YCbCr, plantillas	Geometría	SVN-HMM	1	SI (19 gestos)	E/D	92,8
Chuan [29]	Leap Motion	Modelo 3D, Esqueleto	Geometría	SVM, k-NN	2	SI (13 gestos)	E	79,83

D/E: Gestos dinámicos o estáticos, ACC: Exactitud reportada, Us: Número de usuarios.

control de acceso.

Dinh et al. [43], presentan novedoso sistema de interfaz gestual, basado en gestos de la mano, a través de un sensor de imágenes en profundidad para el control de aparatos en entornos domésticos inteligentes (TV, ventilador, iluminación, puertas, temperatura y volumen). Para lograrlo usan siluetas de profundidad de la mano además de mapas etiquetados de sus partes. Esta información es usada para entrenar un algoritmo de clasificación basada en Bosques Aleatorios (Random Forest, RF).

Premaratne et al. [145], presentan un sistema de control para dispositivos y equipos electrónicos basados en tecnología de la universidad de Wollongong, conocido como “Wave Controller”. Se basan en segmentación de la piel y extracción de regiones de la mano, aplicando el algoritmo de Lucas-Kanade y Momentos Invariantes (Moment Invariants) basado en plantillas y descriptores de Fourier para resaltar las ventajas y limitaciones del sistema propuesto.

Ding et al. [42], proponen una solución de tiempo real, basados en Kinect, para lograr que un robot humanoide pueda reproducir las acciones humanas. Este trabajo utiliza tres diferentes mecanismos de reconocimiento, DTW, modelos Ocultos de Markov (HMM) y análisis de componentes principales (PCA) para llevar a cabo la tarea de reconocimiento de los gestos. Cada gesto es reconocido y usado para controlar la acción del robot el cual imita el gesto humano realizado. En su trabajo usan modelos esqueléticos y articulados, propuestos por el SDK del dispositivo Kinect.

Sánchez-Nielsen et al. [160] proponen un sistema basado en cámara y en tiempo real, aplicado a entornos de interacción visual mediante el reconocimiento de gestos de manos, utilizando hardware de uso general y sensores de bajo costo. Su enfoque se basa en un proceso de segmentación temporal para obtener la imagen de la mano en varias formas, sobre diferentes fondos y condiciones de iluminación. La parte más importante del proceso de reconocimiento es una robusta comparación de formas, llevada a cabo usando el algoritmo de distancia de Hausdorff, el cual trabaja sobre mapas de borde (edge maps). Ellos hacen uso de una memoria visual, la cual permite al sistema manejar variaciones dentro de un gesto y acelerar el proceso de reconocimiento, a través del almacenamiento de diferentes variables relacionadas con cada gesto.

Su [178], cambia el enfoque y dedica su estudio a instalaciones de rehabilitación en un centro hospitalario. Proponen un sistema de rehabilitación domiciliar llamada KEHR, usando un sistema basado en Kinect. Ellos usan un algoritmo DTW

(Dynamic Time Warping) y lógica difusa (Fuzzy Logic) para identificación del gesto y movimiento gestual. Al igual que Ding et al. ellos usan modelos esqueléticos y articulados, propuestos por el SDK del dispositivo Kinect.

Setiawan [167], nuevamente nos cambia el enfoque de uso y explotación del reconocimiento visual y propone un sistema de reconocimiento de gestos manuales en un sistema embebido. Para lograrlo usa una tarjeta OK641OB. Esta placa tiene un procesador que trabaja a 532 MHz, que es relativamente alto para un procesador pequeño. El método de reconocimiento de gesto de la mano propuesto en este artículo se basa en el ajuste de forma de Redes Neuronales con algunas mejoras. Dichas mejoras la aleatoriedad de píxeles durante el paso de inicialización, la adición de varias neuronas en las iteraciones, la utilizando la tabla de búsqueda para la medición de la distancia y la simplificación de la detección de dedos. Estas modificaciones lograron un procesamiento más rápido ($O, 95$ s en el OK641OB) y una alta tasa de reconocimiento, tanto para imágenes fijas como captura en vivo, usando una cámara web. En su enfoque uso características de modelos de apariencia multivista y activa.

Sarkar et al. [163] proponen el control de un Dron Parrot $\text{\textcircled{R}}$ AR-DRONE 2.0 basado en gestos de la mano. Para lograrlo usan a al controlador Leap Motion. El AR-DRONE está conectado a la estación de tierra vía Wi-Fi y, en la misma estación, el Leap Motion vía el puerto del USB. El controlador Leap Motion reconoce los gestos de las manos y lo transmite a la estación terrestre. La estación terrestre utiliza dicho reconocimiento y lo transforma en comandos para la interacción con el AR-DRONE. En su enfoque, ellos usan gestos simples (estáticos) de las manos. Usan el SDK de desarrollo del Controlador para el reconocimiento gestual.

Mehler et al. [114], proponen una valoración teórica de los gestos de las manos, en el contexto de la creación de hipertextos relacionados con la imagen, siguiendo el ejemplo del sistema de información del museo llamado Wi-kinect, basado en Kinect. Con este fin, se diseña una implementación que logra la escritura gestual basada en esquemas de imagen (avisos). La escritura gestual se define como una especie de codificación en la que las proposiciones sólo se expresan por medio de gestos. En este sentido, se muestra que los esquemas de imagen permiten tender puentes entre los predicados del lenguaje natural y las manifestaciones gestuales. Se logra cierto nivel de expresividad de la escritura gestual, para alcanzar a su vez, el nivel conceptual de las descripciones de imágenes, es decir, se reconoce un gesto y a partir de aquel su concepto. Desde nuestro punto de vista, este es uno de los primeros enfoques en los cuales se usa el gesto para lograr una forma de escritura

gestual. Su enfoque se basa en características de apariencia activa.

Stergiopoulou & Papamarkos [175], proponen un nuevo métodos de reconocimiento gestual basado en un procedimiento de ajuste a través de una Red Neuronal gasificada (Self-Organized Neural Gas, SGONG) inspirada en un mapa auto-organizado (self-organizing map), en el cual inicialmente, la región de la mano se detecta aplicando una técnica de segmentación de color basada en un procedimiento de filtrado de color de piel en el espacio de color YCbCr. Entonces, la red SGONG se aplica en el área de la mano para acercarse a su forma. Luego, basadas en rejillas de salida de las neuronas producidas por la red neuronal, se extraen las características morfológicas de la palma. Estas características, están en concordancia con las características de los dedos y permiten su identificación. Finalmente, el reconocimiento del gesto de la mano se realiza mediante una técnica de clasificación basada en la verosimilitud y logra reconocer con éxito la mano y los dedos de la mano. Este enfoque es muy similar a Setiawan, pero a diferencia de aquel, la carga computacional es alta, por lo tanto sujeta de un ordenador.

Chen et al. [26], exponen un sistema de reconocimiento gestual en tiempo real, capaz de reconocer gestos formados por los dedos de la mano en una numeración del 1 al 9. Para lograrlo usan una cámara Web de gama baja. Su método detecta la región gestual utilizando un algoritmo adaptativo de segmentación de la región de la piel basado en el conocimiento previo. Características como rotación e invariabilidad de la escala son extraídas de la región del gesto. Estas características son enviadas a un clasificadas basado en Máquinas de Vectores de Soporte (Support Vector Machines, SVM) el cual genera el resultado que se usa en el reconocimiento gestual. Algo importante, es el tiempo de reconocimiento promedio, el cual alcanza aproximadamente 31,43 ms por frame, usando el poder de cálculo de un Core I3 CPU a 2,93 GHz.

Qian et al. [146], logra una interfaz para control robótico remota, basados en Kinect para lograr el reconocimiento gestual. El método de reconocimiento gestual, combina la información de profundidad de la imagen y lo envía al algoritmo de reconocimiento. El algoritmo implementa el clásico algoritmo de seguimiento Camshift para imagen, basado solamente en información del color y profundidad. Como clasificador usa el Modelo Oculto de Markov. Se logra así, la clasificación de gestos dinámicos. El sistema propuesto convierte el gesto en comandos para el robot y se basa en una arquitectura cliente-servidor. La comunicación es realizada por TCP/IP y no se indica la latencia del sistema, lo cual sería algo importante de evaluar. Se utilizó a un ordenador con CPU Core 2 Duo.

Itkarkar et al. [75] presentan un sistema de reconocimiento gestual, para el lenguaje de señal de la India. Los gestos son dinámicos son reconocidos usando características de contorno y de envolvente convexa. Como clasificador usa el algoritmo de detección de esquinas de Harris. Los resultados de exactitud se obtienen bajo iluminación fuerte, oscura y normal. La identificación del gesto en condiciones normales de Luz llegó al 80% y en condiciones de iluminación intensa a un 60%. Esto demuestra la sensibilidad del sistema al la cantidad lumínica existente. Este enfoque es válido también para se aplicado a interacción gestual por sus buenos resultados, aunque sujeto a condiciones de luminiscencia del ambiente.

Hsieh & Liou [69], proponen, a diferencia del anterior propuesto por Itkarkar et al., un sistema robusto que consta de tres módulos: zoom digital, detección adaptativa de la piel y reconocimiento de gestos de la mano. El primer módulo detecta la cara del usuario y la amplía usando el zoom. El segundo módulo, utiliza la información de color facial del usuario para detectar otras regiones de color de piel, tales como las manos. El último módulo, es la parte más importante la cual se encarga del reconocimiento del gesto de las manos tanto estático como dinámico. Las características usadas son formas geométricas y puntos, además de Modelos de apariencia multivista. Se logra la clasificación, manteniendo un historial de movimiento de la mano y aplicando el clasificador SVM. Se aplica el gesto al control de dispositivos de casa en un entorno controlado.

Ohn-Bar & Trivedi [130] muestran un sistema basado en la visión que emplea un RGB combinado y un descriptor de profundidad para clasificar los gestos de las manos. El método es una buena referencia para considerarlo una aplicación de interfaz humano máquina aplicada a un vehículo. Se emplean dos módulos que se hallan interconectados entre si: uno que detecta la mano en la región de interacción y realiza la clasificación, y otro que realiza el reconocimiento gestual. El enfoque que presentan es robusto al ruido por luminosidad y posible oclusión. Para reconocimiento dinámico del gesto usa SVM con núcleo o kernel Radial (RBF). Es necesario indicar que Ohn-bar & Trivedi no usaron el gesto como lenguaje, por lo tanto se vieron forzados a incrementar el número de gestos para cada comando. Este planteamiento, refuerza nuestra propuesta, la cual es alcanzar un método basado en el gesto que optimice el catálogo gestual empleado.

Chuan et al. [29] logran implementar un sistema de reconocimiento gestual usando Leap Motion. Los autores enfatizan que fue una solución más portátil y económica que Cyblerglove o Microsoft kinect utilizados en estudios existentes. El reconocimiento fue posible aplicando dos algoritmos de clasificación k-NN y SVM

sobre 26 letras del alfabeto inglés. El resultado más alto de exactitud lo obtuvo SVM.

De la extensa revisión científica realizada, podemos concluir que, el uso de dispositivos ópticos para lograr el seguimiento de la mano y la identificación del gesto es muy rica y variada. Muchos de trabajo presentados, realizan propuestas interesantes, para un mejor y óptimo reconocimiento usando cámaras. Pero, casi todos ellos, están limitados a un conjunto formado por hardware y software, que especializan en gran manera el reconocimiento. Como respuesta a la necesidad de integrar todos los recursos necesarios para un óptimo reconocimiento gestual en un mismo dispositivo, se han presentado al mercado informático dispositivos tales como Kinect y, recientemente, Leap Motion [®]. Kinect, cuyo fabricante es Microsoft ha sido extensamente usado en trabajos de investigación donde se requiere la detección del gesto del usuario a un nivel mucho más amplio, como es el gesto corporal inclusive. Mientras, que Leap motion, se restringe a la detección del gesto de las manos, en cercanía con el usuario.

Nuestro propósito, trata de circunscribirse al estudio del gesto de las manos del usuario para lograr una interacción natural. En este sentido, Kinect presenta un problema de adaptación para su reconocimiento, pues su rango de trabajo óptimo varía desde una distancia mínima de 70 cm. a una distancia máxima de 4 m. Cuando excedemos estos límites, tanto superior como el inferior, se pierde exactitud de reconocimiento. Esto, fue confirmado también por Rodríguez et al. [154], quienes concluyeron que “era imposible captar los puntos de la mano a distancias menores de 30 cm, ya que, ha esta distancia no hay convergencia entre la cámara de profundidad y el infrarrojo, que tienen un umbral mínimo de 70 cm”. Kinect también requiere una calibración externa [112]. En cambio, Leap Motion logra captar escenas a distancias mínimas, incluso inferiores a 1 mm y máximas de hasta 60 cm, sobre el dispositivo [188].

Otro punto a favor de Leap Motion es su baja latencia ($85ms$) en comparación con Kinect ($170ms$) [20]. No requiere calibración externa y puede ser calibrado directamente por sus herramientas software. Además, logra detecciones muy precisas con un pequeño número de características con respecto a Kinect [112]. Podemos concluir entonces, que el dispositivo que mejor se acopla a tareas de control gestual, minimizando la oclusión y maximizando la exactitud en tareas de control reactivo es Leap Motion. Se describe en mayor profundidad al dispositivo en la Sección 2.2.

2.1.3. Enfoques para clasificación de gestos

El análisis de los trabajos científicos, presentados anteriormente, también permitió concluir que existen dos enfoques principales para lograr la clasificación gestual usando medios ópticos, y son los siguientes: Enfoque basado en reglas y enfoque basado en Machine Learning.

Enfoque basado en reglas

Básicamente, consiste en un conjunto de reglas codificadas manualmente entre las entradas de entidades. Su proceso en forma resumida sería: Una vez determinado el gesto a identificar, se extraen el conjunto de características y se las compara con las reglas codificadas y aquella regla que coincide con la entrada se dice que es el gesto buscado. Como ejemplo, Murthy & Jadon [122] citan lo siguiente: *“predicados relacionados con características de bajo nivel del movimiento de las manos se definen para cada una de las acciones en consideración. Cuando un predicado de un gesto se satisface sobre un número fijo de marcos consecutivos, el gesto es devuelto. Un problema importante con los enfoques basados en reglas es que se basan en la capacidad de un ser humano para codificar reglas”*.

Enfoque basado en Machine Learning

En este enfoque se usa algoritmos cuya salida permiten determinar si una clase a la cual pertenece el gesto. Su objetivo es identificar patrones ocultos en un gran conjunto de datos. Esta definición y otras aproximaciones fueron expuestas en el Capítulo 1 Sección 1.4. Uno de los algoritmos que mayor atención ha merecido en este enfoque es el llamado Modelo Oculdo de Markov (Hidden Markov Models, HMMs).

2.2. El dispositivo Leap Motion

El dispositivo Controlador Leap Motion ®(Leap Motion Ccontroller, LMC), escogido por las razones argumentadas en la Sección 2.1.2, nos permite profundizar y analizar el uso de tecnología óptica e infrarroja, para lograr un sistema de reconocimiento gestual basado en él. A pesar de ser relativamente nuevo en el mercado, este wearable de bajo costo ha sido utilizado con éxito en varios trabajos de investigación dedicados al control gestual, especialmente en dispositivos de casa y en robots, que usan la técnica de tracking para seguir los movimientos de la mano

[142, 51, 41, 137]. También ha sido usado con éxito en el manejo de dispositivos médicos que aplican el tracking como medio operativo [131, 17]. Pero, a pesar de ello, también se ha abierto un claro debate sobre su real aplicación y exactitud demostrada [188, 182, 9].

Al respecto, LMC reporta una gran exactitud en la detección del gesto en tiempo real como lo indican [112]. Permite un seguimiento del gesto tanto estático como dinámico, y además, estudios anteriores indican que su exactitud está inclusive por debajo del milímetro [61, 9]. Sin embargo, al compararlo en precisión con un dispositivo mouse estándar, por ejemplo, es notoria su alta tasa de error (LMC 7.8% y mouse 2.8%) [9]. Sin embargo, la gran ventaja de LMC es que el reconocimiento del gesto es independiente de las posibles características físicas del usuario (estatura, peso, masa muscular, edad, etc), es decir, el gesto no requiere ser personalizado, a diferencia de otros dispositivos no ópticos (por ejemplo, MYO [®], Guantes de datos). Desgraciadamente, hasta el momento, el fabricante no ha liberado el algoritmo de reconocimiento, lo cual limita a los trabajos de investigación que desean hacer mejoras en este sentido. Quizá, otra desventaja es que el movimiento gestual del usuario debe ser realizado directamente sobre el controlador y dentro de su área de interacción. Pero a su vez, constituye una fortaleza porque para aplicaciones de control, la cuales requieran de una consola de administración gestual en un ambiente controlado es ideal. En este sentido, un sistema electrónico (por ejemplo, robótico) que requiera la presencia obligatoria del operador, como parte de un protocolo de seguridad, sería la aplicación perfecta.

La Figura 2.2.a muestra una vista externa de Leap Motion en plena operación y, la Figura 2.2.b, trata de indicar si su campo de visión efectivo. Los datos de ángulos, distancias y otras características son ofrecidos por el fabricante en su página Web.



Figura 2.2: Leap Motion

LMC detecta el gesto de la mano del usuario, realizado en el aire y sin tocar el dispositivo. Éste dispositivo contiene en su interior, dos cámaras estéreo monocromáticas y tres led infrarrojos de una longitud de onda de 850 nm, que permiten construir escenas en 3D [182, 9]. La Figura 2.3 muestra una vista interior del dispositivo, su tarjeta principal y sus componentes hardware. Según el fabricante (Leap Motion Inc., San Francisco, CA, US) reporta posiciones discretas, gestos y movimientos, su unidad de tiempo es el microsegundo (μ), de velocidad, es el milímetro por segundo (mm/s), sus ángulos están dados en radianes y se conecta vía USB al ordenador. Para mayores detalles consultar ¹.



Figura 2.3: Leap Motion y sus componentes

El LMC combina, los datos obtenidos por sus dos cámaras con los datos obtenidos de sus tres LEDs infrarrojos, para detectar el gesto y sus características (Ver Figura 2.3). Esta combinación le permite formar un conjunto de datos (fotograma o frame) o instantánea, que se genera durante el muestreo a la frecuencia del dispositivo. El frame puede o no contener información válida. La información contenida en el frame incluye esencialmente datos tales como: número de dedos, número de manos, tipo de gesto, marca de tiempo, *id* de frame, tamaño de mano (basado en una esfera trazada alrededor de la circunferencia de la palma y los dedos) Etc.

Un gesto puede ser simple (también llamado .estático”) si la posición de la mano no cambia, o compleja (también llamada dinámica) si cambia durante la ejecución del gesto con algunos movimientos. Uno, dos o más frames pueden caracterizar un gesto. Las variaciones entre frames pueden definir los factores que logran identificar a un gesto específico. Estos factores, pueden definirse como parámetros para el reconocimiento de gestos, y son los siguientes: escala, rotación y movimiento. Un gesto puede implicar dedos y manos, pero tiene que ser ejecutado directamente encima del dispositivo.

¹ <https://www.leapmotion.com/>

2.2.1. Gestos predefinidos

El SDK de desarrollo de LMC establece un conjunto de gestos predefinidos, que son una combinación de gestos dinámicos y estáticos y que se muestran en Figura 2.4. Las imágenes han sido capturadas al realizar el gesto directamente sobre el LMC. Sin embargo, debemos recordar que el gesto es seguido por el dispositivo en todo momento, logrando realizar el tracking del movimiento.

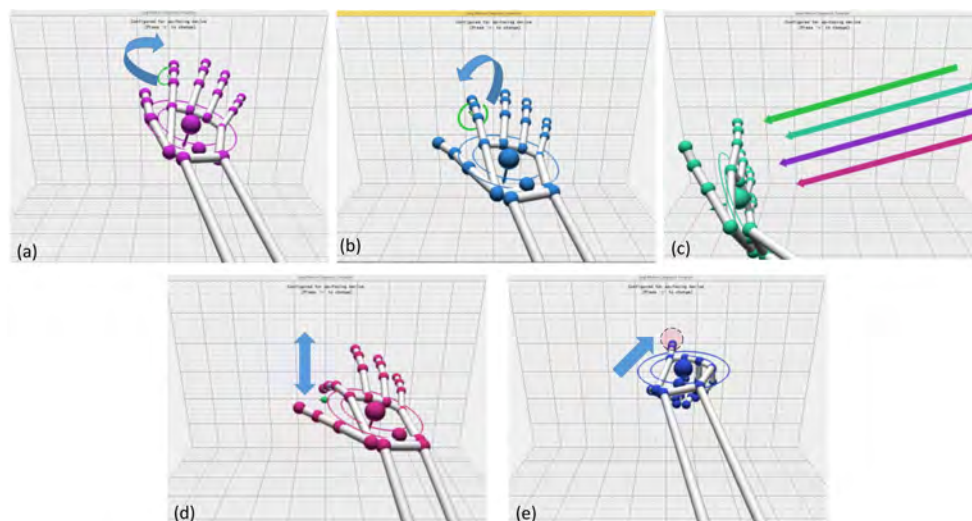


Figura 2.4: Gestos predefinidos de Leap Motion

Atendiendo a lo mostrado en la Figura 2.4 podemos detallar a continuación los gestos predefinidos:

- Gesto Círculo o Circle, a favor y en contra de las manecillas de reloj. En el primer caso, el dedo índice traza un círculo a favor de las manecillas del reloj mediante un movimiento circular (Figura 2.4.a); en el segundo caso, el dedo índice traza un círculo en contra de las manecillas del reloj mediante un movimiento circular en esa dirección (Figura 2.4.b). La mano permanece estática sobre el LMC en ambos casos, con los dedos separados.
- Gesto barrido o Swipe. Es asumido al realizar un rápido movimiento o desplazamiento lineal (de izquierda a derecha o viceversa) de la mano sobre el dispositivo LMC (Figura 2.4.c).
- Gesto toque de tecla o Key Tap. Este gesto describe un desplazamiento de arriba hacia abajo y viceversa del dedo índice y, trata de simular, al toque

de tecla (Figura 2.4.d). Mientras se ejecuta el movimiento, debe mantenerse estática la mano y resto de dedos.

- Gesto toque de pantalla o Screen Tap. Este gesto es similar al anterior, pero con el dedo índice fijo y apuntando hacia adelante, mientras, se desplaza hacia adelante toda la mano longitudinalmente. Este gesto, describe una posición inicial (sobre el LMC) y una final (con desplazamiento hacia adelante). Los demás dedos levemente recogidos sobre la palma. Para realizarlo correctamente, se debe suponer tocar un punto en pantalla (touch point) simulado un gesto de toque (Touch) en una pantalla táctil. (Figura 2.4.e).

Aquellos gesto que involucran características de movimiento, rotación o traslación, son considerados gestos continuos porque para determinarlos el LMC utiliza varios frames. Aquellos gestos que no requieran estas características o permanezcan estático, son discretos y pueden ser detectados inclusive en un solo frame. Esto quiere decir, que en un gesto dinámico, LMC verificará cada frame para comprar las características que le indiquen un cambio. Un gesto dinámico iniciado, crea un ciclo de actualización permanente de su estado, hasta su detectar la terminación, interrupción o abandono del gesto. Por ejemplo, en el caso del gesto Circle, cuando el dedo se sale de la geometría del círculo o su velocidad es muy baja, y en el caso de Swipe, cuando su movimiento es demasiado lento. LMC admite una velocidad máxima de $1,000mm/s$.

Esta característica del dispositivo, produce varias hebras de reconocimiento que acompañan a los frames que iniciaron un gesto. El resultado de estos procesos pueden ser, el gesto detectado o la pérdida de frames usados en la detección de ese gesto. Es decir, pueden existir falsos positivos de reconocimiento, provocados por abandono del gesto, no finalizado o caducidad de tiempo. Esto se logra, debido a que LMC asigna un *id* a cada frame correspondiente a un gesto detectado, el cual es etiquetado con el posible gesto a identificar y que será comparado con aquel para lograr determinar el inicio, transición y fin de dicho gesto. Se trata de evitar así, en lo posible, los falsos positivos en el reconocimiento pero, la naturaleza dinámica de muchos gestos aumentan la complejidad del reconocimiento, siendo necesario establecer características que faciliten su reconocimiento.

2.2.2. Gestos implementados

El agregar gestos propios, a la librería por defecto del dispositivo nos permitirá evaluar su comportamiento y además, establecer el funcionamiento del algorit-

mo de reconocimiento en base a las características del gesto. Los nuevos gestos se ajustaron en su diseño a los parámetros establecidos por la API del controlador. Así mismo, se implementaron los cambios necesarios en la forma de identificación del gesto, para lograr establecer un nuevo programa de reconocimiento, que integraba los gesto predefinidos por el LMC y aquellos que planteamos en la presente investigación.

En esta etapa, el objetivo principal fue determinar además, la adaptabilidad para implementación de nuevos gestos y si, la nueva librería gestual, podría ser aplicada con éxito en una Interfaz Natural de Usuario de tipo Gestual (INU-G), orientada al control reactivo en dispositivos robóticos.

Los gestos, fueron pensados tratando de cumplir en lo posible, con la recomendación dada por Pomboza & Holgado [142]. Esta recomendación sugiere que, un gesto debe ser lo suficientemente cómodo, para ser mantenido por un tiempo medianamente largo y así garantizar la mayor exactitud de reconocimiento, disminuir la tasa de error (ruido), evitar cansancio o fatiga del usuario y garantizar la detección. Los gestos y su representación 3D se muestra en la Figura 2.5.

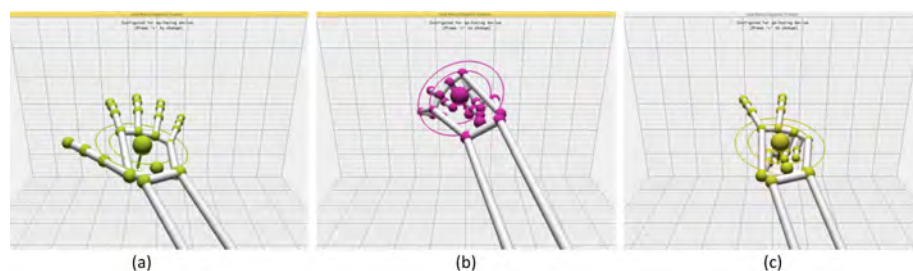


Figura 2.5: Gestos implementados en esta investigación

A continuación realizamos una breve descripción de cada uno de los gestos implementados.

- Gesto mano abierta o Open Hand, con los dedos totalmente extendidos y la palma de la mano hacia abajo como se indica en la Figura 2.5.a.
- Gesto mano cerrada o Close hand, todos los dedos recogidos en la palma formando un puño cerrado (Figura 2.5).b.
- Gesto dos o Two, que se logra con los dedos índice y medio extendidos, y abiertos, en sentido contrario a cada uno. Los demás dedos están recogidos en la palma y cubiertos con el dedo pulgar. (Figura 2.5).c.

Todos los gestos fueron realizados a una altura de 10 cm aproximadamente sobre el LMC. Los gestos, a pesar de ser posturas estáticas, pueden sufrir variaciones en su localización sobre el dispositivo, pues en tracking se considera en realidad que ningún gesto es estático en su ubicación. Lo que permanece constante del gesto, es la pose.

2.2.3. Características gestuales

El área de interacción efectiva tiene la forma de una pirámide invertida con centro en el dispositivo. Cubre un volumen de $0,23m^3$ y el rango se extiende aproximadamente de 25 a 600mm por encima del dispositivo. El sistema de coordenadas que utiliza LMC, es cartesiano, siguiendo la regla de la mano derecha y centrado en el punto medio del dispositivo (Figura 2.6).

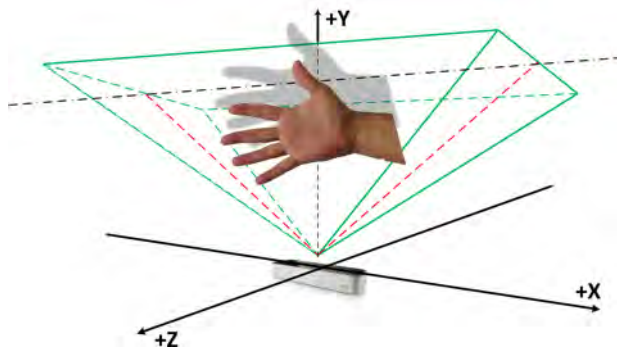


Figura 2.6: Leap Motion y su área de interacción

Componentes del frame

Los datos, que Leap Motion captura, son recogidos por su controlador y asignados al último frame capturado. Crea entonces, un objeto de tipo *Frame*, el cual contiene toda la información correspondiente a los puntos rastreables de la mano, tales como: número de manos, el tipo de mano (izquierda o derecha), su posición (coordenadas x, y, z), el número de dedos, circunferencia de la palma, un identificador del frame ($frame_id$), gesto iniciado (con etiqueta de que empieza, actualiza o termina) y, otros datos de posibles objetos detectados en la escena tales como una herramientas. Una herramienta, para Leap Motion, es más larga, fina y rígida que un dedo. Los datos son transmitidos mediante conexión USB a un host o computador quien ejecuta el algoritmo de reconocimiento. Un objeto *Hand* guarda

las características correspondientes a una mano o posibles manos detectadas en el campo de visión de LMC.

En la Tabla 2.2 se muestran todas las características que proporciona la API y que han sido usadas en este estudio.

Tabla 2.2: Características de la API

Palma		Dedos	
Nombre	Tipo	Nombre	Tipo
Normal	Vector	Dirección	Vector
Posición	Vector	Longitud	en mm
Velocidad	Vector (mm/s)	Velocidad de la punta	mm/s
Esfera (centro)	Vector	Posición de la punta	Vector
Esfera (radio)	en mm	Sentido de giro	Vector

LMC, analiza el movimiento desde el frame anterior al actual, mostrando datos de traslación, rotación y factores de escala. Se analizan todos los objetos presentes en un determinado frame. De este modo, si se colocan dos manos sobre LMC, el objeto *Frame*, basa las transformaciones realizadas entre ambas manos. Sin embargo, se puede acceder a los atributos de cada *Hand* por separado, si así se requiere. Los atributos de cada objeto *Frame*, basados entre el frame actual y el anterior, son: eje de rotación, ángulo de rotación, matriz de rotación, factor de escala y traslación. La Tabla 2.3 presenta en forma resumida, todos los factores que afectan al movimiento y que son parte del frame.

Tabla 2.3: Factores que afectan al movimiento

Tipo	Frame	Mano
Escala	La escala de un frame, refleja el movimiento de los objetos de la escena hacia él o lejos de él. Por ejemplo, cuando una mano se mueve más cerca de la otra	La escala de la mano refleja el cambio en la propagación de los dedos o la apertura de la mano
Rotación	La rotación del frame, refleja el movimiento diferencial de los objetos dentro de la escena. Por ejemplo, con una mano arriba y la otra hacia abajo	La rotación de la mano refleja el cambio en la orientación de una sola mano
Traslación	La traslación del frame muestra el cambio medio en la posición de todos los objetos en la escena. Por ejemplo, las dos manos se mueven hacia la izquierda, hacia arriba o hacia adelante	La traslación, refleja el cambio en la posición de esa mano

El movimiento, en el controlador Leap Motion, está dado por los cambios que el gesto pueda presentar ante el dispositivo y, como ya lo habíamos mencionado, eran escala, rotación y traslación. Un gesto que, no implique traslación puede estar definido en un solo frame, en caso contrario será necesario varios frames para lograr determinar la transición y finalización del gesto. Por ejemplo, en lugar de seguir el cambio de posición de los dedos individualmente en varios frames, se puede usar el factor de escala calculado entre dos frames para lograr el cambio del tamaño de un objeto. Estos factores se describen en la Tabla 2.3.

El LMC proporciona principalmente los siguientes datos, que ayudan a caracterizar el gesto: *Hand tracking*, vector de dirección (vector unitario q) y vector de traslación (vector unitario k); Velocidad y dirección del movimiento de la mano (mm/s); Seguimiento de los dedos: vectores de dirección y traslación de dedos visibles sin nombres ($F_i, N \in [0, 5]$); Y, finalmente, la capacidad de rastrear objetos largos y rectos (proporciona dirección y su vector de traslación), calculando el radio de una circunferencia del interior de la palma (centro de palma C , orientación de la mano con dos vectores unitarios q y k , radio r). Esto se muestra en la Figura 2.7.

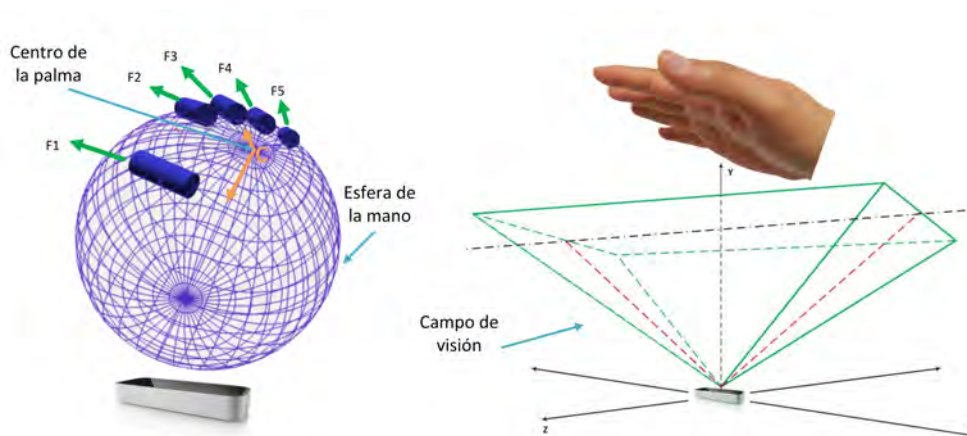


Figura 2.7: Componentes para identificación del gesto.

Frecuencia de muestreo

LMC mostró una tasa de muestreo de 57,47 Hz (f_i) sin ninguna mano u objeto en su área de visión, es decir, sin carga gestual alguna (Figura 2.8.a). Los frames fueron generados y entregados con un intervalo constante aproximado de $17391\mu s$ ($0,017391s$). En contraste, al dar una carga gestual al dispositivo, éste aumentó su

frecuencia hasta los $119Hz$ (f_f) inclusive (Figura 2.8.b). Los frames, en este caso, fueron lanzados a una mayor frecuencia lo cual nos permitió conocer mejor el comportamiento del dispositivo y, como su algoritmo trata de realizar el reconocimiento gestual.

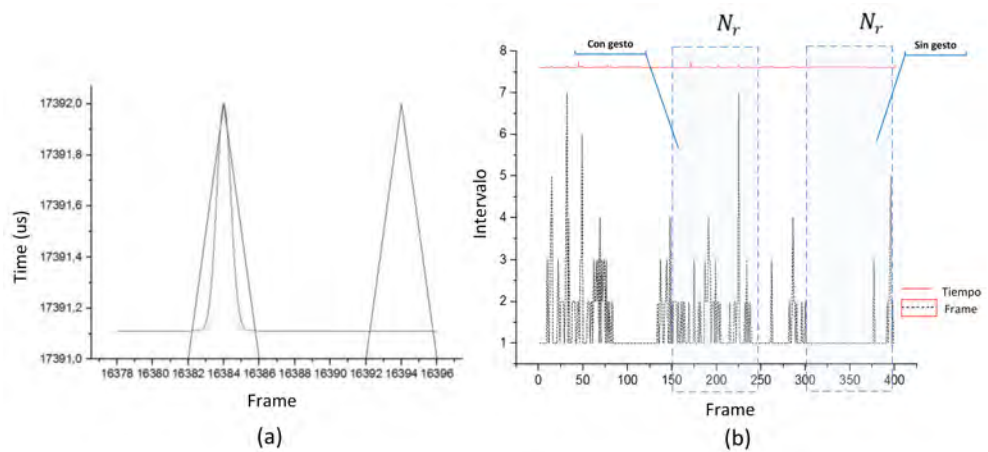


Figura 2.8: Tasa de muestreo con y sin carga gestual

2.3. Análisis del proceso de reconocimiento

El diseño experimental del enfoque presentado en este trabajo de investigación, para analizar el LMC se realiza en dos fases. En la primera fase (Figura 2.9.a), recogimos los datos generados por el LMC a través de su API y usando el SDK ofrecido por el fabricante. La identificación de un gesto, depende de la información contenida en uno, dos o más frames, generados durante las pruebas. Luego, se ha analizado la información frame por frame, verificando su contenido y su validez, para verificar la efectividad del reconocimiento. Este análisis es esencial para reconocer los gestos y evaluar la calidad del reconocimiento. En nuestro estudio, hemos empleado sólo la mano derecha. La Figura 2.9, muestra algunas características del frame y posiciones.

LMC explora continuamente los cambios ocurridos en las características, las posiciones de la mano y dedos con respecto al modelo 3D generado de la mano. En base a este análisis, podremos diagnosticar los intervalos de tiempo que emplea el reconocimiento y determinar el más apropiado para un gesto de control.

En la segunda fase (Figura 2.9.b), establecemos un indicador, la efectividad del gesto, que especifica un criterio para determinar aquellos gestos que pueden ser

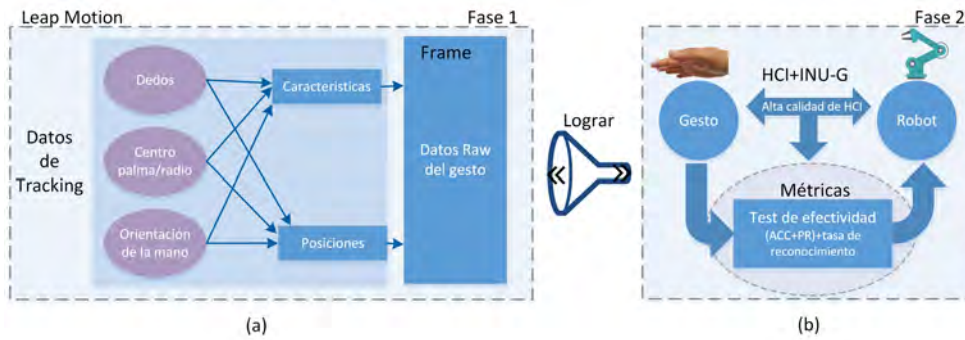


Figura 2.9: Enfoque dado al análisis del dispositivo.

válidos para la interacción con dispositivos electrónicos, especialmente dispositivos robóticos. En esta fase se realiza una implementación usando el sistema robótico LEGO Mindstorms EV3, como un caso de estudio. El sistema robótico usa una interfaz gestual de control reactivo, la cual logra validar los resultados encontrados en la fase 1, con aquellos gestos que satisfagan el indicador de efectividad.

2.3.1. Participantes

Ocho voluntarios participaron en nuestro experimento (5 hombres y 3 mujeres), entre las edades de 19 y 38 ($M = 26,75$, $SD = 8,11$). Todos los voluntarios tenían experiencia en el uso del ordenador y todos los hombres indicaron tener también experiencia en el uso de videojuegos.

2.3.2. Biblioteca gestual propuesta

Además de las recomendaciones dadas por [142], en cuanto a la facilidad de creación del gesto por parte del usuario, podemos mencionar que, los gestos implementados están pensados para asegurar el mayor grado posible de exactitud en el reconocimiento, disminuir en lo posible la tasa de error (ruido), evitar el cansancio o agotamiento y, por último, garantizar la fiabilidad. La biblioteca gestual completa es mostrada en la Figura 2.10 la cual incluye a gestos simples (a, b y c) y a gestos complejos (d, e, f, g y h).

2.3.3. Protocolo experimental

Un software especialmente diseñado en Java fue usado para la captura de datos. Las pruebas se realizaron con una temperatura ambiente constante de $20^{\circ}C$ y una intensidad luminosa de 245 lux. El LMC estaba conectado a una computadora con

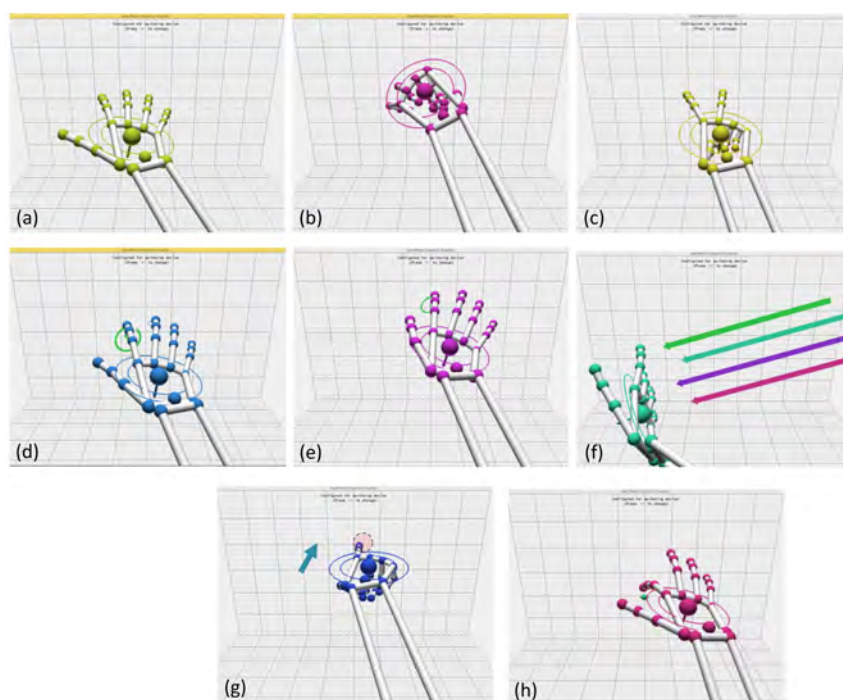


Figura 2.10: Biblioteca gestual propuesta.

CPU Intel Core i7-X455L a 3.1 GHz y 12 GB de RAM, OS WIN 8.1, HDD 1TB. Tarjeta gráfica NVIDIA Geforce 820M con 2 GB. El área de pruebas consistió en un gran escritorio y una cómoda silla con respaldo. El LMC se conectó a una computadora frente al monitor, teniendo en cuenta las distancias y la ubicación recomendadas por el fabricante y, aquellas consideradas en nuestro experimento, como se muestra en la Figura 2.11.

El software almacenó los datos generados por el objeto de Frame al realizar el tracking de los gestos. El procedimiento para obtener la recolección de muestras se realizó de la siguiente manera:

- El investigador explicó brevemente (verbal y gráficamente en el monitor) el gesto que debía realizarse antes de cada prueba. Además, para los gestos complejos, le indicó al voluntario que mantuviera una velocidad estable, siguiendo el movimiento del objeto en pantalla. En ocasiones fue necesario un ensayo previo para evitar muestras defectuosas (ver Figura 2.11).
- El movimiento gestual, en aquellos gestos que incluían traslación, la velocidad próxima esperada fue de 500 mm/s (el fabricante indica que puede alcanzar una velocidad de hasta 1000 mm/s). Esto, gracias al dibujó una esfera (punto

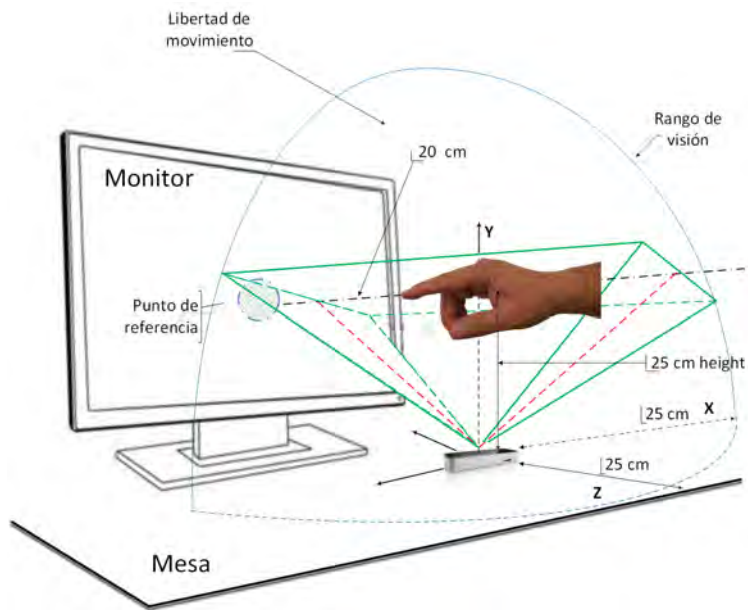


Figura 2.11: Condiciones experimentales

de referencia) en el monitor, el cual se desplazaba de izquierda a derecha manteniendo la velocidad indicada (Figura 2.13). Esto garantizó la adecuada ejecución del gesto.

- Para todos los gestos, una imagen en el monitor ilustró el gesto deseado y marcó el tiempo restante, mediante un temporizador de cuenta regresiva en pantalla (ver Figura 2.12). El temporizador fue iniciado por el investigador al momento de inicio de la prueba. Un sonido marcaba el inicio y fin de la prueba y servía para que investigador y voluntario detuviesen el gesto y la captura respectivamente.
- En cada prueba, el software almacenaba los datos pertinentes y creaba un nuevo archivo para el siguiente gesto de la prueba.
- Este proceso fue repetido hasta que todos los gestos fueron completados por cada voluntario.

El tiempo de muestreo para cada gesto fue de 15 segundos por usuario. Cada participante realizó ocho gestos para un tiempo total de muestreo de 960s. La aplicación de software también añadió información en cada lectura a fin de vincular el marco al gesto. Inicialmente se supone que todos los gestos pueden ser válidos para control, hasta determinar las características ideales del gesto de control.



Figura 2.12: Escenarios de entrenamiento y prueba para la captura de datos.

La información obtenida en cada prueba fue guardada en un archivo individual correspondiente a cada una de ellas. Los datos eran etiquetados, para que tengan correspondencia con cada gesto, con el fin de simplificar el posterior procesamiento. Cada muestra, contenía la información que entregaba el frame. La información, eran datos relacionados al gesto y al frame. En los gestos dinámicos especialmente, la información contenida en el frame no permitía reconocer al gesto, pues como se expuso anteriormente, dichos gestos necesitan más de un frame. Por ello, algunos frames, podían contener poca información como para lograr identificar al gesto. Esto coincide con lo informado por Guna et al. [61] respecto al comportamiento del dispositivo para casos de gestos complejos.

Todos los gestos debían mantener una altura uniforme en cada lectura para garantizar la muestra. Para lograrlo, en gestos dinámicos y estáticos, durante nuestro experimento se tomó como referencia de altura, y de velocidad de movimiento, a una aplicación especialmente diseñada para el efecto. Dicha aplicación, mostraba un círculo de color rojo en pantalla, de referencia de altura, para el caso de los gestos Circle, Key Tap y Screen Tap; mientras, que para el gesto Swipe, el punto se desplazaba en la pantalla, referencia de velocidad y altura, a una velocidad de 500mm/s calculada en base a las dimensiones del monitor de pruebas que era de 21 pulgadas. Este proceso, aunque sencillo, garantizaba medidas aproximadas en cada prueba. El proceso se ilustra de mejor manera en la Figura 2.13.

Para identificar los frames que corresponden a un gesto, tanto dinámico como estático, se acudió al *id* de identificación del frame junto a la etiqueta de informa-



Figura 2.13: Gestos y punto de referencia

ción del gesto y de su estado (inicio, transición o fin). Esta información, lograba determinar si el gesto ha sido o no detectado, mantenido o finalizado.

El volumen de espacio que LMC, bajo estas condiciones, podía sensor fue de $62,500cm^3$ ($-250\text{ mm} < x < 250\text{ mm}$, $-250\text{ mm} < z < 250\text{ mm}$, $0\text{ mm} < y < 250\text{ mm}$). La dimensión en y ha sido limitada hasta la altura estimada al punto de referencia. Este espacio fue sistemáticamente cubierto al realizar las pruebas. En cada una de ellas, se recorría una longitud de aproximada de 500 mm, sobre la mesa de pruebas (dimensión x) y, como lo explicamos anteriormente, manteniendo en lo posible la altura (dimensión en y).

Como se puede apreciar en la Figura 2.13, los gestos fueron realizados de acuerdo a su naturaleza, es decir, tanto estáticos como dinámicos, manteniendo el gesto y el antebrazo en la posición correcta mientras se realizaba el gesto. Los gestos dinámicos, fueron realizados en forma sucesiva hasta completar el tiempo determinado.

Las muestras de datos obtenidas en cada experimento se agruparon posteriormente por cada gesto. El *id* del frame y la marca de tiempo se mantuvieron constantes durante todo el experimento. El análisis del *id* del frame determinó frames secuenciales y otros con intervalos variables relacionados a los gestos. En gestos simples (posturas) el gesto fue identificado inclusive en un mismo frame.

Para gestos complejos (con movimiento de traslación) la identificación del gestos requería una serie de frames. En este caso, el reconocimiento se logró identificando todo el conjunto de frames que conformaban el gesto teniendo en cuenta que pasaba por estados como inicio, mantenimiento y fin del gesto. Los frames que conformaban un gesto podían inclusive no contener datos, lo que anulaba el seguimiento de un gesto en un momento dado. Se realizó un análisis de los frames obtenidos por cada gesto desde el inicio hasta hasta el final del gesto.

2.3.4. Frecuencia de muestreo

En el reconocimiento por tracking realizado con LMC para un gesto complejo éste alcanzó una frecuencia de muestreo de f_i de $114,932Hz$. En este caso, los frames fueron generados y entregados en un intervalo de $8,701ms$. Sin embargo, también observamos que la tasa de muestreo no era estable y variaba de acuerdo a la complejidad del gesto. Esto es debido quizá al comportamiento observado comúnmente en dispositivos ópticos diseñados para tracking, como lo indicar Erol et al. [49], los cuales requieren un conjunto de frames mínimo para lograr reconocer el gesto dinámico. La Figura 2.14 trata de mostrar este comportamiento, tomando como referencia al gesto (f).

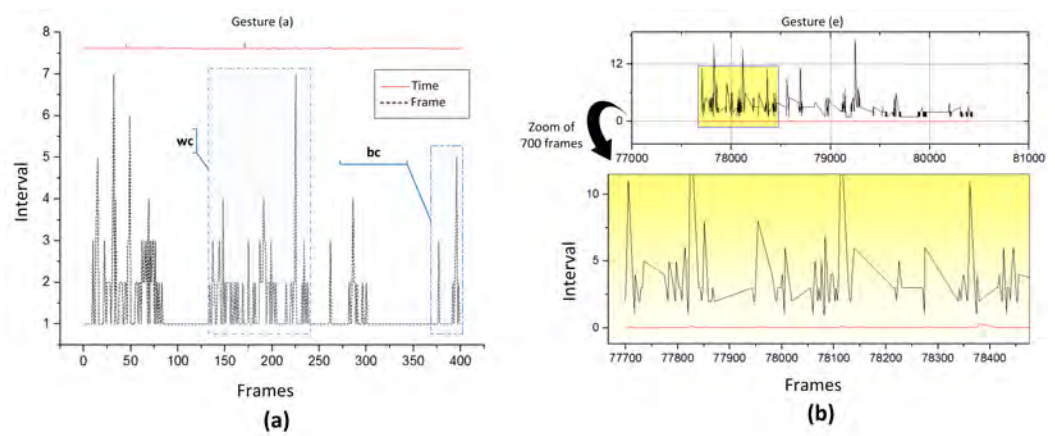


Figura 2.14: Variación de frames para reconocimiento del gesto (a) y (e).

En la Figura 2.14.a se puede apreciar el retardo provocado en los intervalos entre frames, causados en el proceso de reconocimiento del gesto (a). En este caso el retardo se mantiene mínimo, alcanzando el reconocimiento en uno o dos frames. La Figura 2.14.b, nos permite observar el retardo en el reconocimiento provocado por el gesto (e) al intentar lograr el reconocimiento mediante los estados intermedios o frames parciales, siendo necesario cinco o más frames para lograrlo. Esto, como ya lo mencionamos, obliga a aumentar la frecuencia de trabajo del dispositivo, el cual lanza más frames para lograr el reconocimiento del gesto. La forma en que lo interpreta el algoritmo de reconocimiento de LMC, obliga a esperar un frame de cierre el gesto iniciado con alguno posible anterior. La representación en la figura, toma como referencia a los frames $(n+1)$ que han logrado identificar al gesto iniciado en el frame (n) . Como ya se dijo anteriormente, el reconocimiento puede ser logrado en 1, 2, 3, o más frames dependiendo de la complejidad del gesto.

También, debemos indicar algo que concuerda con lo anterior, y es que, el LMC entregó frames que, a pesar de ser considerados parte de un gesto, no contenían información que identifique o corresponda al gesto. En estos casos, se provocó un aumento en el tiempo de reconocimiento, lo cual, afectó negativamente el rendimiento, para un gesto en particular. Esto se debe probablemente al algoritmo implementado y su forma de implementación, para el reconocimiento gestual en base al controlador, el cual depende directamente del fabricante.

2.3.5. Características: manos y componentes

Las características que emplea LMC para determinar el gesto, están contenidos por su API. El tracking de la mano realizado por LMC, nos entrega los siguientes datos: posición del centro de la palma (centriode), velocidad de la palma, la normal asociada a la palma, dirección de la mano, centro de la esfera que forma la mano y radio de curvatura de dicha esfera. Todo esto, ha tratado de representarse en la Figura 2.15. Concretamente, el objeto *Hand* de LMC proporciona información, capturada por el frame, de: la posición 3D, características y movimiento de la(s) mano(s) detectada(s) en su campo de visión descrito anteriormente. La información sobre número de dedos y otros objetos detectados es presentada en este objeto, pudiendo inclusive detectar más de dos manos. Esto, indicaría que hay más de una persona interactuando con el dispositivo, pero, el problema de oclusión (típico en medios ópticos) hace que sus características y datos se vean alterados y sean poco fiables. Por ello, solo se recomienda trabajar hasta con dos manos a la vez.

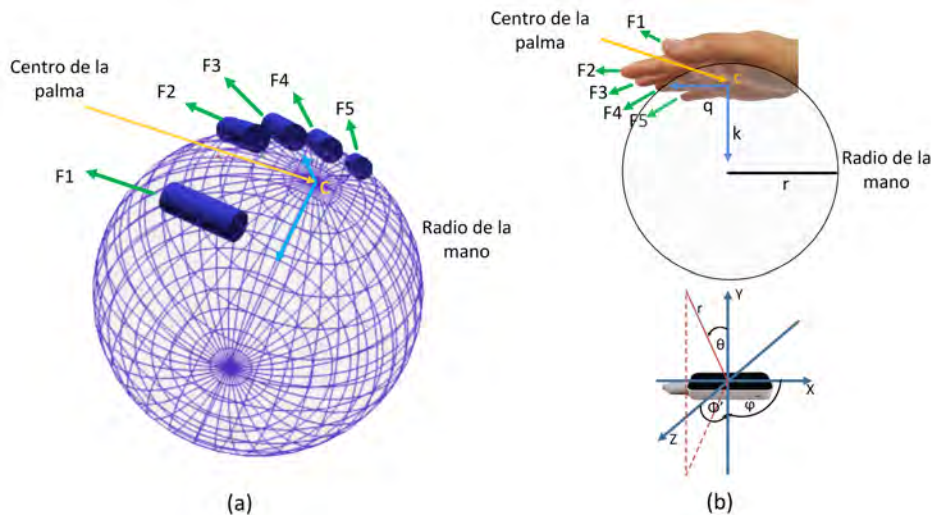


Figura 2.15: Datos de reconocimiento entregados por la API de LMC.

La dirección de la mano q , el radio r y la normal k , que describe la palma de la mano, se encuentran especificados en sus vectores unitarios, como se indica en la Figura 2.15.b. Estos vectores, describen la orientación de la mano con respecto al sistema de coordenadas de LMC. Del mismo modo, los datos que muestra la Figura 2.15.a nos permiten apreciar la esfera, que describe la palma de la mano, asociada a su radio de curvatura y centro C . Cada dedo u objeto *Finger* es asociado a un vector F_1, F_2, F_3, F_4, F_5 , el cual contiene sus características en longitud, anchura, dirección, velocidad y posición. Los vectores asociados a cada dedo, tratan de hacer una descripción de los llamados objetos *Pointables*, que incluyen a *Fingers* y *Tools*. Los dedos, limitan el radio de la circunferencia asociada a la palma de la mano. Además, es posible acceder a la información de traslación, rotación y factor de escala mencionados anteriormente, incluidos como parte del objeto *Frame*.

Parámetros de reconocimiento gestual

En el caso de los nuevos gestos (a, b, y c), necesitamos definir el patrón de gestual con las características apropiadas, requeridas para identificar los gestos. La caracterización de los gestos esta dada por de dos definiciones principales: la mano y los dedos. La mano está determinada por el centro de la palma C , la orientación de la mano (vectores q y k) y la curvatura de la mano en base a el radio de mano r y el ángulo θ entre r e y , lo cual fue descrito en la Sección 2.3.5. Con respecto al objeto *Finger*, nos interesa el número de dedos detectados y los vectores asociados con cada dedo F_i .

La mano está determinada por, el centro de la palma C , la orientación de la mano (vectores q y k) y la curvatura de la mano con respecto al radio que describe el ángulo θ entre r e y . Con respecto a los dedos, nos interesa el número de dedos detectados y los vectores asociados con cada uno F_i . Además, en caso necesario, el ángulo de las yemas de los dedos está definido por [111, 112]. Para lograr una estimación del valor de r y θ hemos tomado la definición realizada por Guna et al. [61], en la cual:

$$r = \sqrt{x^2 + y^2 + z^2} \quad (2.1)$$

$$\theta = \arccos\left(\frac{y}{r}\right) \quad (2.2)$$

$$\varphi = \begin{cases} \arctan(\frac{z}{x}), & x \geq 0 \\ \arctan(\frac{z}{x}) + \pi, & z \geq 0 \quad x < 0 \\ \arctan(\frac{z}{x}) - \pi, & z < 0 \quad x < 0 \\ \frac{\pi}{2}, & z > 0 \quad x = 0 \\ -\frac{\pi}{2}, & z \leq 0 \quad x = 0 \\ \text{no definido}, & z = 0 \quad x = 0 \end{cases} \quad (2.3)$$

Asumiendo la simetría en el LMC, sobre los ejes x y z , según Guna [61] se define al azimut de los ángulos como:

$$\varphi' = \begin{cases} \arctan(|\frac{x}{z}|), & z \neq 0 \\ \frac{\pi}{2}, & x \neq 0 \quad z = 0 \\ \text{no definido}, & x = 0 \quad z = 0 \end{cases} \quad (2.4)$$

El ángulo φ' es medido desde el eje x (no desde el eje z como en el caso del ángulo φ) y la línea que conecta el origen de coordenadas con la proyección de la localización medida en el plano $x-z$. Como el ángulo φ' se define bajo la suposición de simetría en el rendimiento del controlador sobre los ejes xyz , se define por lo tanto en el intervalo $(0, \frac{\pi}{2})$ radianes.

2.3.6. Métricas

Para lograr definir la variación de tiempo en el reconocimiento del gesto, hemos definido una *tasa de reconocimiento* (S_g), como la velocidad a la que LMC reconoce un gesto específico. Esta velocidad, puede ser usada para estimar la frecuencia de trabajo del LMC. Y la hemos definido como:

$$S_g = \frac{F_g}{T_g} \quad (2.5)$$

donde F_g es el número de frames que definen al gesto y T_g es el tiempo para detectar ese gesto. S_g se expresa en frames por segundo (fps). Es variable y depende del gesto a ser reconocido y se la obtiene dividiendo la frecuencia de muestreo inicial F_i y una tasa de muestreo máxima cuando se realiza un gesto. Además, podemos determinar la velocidad límite del proceso de reconocimiento para cada gesto a través de la tasa de reconocimiento R_g y del tiempo T_g requerido para detectar un gesto. Se expresa en gestos por segundo y se ha definido como:

$$R_g = T_g^{-1} \quad (2.6)$$

El incremento en la tasa de muestreo del LMC se produce aparentemente por la necesidad de lograr el reconocimiento de un gesto y varía en forma automática conforme lo ordena su algoritmo, hasta lograr el reconocimiento.

Efectividad, exactitud y precisión

En este sentido, Steinfeld et al. [174] definieron la **efectividad** como el número y la duración de las intervenciones del operador. Sin embargo, proponemos una definición de efectividad como medida del proceso de reconocimiento cuando, un gesto es reconocido en el menor tiempo posible con el mayor número de frames válidos. Este concepto involucra a dos variables adicionales, que son exactitud y precisión.

Un gesto puede tener alta exactitud y baja precisión o puede tener alta precisión y baja exactitud. Es importante determinar los valores de referencia para la exactitud (AC) y la precisión (PR) de cada gesto con el fin de identificar la validez de estos gestos que se utilizarán en la interacción humano-robot.

Para lograr medir esta variación entre exactitud y precisión, hemos definido una nueva métrica, denominada efectividad, que combina la exactitud y la precisión alcanzadas en el proceso de reconocimiento para cualquier gesto dado. Esta métrica determina cuán aceptable es el proceso de reconocimiento y expresa una garantía para reconocer correctamente el gesto. La hemos expresado como:

$$EF_g = (AC \times PR) \quad (2.7)$$

Donde el valor máximo de EF_g es 1 y esta en relación directa al nivel de exactitud y precisión, evaluados en un rango desde 0 hasta 1 ($[0..,1]$).

Hemos definido a la exactitud (AC) de un gesto como medida de las respuestas positivas y negativas, de entre todas las posibles [182]; Y la precisión (PR) como la calidad, evaluada en relación a las respuestas positivas del proceso de reconocimiento.

Es necesario entonces definir una ventana de tiempo, la cual servirá para este análisis. La hemos ubicado en 120s, de la muestra total lograda en cada gesto.

Estas definiciones, hacen referencia directa al proceso de reconocimiento, el cual ha sido evaluado en relación a:

- Los frames que contienen información y corresponden al gesto (nf_p).
- Los frames que contienen información y no forman parte del gesto (nf_{fp}).
- Los frames que no contienen información y forman parte del gesto (nf_{fn}).
- Los frames que no contienen información y no corresponden al gesto (nf_n).

Por lo tanto, el número de frames disponibles en una ventana de tiempo (nf_T) puede expresarse como:

$$nf_T = nf_p + nf_n + nf_{fn} + nf_{fp} \quad (2.8)$$

Asumiendo que todos los frames no válidos han sido removidos, aplicando un filtro de ruido, tal como la media acotada.

Luego entonces, AC puede definirse como:

$$AC = \frac{nf_p + nf_n}{nf_T} \quad (2.9)$$

Y a PR como:

$$PR = \frac{nf_p}{nf_p + nf_{fp}} \quad (2.10)$$

Gesto de control efectivo

El concepto de Gesto de Control Efectivo (GCE), que se plantea en esta investigación, es establecer una métrica capaz de proporcionar una medida global del desempeño y calidad del controlador LMC, u otro dispositivo óptico, con respecto al proceso de reconocimiento de un gesto específico, pero aplicado a un problema de control reactivo. En control reactivo, se desea que una orden sea aceptada en el menor tiempo posible y con la mayor exactitud [174]. En consecuencia, sólo los gestos que sean catalogados como GCE tendrán garantía de reconocimiento y acción inmediata. También, tendrá una alta probabilidad de ser aceptado y ejecutado dentro de los parámetros temporales de un sistema de control, en cuyo ámbito un delay máximo de $300ms$ es admisible [180, 32]. Un GCE, podrá ser considerado como una recomendación válida en interacción humano-máquina.

Por todo lo expuesto, definimos a un GCE como aquel gesto que debe satisfacer las siguientes condiciones:

- Su efectividad debe ser mayor o igual a 0,5 ($EF_g \geq 0,5$).
- Su tasa de reconocimiento promedio debe ser mayor o igual a 25 gestos/segundo (g/s).

El nivel mínimo de efectividad determina el límite, para la efectividad de un gesto. Esto, es requerido para distinguir la capacidad del controlador para un reconocimiento óptimo, y designar a un gesto como “aceptable.” “no aceptable”. Bajo el enfoque de gestos orientados al control, el gesto de control no sólo se basa en los datos de reconocimiento gestual del controlador, sino también en la posibilidad de que un comando pueda ejecutarse correctamente en un dispositivo. El comando recibido, por la plataforma robótica, ofrece dos alternativas para controlar un dispositivo a saber: aceptado/ejecutado y rechazado/ignorado.

Por otro lado, el límite de la tasa media de reconocimiento de 25 gestos/segundo (g/s) determina la velocidad mínima que debe tener el controlador, para reducir el posible retraso (delay) en el sistema y que, un usuario pueda percibir entre el reconocimiento gestual y la ejecución del comando correspondiente.

2.4. Resultados y discusión

2.4.1. Proceso de reconocimiento

La Figura 2.16 presenta los gestos desde la perspectiva de ejecución del usuario y la hemos empleado, para lograr una mejor comprensión, siendo en realidad los mismos gestos presentados en la Figura 2.10, los cuales representan el gesto, desde la perspectiva 3D del dispositivo LMC.

Una vez obtenidos los datos, en los experimentos de reconocimiento realizados, ejecutando la biblioteca gestual que se muestra en la Figura 2.16, se procedió a evaluar dicho proceso. Para lograrlo, se aplicaron las métricas detalladas en la sección 2.3.6 a todos los gestos.

La Tabla 2.4 muestra los valores medios de \bar{F}_g , \bar{T}_g , \bar{S}_g y \bar{R}_g , junto a su desviación estándar. Estos datos representan el tamaño medio de frames, el tiempo medio, la frecuencia media de muestreo y la tasa media de reconocimiento, respectivamente. La muestra completa de cada gesto fue utilizada con el fin de minimizar el ruido. A pesar de ello, encontramos algunas valores que podrían considerarse como valores atípicos, porque están muy alejadas de los valores medios y se producen aleatoriamente.

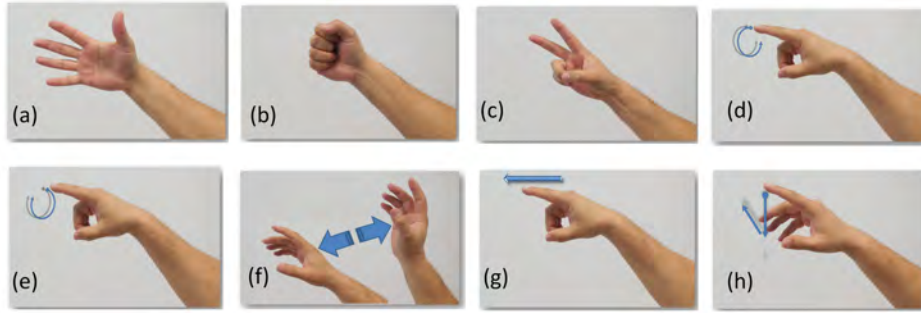


Figura 2.16: Librería gestual completa

Tabla 2.4: Resultados de reconocimiento gestual por tracking

Gesto	\bar{F}_g		\bar{T}_g		\bar{S}_g		\bar{R}_g	
	Media	DS	Media	DS	Media	DS	Media	DS
a	1.88869	2.9181	0.02141	0.03448	96.3292	26.79301	64.41069	30.11737
b	1.16309	2.6933	0.01012	0.02341	114.92936	0.00337	113.71665	10.58692
c	1.46135	4.56961	0.02563	0.08476	62.82063	24.97718	56.43371	25.36467
d	1.63976	21.57163	0.01427	0.18769	114.9346	0.00643	113.70243	10.39264
e	2.61389	7.76275	0.02453	0.08556	114.62314	4.79056	69.9823	30.87502
f	6.05379	17.96521	0.05674	0.18061	114.39564	6.38891	62.50944	35.77506
g	23.76119	57.96882	0.20674	0.50436	114.93462	0.00328	54.88147	32.28344
h	31.675	32.83061	0.29463	0.39502	114.28383	6.87871	11.23558	17.20678

Como puede observarse en la Tabla 2.4 al comparar (\bar{F}_g), algunos gestos pueden ser reconocidos basándose en un pequeño número de frames (gesto a, b, c, d y e) en contraste con otros que requieren una mayor cantidad de frames (gestos f, g y h); logrando incluso, un valor treinta veces mayor como es el caso del gesto (h). Las diferencias en el promedio de frames (\bar{F}_g) pueden observarse gráficamente en la Figura 2.17.a. En consecuencia, la métrica (\bar{T}_g) muestra un comportamiento similar que indica que los gestos con un alto número de frames también requieren un mayor tiempo medio de reconocimiento. La Figura 2.17.b muestra, usando un mapa de calor, las similitudes entre las métricas \bar{F}_g y \bar{T}_g (Los valores fueron normalizados para su representación).

Observando estos resultados, podemos deducir que la naturaleza del gesto (simple o complejo) no tiene una influencia determinante en la dificultad para reconocer el gesto, aunque se puede apreciar una tendencia. De hecho, los gestos simples (a, b y c) fueron reconocidos con tamaños de frame medios y tiempos promedio bajos, y con un valor cercano a $1frame$ y $8,701ms$ respectivamente. Mientras que, los

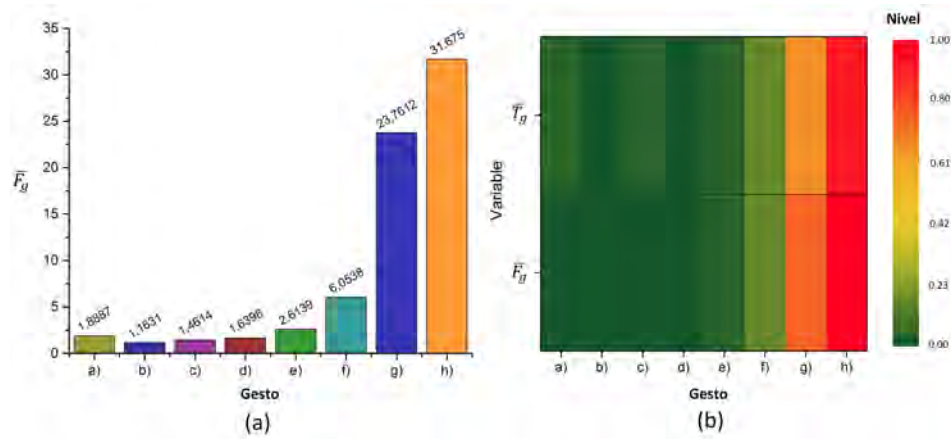


Figura 2.17: Proceso de reconocimiento, estimado en frames y tiempo

gestos complejos, muestran una amplia variabilidad para ambas magnitudes. La desviación estándar de (\bar{F}_g) y (\bar{T}_g) puede darnos información sobre cuán fiable es el proceso de reconocimiento. Los gestos simples, muestran un valor de reconocimiento más estable con una desviación estándar más baja, en contraste con gestos complejos donde las fluctuaciones tienen un impacto importante en el reconocimiento. La tasa de muestreo promedio (\bar{S}_g) obtenida para cada gesto mostró una variación entre 60 y 115 frames por segundo (fps). Variaciones similares entre 50 y 120 fps fueron reportadas por Tung et al. [182].

Podemos observar en la Figura 2.18 que \bar{S}_g intenta mantener una tasa de muestreo promedio alta, para reconocer cualquier gesto tan pronto como sea posible. Sin embargo, en algunos casos, la tasa de reconocimiento promedio disminuye cuando el gesto es más fácil de reconocer, con el fin de reducir la frecuencia de muestreo, aunque esto podría también aumentar su desviación estándar. Este comportamiento fue significativo en los gestos (a) y (c). Además, mantener una alta tasa de muestreo puede sobrecalentar el LMC, como sucedió en nuestras pruebas, y en consecuencia, dañarlo.

Se obtuvo la tasa media de reconocimiento de los gestos (\bar{R}_g) para cada gesto. Como se puede observar en la Tabla 2.4, sólo el gesto (h), cuyo \bar{R}_g es muy bajo, proporcionando una tasa de reconocimiento promedio cuestionable ya que no es posible reconocer estos gestos con una frecuencia de muestreo suficiente para evitar un retraso considerable (delay). En los otros casos, obtuvimos un \bar{R}_g por encima de 50 gestos/segundo con un máximo de 114 gestos/segundo, cerca del valor máximo de la tasa de muestreo del dispositivo. Esta métrica nos puede proporcionar un indicador de la velocidad real del reconocimiento gestual de LMC para un gesto

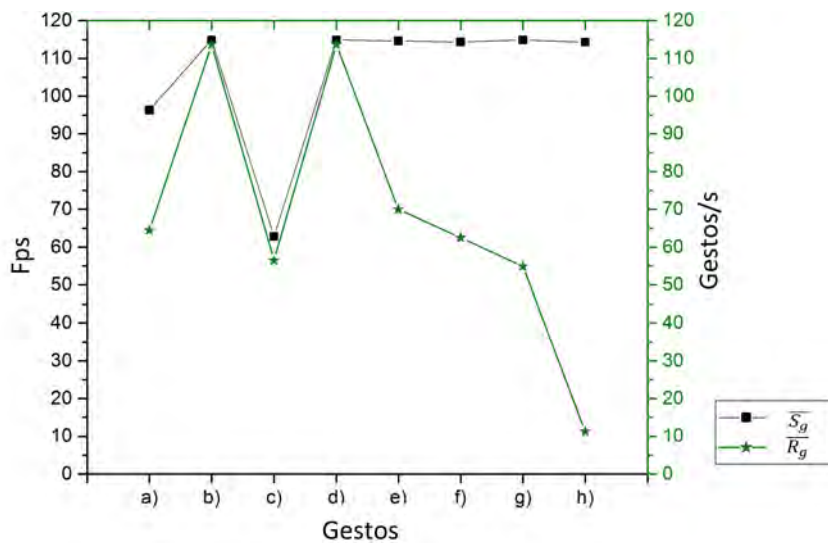


Figura 2.18: Tasa media de muestreo vs tasa media de reconocimiento de LMC

determinado. Esto podría ser una buena alternativa, ya que aún el fabricante de LMC no ha liberado su algoritmo de reconocimiento.

Estas nuevas métricas nos permiten establecer una clasificación de los gestos dependiendo de la dificultad de reconocimiento que muestra el dispositivo (Figura 2.18). La figura posee dos ejes, que indican: en el eje izquierdo (color negro), la gráfica muestra la métrica (\bar{S}_g). El LMC intenta mantener una alta tasa de muestreo promedio para lograr el reconocimiento, aunque en algunos casos, algunos gestos pueden reducir la velocidad de reconocimiento; en el eje derecho (color verde), el gráfico muestra el promedio de la velocidad de reconocimiento del gestos (\bar{R}_g). Excepto para el gesto (h), la velocidad de reconocimiento de los gestos, mantienen una velocidad suficiente para ser utilizada en una aplicación.

2.4.2. Efectividad

Al aplicar el modelo de análisis para calcular la efectividad del reconocimiento gestual, tal como se describió en la sección anterior, se pudo determinar los frames válidos, negativos y falsos positivos en el reconocimiento de cada gesto [182]. No se determinó los falsos negativos, pues no contribuyen al cálculo de la efectividad. La Tabla 2.5 muestra los valores para cada gesto luego del análisis realizado.

Como podemos ver en la Tabla 2.5, existe una estrecha correspondencia entre el número de frames válidos de cada gesto y la tasa de reconocimiento promedio. Para determinar aquellos gestos que cumplen los criterios de EF_g aceptable, las

Tabla 2.5: Reconocimiento gestual estimado en relación a los frames de cada gesto

Gesto	Frames	Validos	Positivos	Falso Positivo	Negativos
a	11528	6332	5269	1063	5196
b	13528	13318	12125	1193	210
c	6720	6597	5023	1574	123
d	12712	5954	3763	2191	6758
e	12752	5365	2478	2887	7387
f	12888	4771	356	4415	8117
g	8136	6198	5486	712	1938
h	13040	7587	96	7491	5453

regiones de aceptación y rechazo se establecieron como se muestran en la Figura 2.19. La exactitud, precisión y efectividad de cada gesto se muestran en la Tabla 2.6 y fueron calculadas de acuerdo a las ecuaciones 2.7, 2.9 y 2.10.

Tabla 2.6: Niveles de exactitud, precisión y efectividad de cada gesto

Métrica	Gestos								Media
	a	b	c	d	e	f	g	h	
Exactitud	0,9078	0,9118	0,9125	0,7658	0,8276	0,7736	0,6574	0,4255	0,7728
Precisión	0,8321	0,9104	0,8851	0,7614	0,6320	0,4619	0,0746	0,0127	0,5713
Efectividad	0,7554	0,8301	0,8077	0,5831	0,5231	0,3573	0,0491	0,0054	0,4889

Los valores de exactitud están por encima del 65 %, excepto para el gesto (h), donde cae al 42 %. La precisión se mantiene por encima del 46 %, excepto para los gestos (g) y (h) que alcanzan el 7 % y 1 %, respectivamente. Aunque la exactitud puede ser aceptable en el gesto (h), su baja precisión no garantiza un reconocimiento de calidad suficiente para usarse en aplicaciones de control. Por esta razón, la efectividad puede proporcionar una mejor información sobre su reconocimiento en lugar de considerar sólo la exactitud o la precisión. De hecho, la efectividad alcanzó buenos resultados por encima del 50 % para casi todos los gestos, excepto para los gestos (f), (g) y (h). Como podemos ver en la Figura 2.19, la efectividad nos permite clasificar el gesto dentro de una de las zonas de aceptación.

2.4.3. Gesto de Control Efectivo

Cuando el gesto se aplica al control, necesitamos comprobar si aquel satisface los requisitos de un GCE. La Tabla 2.19 identifica las dos características que contribuyen a la definición de GCE, una efectividad mayor a cincuenta ($EF_g \geq 50\%$) y

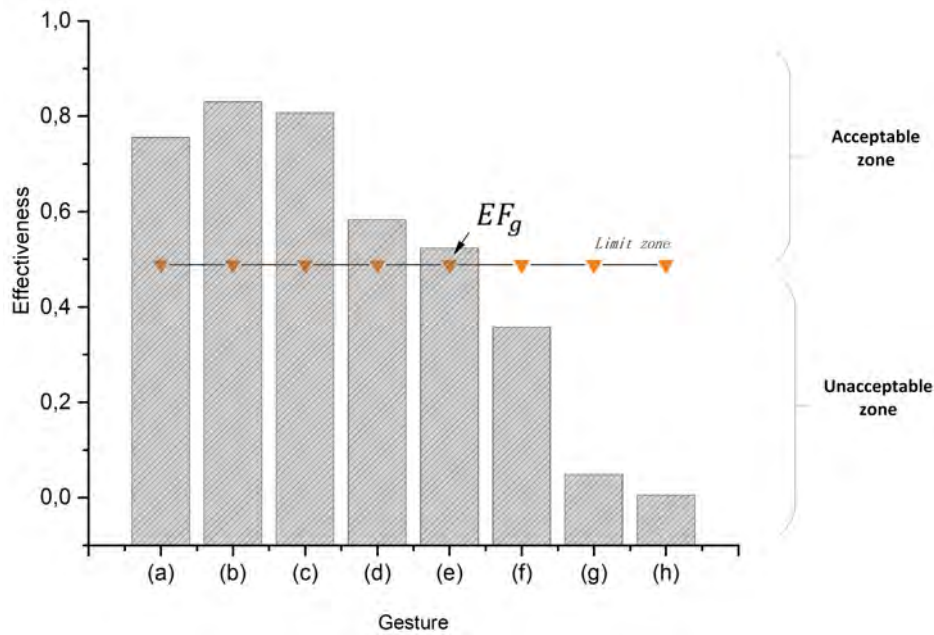


Figura 2.19: Efectividad de los gestos propuestos en la biblioteca gestual de prueba

la tasa de reconocimiento promedio ($\bar{R}_g \geq 25g/s$). En nuestro caso, sólo los últimos tres gestos no pueden ser válidos para ser considerados GCE, porque no cumplen con ambos criterios de evaluación al mismo tiempo. Por lo tanto, los gestos clasificados como GCE, son válidos para las tareas de control, ya que garantizan una eficacia suficiente en el proceso de reconocimiento con una tasa de reconocimiento adecuada. Un conjunto de gestos, con baja efectividad y baja tasa de reconocimiento provocará una pérdida significativa de eficacia en el sistema de control gestual, lo cual puede comprometer el comportamiento del sistema robótico y su interfaz.

Tabla 2.7: Gestos calificados como GCE por niveles de EF_g y \bar{R}_g

Métrica	Gestos							
	a	b	c	d	e	f	g	h
Rg (g/s)	64,4107	113,7167	56,4337	113,7024	69,9823	62,5094	54,8815	11,2356
EF_g (%)	75,54	83,01	80,77	58,31	52,31	35,73	4,91	0,54
GCE	SI	SI	SI	SI	SI	NO	NO	NO

2.5. Propuesta experimental

El análisis presentado, nos permite proponer una aplicación usando una interfaz de tipo gestual, basada en gestos de la mano del usuario que usa a Leap Motion como dispositivo de captura gestual. Dicha interfaz deberá aplicar los principios de efectividad del gesto, propuestos anteriormente y verificar la validez en la interacción en un sistema robótico.

2.5.1. Robot

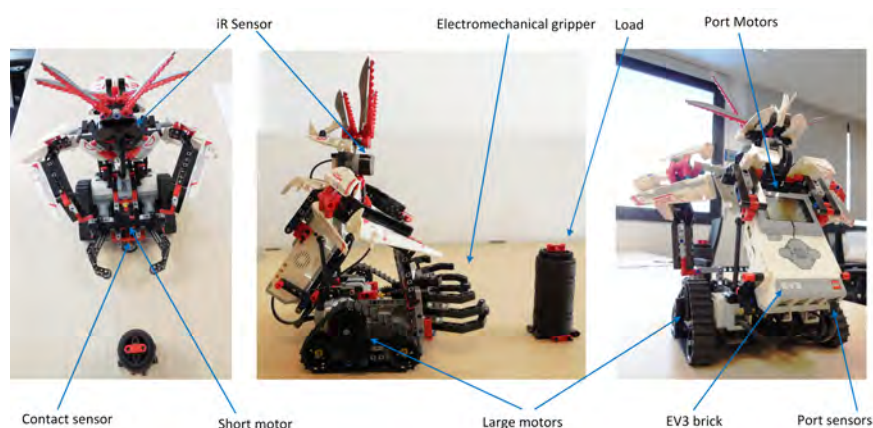


Figura 2.20: Sistema robótico EV3 GRIPP3R y sus componentes.

Se seleccionó al sistema robótico Lego EV3 Mindstorms, ensamblado en el su modelo GRIPP3R. Dicho modelo implementa una pinza electromecánica (Figura XX6). Este modelo fue elegido, porque las pinzas son las más utilizadas en la industria y la medicina [46]. El sistema robótico presentó las siguientes características: Microprocesador ARM-9 a 300 Mhz. 64 MB de RAM, 16 MB de memoria Flash expansible con tarjeta SD, Bluetooth 2.1, USB para conexión WiFi, 4 puertos para sensores y 4 puertos para motores, monitor LCD y altavoces. También tiene un sensor de contacto y un sensor infrarrojo para medir la proximidad. El sistema es compatible con iOS y Android.

2.5.2. Arquitectura del sistema

Para esta implementación, hemos propuesto la arquitectura que se muestra en la Figura 2.21. Dicha arquitectura esta compuesta de tres capas y puede ser usada en aplicaciones similares.

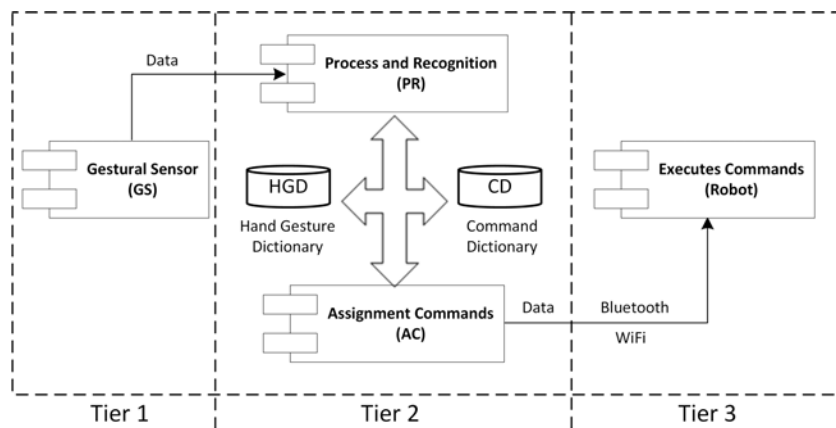


Figura 2.21: Arquitectura propuesta para la implementación del sistema de control gestual para dispositivos robóticos.

- Capa 1: Incluye el LMC como sensor de captura de gestos (GS), encargado de capturar los frames correspondientes a los gestos realizados por el usuario. Se conecta mediante USB a una computadora en la capa 2, la cual recibe los datos enviados para su procesamiento.
- Capa 2: Es el controlador central del sistema. Es donde se realiza el reconocimiento gestual y se preparan y ejecutan los comandos correspondientes, los cuales serán enviados al sistema robótico. El software ha sido desarrollado en Java y lo hemos llamado LEAPEV3. LEAPEV3, integra el sistema de reconocimiento gestual diseñado y las bibliotecas necesarias para enviar los comandos a la unidad robótica. Está compuesto por: el componente de Proceso y Reconocimiento (PR) y además por el componente de Asignación de Comandos (AC). Estos componentes son asistidos por dos diccionarios, uno dedicado a los gestos implementados (HGD) y el otro, a los comandos (CD).
 - Componente PR: Este componente aplica los algoritmos de reconocimiento a los datos obtenidos de LMC los cuales permiten determinar un gesto en base a sus características. Las características gestuales han sido implementadas previamente en patrones gestuales disponibles.
 - Componente AC: Asigna los comandos relacionados a cada gesto, conforme sea identificado por el PR. Cada gesto está asociado con un comando preestablecido y que, está configurado en el diccionario de comandos (Tabla 2.8). Posteriormente, el robot ejecutará el comando. El usuario puede acceder al CD para definir las posibles relaciones entre comandos

y acciones. Esta capa además, es responsable de mantener la comunicación con la Capa 3, utilizando un protocolo de comunicación que puede ser Bluetooth o WiFi. LEAPEV3, permite replicar en su interfaz gráfica la salida de la consola remota del robot y además, controla los modos de funcionamiento del robot. Los modos de funcionamiento pueden ser por interfaz gráfica o por interfaz gestual. La configuración predeterminada es por interfaz gestual, a menos que el usuario desea el control por GUI.

- Capa 3: En esta capa, el dispositivo robótico, recibe el comando enviado por la capa 2 y lo ejecuta. El robot cuenta con movilidad en cualquier dirección (hacia delante, hacia atrás, a la izquierda y a la derecha) y además, puede tomar y soltar objetos gracias a la mano electromecánica con que cuenta. Dicha pinza, posee un sensor de contacto que indica su estado al sistema de control gestual LEAPEV3.

LEAPEV3, permite la comunicación entre el ordenador y el robot por dos vías, esto es por Bluetooth o por WiFi. En este sentido, es el usuario quien decide la forma de conexión. Es necesario indicar que, en nuestras pruebas, la conexión por WiFi mantuvo un gran consumo de energía, restando autonomía operativa al robot. Por esta razón, elegimos a Bluetooth como la conexión por defecto en el sistema, debido a su bajo consumo de energía. De forma predeterminada la opción de consola remota del robot está habilitada, dando una gran ventaja en la administración remota del sistema robótico, pues en ella, se muestra el estado del robot y los parámetros operativos, tales como: el nivel de la batería, el estado de la conexión, etc.

2.5.3. Gestos de Control

Los gestos seleccionados para control en el sistema LEAPEV3, se muestran a continuación en la Tabla 2.8. Fueron tomados de la biblioteca gestual analizada, en base al nivel de efectividad obtenido. Sin embargo, hemos considerado al gesto (f) en la biblioteca de control, para tener un contraste o comparación con los demás y verificar su comportamiento real en control.

2.5.4. Interfaz gestual implementada en LEAPEV3

La interfaz de control gestual implementada en el sistema LEAPEV3 y mostrada en detalle en la Figura 2.22, ha sido diseñada para el control del sistema robótico, a través del gesto. La interfaz gráfica, da la opción al usuario para establecer

Tabla 2.8: Gestos de control efectivo para el sistema LEAPEV3 (gesto (f) de contraste).

Gesto	Comando
Open Hand	Abre la pinza del robot para tomar un objeto. También es el gesto que inicia la interacción
Circle	En sus dos dimensiones, este comando provoca el giro a la derecha o giro a la izquierda.
Key tap	Comando de avance. El robot se mueve linealmente hacia adelante.
Close Hand	Cierra la pinza del robot. Se usa para tomar un objeto.
Two	Comando de retorno lineal o reversa.

conexión con el robot de dos maneras diferentes (Bluetooth y WiFi). Por defecto la conexión Bluetooth esta activada. Sin embargo, también permite al usuario el control manual, por así decirlo, usando la interfaz gráfica, en el momento que sea necesario.



Figura 2.22: Descripción de LEAPEV3 y su operación gestual.

La interfaz de control gestual fue desarrollada en Java y en ella se combina la detección del gesto por medio de Leap Motion y la operatividad de EV3, en una misma aplicación. También, es capaz de mostrar el gesto realizado por el usuario en pantalla en todo momento. Conforme las políticas de seguridad expuestas anteriormente, el sistema LEAPEV3 ejecuta inmediatamente la rutina de parada (STOP) de todos los motores, cuando no se detecta la mano del operador. Esto

sería de gran utilidad en un sistema real para garantizar la seguridad, siguiendo las mismas políticas de control que un robot usando una consola de operación. El estado STOP, se mantiene hasta que se detecta un gesto de inicio o de apertura de la interacción, que corresponde en este caso al gesto (a). Las acciones continúan inmediatamente después de iniciar la interacción, con los comandos que el usuario realice.

2.5.5. Experimentos con el prototipo

Los experimentos realizados con el prototipo LEAPEV3, funcionaron como se esperaba (Figura 2.23). LEAPEV3 fue operativo, tanto en modo automático como manual sin presentar problema alguno. El GCF fue reconocido y ejecutado correctamente por el robot. Se realizaron tareas de carga y descarga, transportando objetos. Los movimientos realizados por el robot respondieron a cada uno de los comandos realizados, respondiendo con precisión a comandos de avance, retroceso, y giro (izquierda y derecha) en forma inmediata. El movimiento era mantenido mientras el gesto permanecía, a excepción de los comandos de carga y descarga que verificaban el estado del sensor de contacto para ejecutarlo.

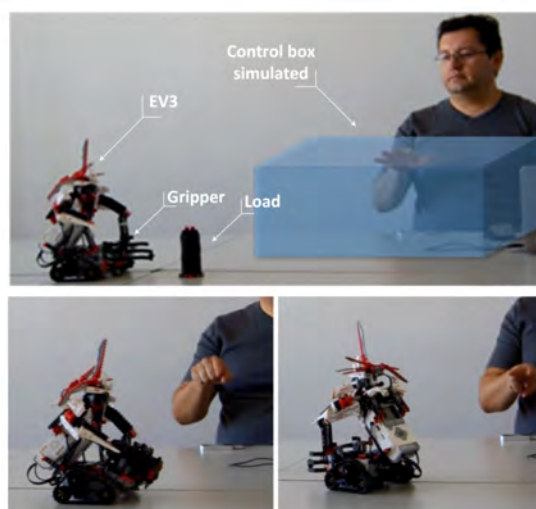


Figura 2.23: Sistema LEAPEV3 en plena operación (modo automático gestual).

Conforme a lo esperado, el gesto (f) llamado gesto de contraste, no era preciso en su detección y ocasionalmente causaba disturbios en el comportamiento del robot, además de problemas saturación del canal de datos. Esto posiblemente, a que dicho gesto causaba retraso en la detección y saturación del buffer de envío-

recepción. Ante problemas de este tipo, el robot detenía su operación. Todo lo mencionado anteriormente, nos permitió validar los resultados obtenidos en la Sección 2.4 para un gesto de control que cumple criterios de efectividad, y a los cuales hemos llamados Gestos de Control Efectivo (GCF), por estar dados en términos de exactitud y precisión.

Finalmente, observamos un efecto perturbador de ambientes con iluminación fluorescente o expuestos directamente a la luz solar, causaban en el LMC. Esto debido quizá a las longitudes de onda del espectro infrarrojo que dichas fuentes producen y que podrían interferir con los componentes de detección de LMC. Para evitar aquel efecto, definimos un área (25cm de altura x 50cm de ancho) al cual hemos llamado Caja de Control (Control Box) y corresponde a la zona de interacción, en donde el LMC no percibe la contaminación lumínica y puede asegurar una mejor respuesta de detección. La caja de control, proporciona el espacio suficiente para tareas de interacción logrando un movimiento fluido de la mano del usuario. Por último, hemos llamado Consola de Administración Gestual (Gesture Administration Console, GAC) al conjunto formado por la caja de control y LMC. La Figura 2.22 muestra la caja de control y el conjunto GAC.

2.6. Conclusiones

En este capítulo hemos realizado un análisis del reconocimiento gestual usando la técnica de tracking. Se han analizado las características principales de detección en medios ópticos y, en este sentido, se ha revisado una amplia bibliografía científica, que ha permitido comprender su aplicación y los enfoques de reconocimiento usados (reglas y técnicas de ML). Hemos analizado en profundidad al dispositivo Leap Motion, el cual combina tecnología óptica e infrarroja, en la detección del gesto. Hemos determinado las facilidades y gran precisión del dispositivo, en la detección de pequeños movimientos y las facilidades que presta para el reconocimiento gestual de la mano del usuario. Con estos antecedentes, se ha realizado un detallado y profundo estudio del dispositivo y las características empleadas por su API para lograr el reconocimiento gestual de la mano del usuario. Hemos combinado los gestos preestablecidos con gestos implementados de nuestra autoría, para obtener una biblioteca gestual, conformada por gestos simples y complejos, que ha sido sometida a evaluación usando métricas propuestas en este capítulo. En este sentido, hemos podido establecer métricas que permiten determinar la dificultad de reconocimiento del gesto usando Leap Motion. Tales como: la tasa de reconoci-

miento (S_g), los frames que logran definir al gesto (F_g), el tiempo de detección del gesto (T_g) y la tasa de reconocimiento (R_g). La métrica tasa de reconocimiento, nos proporciona una medida real de la velocidad de reconocimiento del gesto en LMC, para un gesto concreto. Lo cual, de alguna manera, constituye una forma de evaluación del algoritmo utilizado en el reconocimiento por su fabricante. Además, proponemos un nuevo concepto, basado en la efectividad del gesto, al cual hemos llamado Gesto de Control Efectivo, el cual garantiza un reconocimiento adecuado, dentro de los parámetros temporales requeridos. Finalmente, en base a nuestro prototipo LEAPEV3 y sus excelentes resultados, también podemos decir que los gestos utilizados para las tareas de control debe ser fáciles de realizar, cómodos de mantener (incluso durante largos períodos de tiempo) y fácil de recordar [142, 151]. En este sentido, debe tenerse en cuenta que cualquier labor de automatización no sólo suplanta al humano, sino que transforma la actividad humana relacionada con el control e imponen nuevas demandas de coordinación al operador humano [174].

Este estudio ha dado lugar a la presentación de artículos científicos, cuyos detalles se indican en las conclusiones generales.

Capítulo 3

Análisis de la Interfaz gestual usando sensores (señales EMG)

El profundo análisis del gesto, expuesto en el Capítulo 1, logró identificar al gesto desde varios puntos de vista, tales como: el reconocimiento gestual por tracking, reconocimiento gestual por sensores y el reconocimiento gestual por contacto con superficies táctiles. Ahora, nos corresponde estudiar y analizar el reconocimiento gestual usando sensores y aplicarlos como comandos en un sistema de interacción gestual aplicado a control.

El uso de sensores en la actualidad es más común de lo que aparenta y están en infinidad de dispositivos de uso diario. A diferencia de los sensores ópticos, otros sensores, dependen del contacto directo con el usuario para poder reconocer o medir una variable e inclusive realizar una función, por eso se los denomina wearables o portables. Varios wearables han sido introducidos al mercado, tales como: smartwatch, pulseras de ejercicios, auriculares, ropa, etc. y actualmente alcanzan millones de unidades. En este sentido, solo en el cuarto trimestre del 2016 se vendieron 33.9 millones de unidades [159]. Ante este panorama, es necesario revisar el alcance de los dispositivos wearables que logran capturar el gesto de la mano del usuario.

El auge de estos nuevos dispositivos en el mercado tecnológico, obliga a crear nuevas formas de tratamiento de la información, dinámica y en tiempo real. Pero antes, estos dispositivos deben ser probados para determinar su estabilidad y confiabilidad, el nivel de reconocimiento e interpretación, su grado de adaptación, definición y configuración para que puedan ser utilizados como parte de una interfaz que cubra las necesidades, expectativas y requerimientos de un sistema de

interacción. Se busca además, flexibilidad y facilidad de integración con diversas plataformas y sistemas existentes. Por ello, es necesario validar dichos dispositivos en tareas de captura, procesamiento y reconocimiento gestual, así como implementar bibliotecas gestuales de acuerdo a las necesidades del usuario o sistema, inclusive dando la posibilidad de personalización.

En este capítulo, hacemos un análisis de la interfaz gestual de usuario empleando sensores, concretamente usando la pulsera gestual MYO [®]. Este wearable, logra la captura de señales electromiográficas (EMG) a través de sensores superficiales EMG para conseguir la captura del gesto. Implementamos un sistema de reconocimiento gestual, que separa al wearable del algoritmo de reconocimiento provisto por su fabricante, basado en técnicas de machine learning. En este análisis, se emplea una amplia biblioteca gestual, compuesta de trece gestos estáticos, los cuales fueron efectuados por veintiún voluntarios que colaboraron en esta experiencia. Dos estudios han sido efectuados: el primero, evaluando dos algoritmos de aprendizaje supervisado, a saber k -NN (k -Nearest Neighbour) y SVM (Support Vector Machines), para determinar aquel, que logre una óptima clasificación; y el segundo, implementando un sistema de reconocimiento gestual, basado en el clasificador de mejor rendimiento, aplicado al control de dispositivos de casa el cual se ejecuta sobre un sistema empotrado. Esta característica del sistema de reconocimiento gestual implementado, independiza al dispositivo de la necesidad de uso del ordenador. Por consiguiente, esto abre las puertas a un sin número de aplicaciones, del gesto y del reconocimiento gestual, en la interacción natural de usuario aplicada al control de dispositivos de naturaleza heterogénea.

3.1. Las bioseñales

Varios trabajos científicos tratan de demostrar la validez en la Interacción Humano-Máquina (Human Machine Interaction, HMI) usando señales propias de un organismo vivo, como son las bioseñales o también llamadas señales biomédicas [7].

Una señal biomédica, transmite información de aquel componente orgánico que la generó. La captura de una señal biomédica, posibilita obtener información del sistema fisiológicos y que en determinado momento representa la variable de interés. La Señal Mioeléctrica (MES, por sus siglas en inglés) es una señal biomédica que refleja las variaciones eléctricas generadas en los músculos durante su estimulación (contracción-extensión). Su estudio fue iniciado por Francesco Redi por el

año 1666 de acuerdo a los informes históricos existentes, y en ellos, se indica las propiedades altamente especializadas de los músculos de un pez (anguila eléctrica) capaz de generar electricidad [11]. No es sino, hasta mediados de los años 80's en los cuales se agregan nuevos estudios técnicos, logrando el desarrollo de nuevos electrodos, mejorando su tamaño (haciéndolos más pequeños) y agregando además, componentes de amplificación de señal necesarios. Quince años después, las investigaciones lograron un mejor entendimiento de las propiedades de las señales MES en los tejidos musculares [129, 90]. Nació de esta manera, una forma de estudio basada en señales producidas por los músculos, durante su actividad o reposo, a la cual se la llamó electromiografía o electromiograma (EMG), y no era otra cosa que, una técnica de evaluación y registro gráfico, de la actividad eléctrica producida por los músculos esqueléticos [155]. La mayoría de autores consultados, hacen referencia a EMG como, la señal o grupo de señales recibidas producto de la actividad muscular que la origina. Esta señal, esta normalmente dada en función del tiempo y se describe en términos de la amplitud, frecuencia y fase [148].

Tres tipos de tejidos musculares pueden ser identificados en base a su estructura, propiedades de contracción y mecanismos de control, y estos son: Tejido muscular esqueléticos, Tejido muscular blando y Tejido muscular cardíaco [153]. Las MES están aplicadas a músculos esqueléticos, o que están adheridos al hueso, durante la actividad muscular. El sistema nervioso controla la actividad muscular, esto es, tanto la contracción como la relajación de un músculo. Por ello, la señal EMG replica la señal MES, la cual es controlada por el sistema nervioso y depende de las propiedades anatómicas y fisiológicas del músculo.

Al igual que las neuronas, las fibras de la musculatura esquelética generan potenciales de acción cuando son excitadas por las neuronas motoras, a través de los terminales de las placas motoras [38] (Figura 3.1). Sin embargo, no transmiten los potenciales de acción a otras fibras musculares ni a ninguna neurona. El potencial de acción, de una fibra muscular individual, es aproximadamente de la misma magnitud que el de una neurona, y no esta relacionado forzosamente con la intensidad de contracción de la fibra. La medida de estos potenciales de acción logrados directamente en el músculo o en la superficie del cuerpo (piel), constituye el electromiograma¹. Esta relación trata de ser ilustrada por Navarro et al. [124] en la Figura 3.1.

¹<http://www.bioingenieria.edu.ar>, Barea Navarro Rafael, *Tema 5: Electromiografía*. Instrumentación Biomédica (Departamento de Electrónica, Universidad de Alcalá). Consultado el 10 de agosto del 2016.

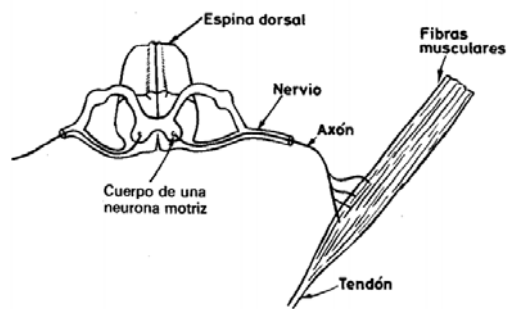


Figura 3.1: Diagrama de una unidad motora [124].

3.1.1. Señales EMG

La señal EMG, es un tipo de bioseñal que ha sido objeto de múltiples y variadas investigaciones, logrando un sinnúmero de aplicaciones. La información que entregan dichas señales, es especialmente útil y ha sido recogida típicamente mediante electrodos bipolares, y sus resultados, aplicados generalmente en el diagnóstico clínico y de rehabilitación. Desde el año 1948 se ha venido investigando su aplicación potencial en el control de prótesis de miembros superiores [201]. Trabajos iniciales pretenden desplazar al *joypad* y *joystick* en juegos de acción, como aquel logrado en *Muscleman* (un juego desarrollado por D.G. Park y H.C. Kim.) [135].

En esta última década, las señales EMG han sido empleadas como fuente de control para dispositivos activos y esquemas de estipulación eléctrica funcional [47]. Han sido usadas para el control clínico de pacientes, aplicando principios de desarrollo de hardware evolutivo (Evolvable hardware, EH)² y aplicaciones orientadas al reemplazo de miembros superiores e inferiores [148, 201]. Entre otros proyectos mucho más ambiciosos, se encuentra aquel planteado por la NASA (California), en el cual se investiga el uso de la señal EMG como una interfaz para sistemas de navegación que responda a los impulsos nerviosos, en tiempo real, del piloto [191]. Para lograrlo, el piloto debería usar un traje equipado con sensores que aprovechan la tecnología EMG y EEG (electroencefalografía), entre otras. Este bio-control de una aeronave, reemplazaría a la pantalla, teclado y *joystick* de mando en un avión de combate o de transporte. Esta y otras ideas, hacen pensar en el control instantáneo de máquinas y equipos, respondiendo directamente a los gestos y pensamientos del usuario [191].

Las diferencias de potencial que conforman la señal EMG, pueden ser voltajes

²Hardware evolutivo se refiere a hardware que puede cambiar su arquitectura y comportamiento de forma dinámica y autónoma al interactuar con su entorno.

positivos o negativos y ocurren a intervalos randómicos [7]. La combinación de los potenciales de las fibras musculares son también llamados *potenciales de acción*. Dado que, la mayoría de las medidas EMG se encaminan a lograr una idea, de la cantidad de actividad que tiene un músculo determinado o un grupo de músculos, en lugar de una fibra muscular individual, las señales generalmente son una suma de los *potenciales de acción* individuales de las fibras, que constituyen el músculo o músculos donde se miden.

Cuando se detecta y registra una señal EMG, existen dos componentes principales que repercuten directamente en la fidelidad de la señal y son:

- La **relación señal-ruido**, la cual se entiende como una porción de señal EMG que tiene un componente adicional, no deseado, pero que forma parte de ella. En general, podemos definir al ruido como una señal eléctrica que no es parte de la señal EMG deseada.
- La **distorsión de la señal**, que significa la contribución de un componente de frecuencia en la señal EMG, la cual no debería alterar a la original de entrada.

3.2. Detección y acondicionamiento de la señal EMG

3.2.1. Capturar señales EMG usando electrodos

Las señales EMG pueden ser capturadas usando electrodos (capacitivos o resistivos), por lo general bipolares que permiten registrarlas con un rango muy variado, que va desde los μV hasta los mV . Los electrodos EMG recogen los potenciales de todos los músculos dentro de su alcance. Esto significa que, los potenciales de músculos grandes cercanos, pueden interferir con los intentos de medir el EMG de músculos pequeños, aun cuando, los electrodos se coloquen directamente sobre dichos músculos. Cuando esto es un problema, hacen falta electrodos insertados directamente en el músculo (electrodos de aguja o intramusculares). Por lo expuesto, los electrodos EMG pueden ser de dos tipos: Intramusculares (iEMG) y Superficiales (sEMG), cuyas características se describen a continuación:

- **Intramusculares (iEMG)**. Permiten la lectura de las señales MES a través de agujas insertadas en el músculo en el cual se desea medir la diferencia de potencial o controlar su actividad. Por lo general, requieren anestesia y

posterior reposo del sujeto luego de su uso. Este métodos es considerado invasivo³.

- **Superficiales (sEMG).** Este tipo de electrodo, se coloca sobre la piel y directamente sobre el músculo del cual se requiere medir su actividad. Por lo general, es necesaria la aplicación de gel para bajar la alta impedancia de la piel. No requiere anestesia y es el más usado, por ser considerado no invasivo.

Hargrove et al. [64] realizaron un trabajo de investigación en el cual compara la captura de señales EMG usando estos dos tipos de electrodos y pudo determinar que no existen diferencias significativas entre las señales obtenidas de modo intramuscular de aquellas obtenidas de modo superficial, lográndose hasta un 97% de exactitud con los de tipo superficial. Estos resultados, garantizan nuestro estudio, al usar en el mismo electrodos de tipo sEMG.

La combinación de los *potenciales de acción* de todas las fibras de un músculo de una unidad motora simple, son llamados unidades motoras potenciales (MUAP) que pueden ser detectadas ya sea con electrodos sEMG o iEMG [129].

La ecuación 3.1 es usada para generar un modelo de señal EMG:

$$x(n) = \sum_{r=0}^{N-1} h(r) e(n-r) + w(n) \quad (3.1)$$

donde $x(n)$, modela la señal EMG, $e(n)$ representa el impulso de disparo, $h(r)$, representa la MUAP, $w(n)$ la media aditiva por cero del ruido blanco Gaussiano y N es el número de unidades motoras de disparo [148].

Hug [71] recomienda tomar en consideración otras variables, que afectan a la coordinación muscular y que, están presentes al usar un sEMG, tales como: diafonía (crosstalk, en inglés), variabilidad espacial de ubicación del electrodo y la frecuencia de corte, utilizada para alisar la envolvente lineal. Esto es debido a la no linealidad de las propiedades mecánicas de los tejidos biológicos [55]. Pero, esto se salió de nuestro objeto de estudio, por lo cual recomendamos leer a [60] para mayores detalles.

En la Figura 3.2, muestra una señal típica capturada por un par de electrodos sEMG. Los electrodos han sido ubicados en el bíceps, para poder captar la señal EMG al momento de realizar dos acciones concretas: la contracción y la relajación

³<https://medlineplus.gov>, MedlinePlus. *Biblioteca Nacional de Medicina de los EE.UU.* Consultado el 8 de agosto del 2016.

del músculo. Nótese la perturbación que ocurre al momento de la contracción y la disminución de las señales al momento de relajar el músculo [184]. Esta característica de la señal EMG que refleja el movimiento muscular, es la que justamente da soporte a nuestro análisis de captura del gesto usando sensores sEMG.

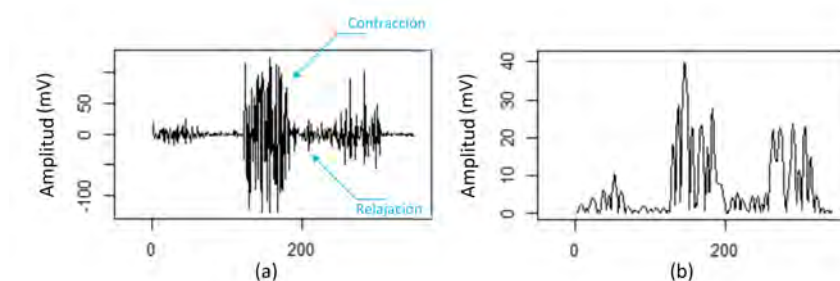


Figura 3.2: Raw de la señal EMG. (a) Señal EMG medida en el antebrazo usando sEMG. (b) Señal rectificada.

Características del electrodo

A continuación en la Tabla 3.1 resume algunos aspectos analizados por Hermens et al. [67] al revisar 144 trabajos científicos, en los cuales se usaron electros de tipo sEMG. En la tabla, hemos plasmado los aspectos más relevantes que conciernen a esta tesis, tales como configuración, material y forma de los electrodos.

Tabla 3.1: Publicaciones que indican aspectos de los electrodos sEMG.

Aspecto analizado	Parámetros verificado	Publicaciones
Configuración del sensor	Monopolar	15
	Bipolar	115
	Linea de electrodos	6
Material	Ag/ClAg	57
	Ag	11
	ClAg	6
	Au	3
	Otros materiales	5
Forma y tamaño del electrodo	Circular	59
	Rectangular	13
	Cuadrado	2
	Ovalado	1

Según, Raez et al. [148] el valor máximo de calidad de una señal EMG puede ser alcanzado siguiendo uno de estos dos caminos:

- La **relación señal-ruido** debería contener una alta cantidad de información de la señal EMG como sea posible y una mínima cantidad de contaminación por ruido.
- La **distorsión de la señal EMG** debe ser tan mínima como sea posible, evitar la distorsión por picos de la señal y el uso de filtros de corte no son recomendables.

La detección precisa de eventos discretos registrados por los sEMG (como el cambio de fase en una actividad asociada a patrones con el inicio de una rápida respuesta motora) es un importante concepto en el análisis del sistema motoro, que involucra el realizar un movimiento o gesto con la mano. Varios métodos han sido propuestos para detectar exactamente el inicio y fin de la actividad muscular. El método más comúnmente usado consiste en la inspección visual por parte de un observador entrenado (médico, fisioterapeuta, etc.).

En la bibliografía revisada, se establece ampliamente que la señal EMG medida con electrodos bipolares durante contracciones de fuerza constante y sin fatiga, puede ser correctamente modelada (usando la ecuación 3.1), como un proceso estocástico de media cero, con distribución Gaussiana, limitado en ancho de banda, estacionario en sentido amplio y contaminado con ruido blanco Gaussiano aditivo [30, 39]. Es por todo ello, que el procesamiento de la señal EMG requiere un conjunto de pasos y métodos que deben ser descritos y aplicados para lograr una identificación adecuada del gesto.

3.2.2. Amplificar la señal EMG

Para poder usar la señal EMG recogida por el electrodo, ésta debe ser amplificada. En la primera fase de amplificación se usa un amplificador operacional (operational amplifier, op-amp)⁴, pudiendo necesitarse múltiples amplificadores para que la señal pueda ser procesada, eliminando los componentes de ruido de baja y alta frecuencia [77]. En ocasiones, es necesario conocer la amplitud de la señal, por lo cual, la señal es frecuentemente rectificadora y promediada, en escalas de amplitud, indicadas para EMG [60]. En la Figura 3.3 se muestra una vista del del circuito integrado que contiene un amplificador operacional cuádruple, TL084CN de propósito

⁴Un dispositivo amplificador electrónico de alta ganancia acoplado en corriente continua, que tiene dos entradas y una salida. Su salida es generalmente muchas veces mayor que la diferencia de potencial entre sus entradas.

general cuyo fabricante es STMicroelectronics (Figura 3.3.a) y el diagrama típico de un amplificador operacional (Figura 3.3.b).

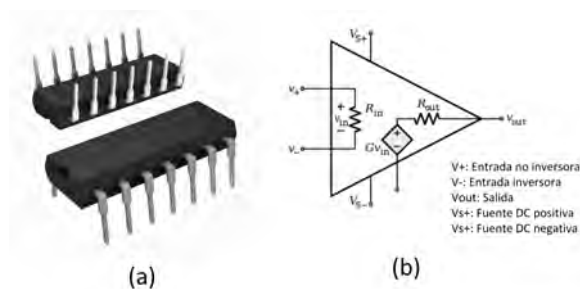


Figura 3.3: Amplificador operacional de propósito general típico.

3.2.3. Medir las señales EMG

Se pueden registrar los potenciales de acción de las fibras musculares en su conjunto o en forma individual. La señal entonces, es la suma de todos los potenciales de acción, dentro del campo de acción de los electrodos, ponderado cada uno por su distancia a los mismos. Dado que, la intensidad global de contracción muscular depende de la cantidad de fibras activadas y del tiempo de contracción, se produce una correlación entre la cantidad global de actividad EMG del músculo y la intensidad de la contracción muscular [161].

La señal EMG se puede cuantificar de varias maneras. El método más simple es medir solo la amplitud, en cuyo caso, se registra la amplitud máxima lograda con un determinado tipo de actividad muscular. Lamentablemente, la amplitud es solo una indicación general de la magnitud de la actividad muscular, y depende de la ubicación del electrodo, respecto al músculo.

El método de cuantificación de EMG más significativo, utiliza la integral temporal de la señal EMG. Con esta técnica, se mide, registra o dibuja, el valor de EMG, integrado en un intervalo de tiempo determinado (por ejemplo, 0, 1s). Tal como se ha indicado anteriormente, esta integral temporal, tiene una relación lineal con la tensión de un músculo bajo ciertas condiciones de contracción *isométrica* (el músculo ni se acorta ni se alarga, pero genera una tensión; por ejemplo, cuando cargamos un peso por un tiempo dado), y también, relación con la actividad de un músculo bajo contracción *isotónica* (acortamiento o alargamiento de las fibras musculares; por ejemplo, actividad deportiva).

Al igual que, en la medida de la amplitud, el valor de la integral de la EMG se ve afectada en gran medida por la ubicación de los electrodos. Si se la hace

correctamente, los valores dan una buena indicación de la actividad muscular. De hecho, bajo ciertas condiciones de contracción isométrica, la integral tensión-tiempo de la señal EMG, tiene una relación lineal con la tensión isométrica voluntaria en un músculo. Cabe mencionar, que hay también formas de EMG características asociadas a condiciones especiales, tales como la fatiga y el temblor.

Los potenciales EMG, de un músculo o de un grupo de músculos, producen una señal parecida a un ruido, muy similar a la que se muestra en la Figura 3.4. Dicha señal, fue captada al realizar un movimiento gestual de la mano, usando electrodos sEMG en el antebrazo del usuario.

La amplitud varía de acuerdo a la magnitud de la actividad muscular, que va desde los μV hasta los mV , pero según Reaz et al.[148] se la ubica entre 0-10 mV (+5V a -5V) antes de la amplificación. Depende en muchos casos, de la ubicación de los electrodos con respecto al músculo y de la actividad de éste. Para una reproducción fiel se requiere una respuesta en frecuencia que puede variar desde los 10 Hz hasta unos 3.000 Hz conforme el Teorema de Nyquist ⁵.

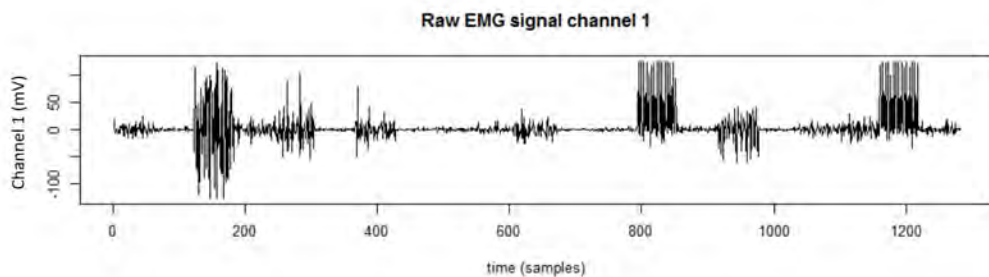


Figura 3.4: Raw de la señal EMG

Tasas de muestreo, superiores a 3.000 Hz , han sido usadas para lograr una reconstrucción fiel de la señal. Pero surge la interrogante sobre **¿a que tasa de muestreo mínima se debe trabajar para no producir una pérdida de información importante?**. Revisada la literatura científica, en la cual se establezca una frecuencia mínima de tasa de muestreo que garantice una representación adecuada de la señal, encontramos que Li et al. [101] determinaron que al reducir la tasa de muestreo a 500 Hz solo se produce un 0.8% de pérdida de exactitud en relación a un 98.0% a 1 kHz . En el mismo estudio se pudo comprobar que, usando

⁵Teorema propuesto por Nyquist-Shanon, en el cual se demuestra que la reconstrucción exacta de una señal periódica continua en banda base a partir de sus muestras, es matemáticamente posible si la señal está limitada en banda y la tasa de muestreo es superior al doble de su ancho de banda. Propuesto por Nyquist 1928 y demostrado por Shanon en 1949.

una tasa de muestreo de 400Hz la media de la exactitud desciende un 1.3% ($96.7\% \pm 1,5$) y a 300Hz desciende un 2.3% ($95.7\% \pm 1,8$). Este estudio nos permite establecer de alguna manera, una referencia de la tasa de muestreo mínima, que los dispositivos wearable pueden usar, junto a la exactitud que puede ser lograda.

3.2.4. Problemas al medir señales EMG

Los conceptos expresados anteriormente nos indican que, es necesario asociar el comportamiento humano con el instrumento de medida y, obtener así, medidas para aspectos particulares de su actividad. Por ello, se deben considerar también los riesgos, que muchos instrumentos pueden tener, al ser usados directamente en el cuerpo. Por ello, existen casos en los cuales una medición no es posible, ya sea por razones fisiológicas o técnicas, que pueden experimentarse. Cromwell et al. [35] determina las siguientes razones:

a) Inaccesibilidad

La inaccesibilidad es la dificultad de acceder a las variables a medir. Esto se da en algunos casos, como aquellos que involucran al cerebro, al tratar de medir la actividad neuroquímica. Al querer medir dicha actividad, usando por ejemplo un transductor, implicaría un alto riesgo de afectación al ser humano o cualquier ser vivo en el cual se lo intente. En consecuencia, se ha optado por medidas indirectas, fundamentadas en otras variables relacionadas. Pero, el investigador debe tener en cuenta constantemente las limitaciones de las variables supletorias y también debe considerar, el poder determinar el momento en el cual es inválida o no dicha relación.

b) Variabilidad de los datos

Las variables que pueden medirse en el organismo humano son de tipo determinístico y deben ser consideradas como estocásticas. Un proceso estocástico, puede definirse como una función temporal relacionada con otras variables de una forma no determinista. Por ello, las variables fisiológicas no se pueden considerar como variables estrictamente determinísticas, sino que, se deben representar por algún tipo de distribución estadística o probabilística. En otras palabras, las medidas realizadas en un instante, bajo un conjunto de condiciones determinado, no serán necesariamente las mismas que aquellas medidas similares realizadas bajo las mismas condiciones, en otro instante. La variabilidad de un individuo a otro es muy alta. Aquí, se deben emplear métodos estadísticos para estimar las relaciones entre

variables. Esta razón, afianza aún más el uso de técnicas de machine learning.

c) Falta de conocimiento de las interrelaciones

La razón anterior, de variabilidad en los valores medidos, se podría explicar mejor si se conociere y entendiere más profundamente las interrelaciones existentes en el organismo vivo. Las medidas fisiológicas, con grandes tolerancias, son aceptadas frecuentemente por un médico por ejemplo, debido a lo escaso del conocimiento de interrelación existente y a la falta de control de las variaciones resultantes. Una mejor comprensión de las relaciones fisiológicas permitiría, además, una utilización más efectiva de las medidas indirectas como sustitutos de medidas inaccesibles y ayudaría al ingeniero o técnico en su labor de acoplar la instrumentación al sistema fisiológico.

d) Interrelaciones entre sistemas fisiológicos

Debido al gran número de lazos de realimentación, incluidos en los principales sistemas fisiológicos, existe un elevado grado de interacción, tanto dentro de un sistema dado, como entre los sistemas principales. El resultado, es que al momento de realizar una estimación de una parte de un sistema dado, por lo general se afecta, de alguna forma, a todas las demás partes de ese sistema (a veces de un modo impredecible) y, a menudo afecta, a otros sistemas. Por esta razón, las relaciones (causa-efecto) se hacen muy poco claras y difíciles de definir. Aún, en el caso de que se abran (rompan) los lazos de re-alimentación, aparecen anillos colaterales y todavía se presentan algunos aspectos del lazo de re-alimentación original. Además, cuando un órgano o elemento se vuelve inactivo, a veces, otro órgano o elemento lleva a cabo su función. Esta situación es cierta especialmente en el cerebro y otros puntos del sistema nervioso de un ser vivo.

e) Efecto del transductor en la medida

El transductor puede alterar la medida, especialmente en sistemas vivos. Un intento de medir los potenciales electroquímicos generados en una célula individual, exige penetrar en la célula por medio de un transductor. Esta penetración puede matar fácilmente a la célula o dañarla de tal modo, que deje de funcionar normalmente. A esto hay que agregar, el efecto que puede causar el transductor sobre otros sistemas que, como ya se dijo anteriormente, al verse afectados pueden alterar la forma de interactuar entre ellos. Un ejemplo podría ser el efecto que causa el temor a la presión arterial alta en un paciente, al realizarse un control que determinará, si es o no hipertenso.

f) Artefactos

En medicina y biología, el termino artefacto se refiere al cualquier componente de una señal que sea extraña a la variable representada por la señal. De este modo, se considera como artefacto: el ruido blanco generado por el instrumento de medida, la interferencia eléctrica (incluyendo la captación de 60 Hz de la red eléctrica), la diafonía (ya explicada anteriormente), y todas las demás variaciones inesperadas de una señal. Una fuente importante de perturbación, al realizar una medición en un sistema vivo, es el movimiento (voluntario o no) del individuo, el cual a su vez produce el movimiento del dispositivo de medida. Dado que muchos transductores son sensibles al movimiento, cualquier variación en el paciente o usuario, produce frecuentemente también variaciones en la señal de salida. En ocasiones, estas variaciones, no se pueden distinguir de la variable medida; y, en otros casos, puede ser suficiente para alterar completamente la información deseada. Por ejemplo, la aplicación de anestesia para reducir el movimiento puede producir cambios inesperados en el sistema.

g) Limitaciones de energía

Muchas técnicas de medida fisiológica necesitan que se aplique una cierta energía al sistema vivo para poder obtener una medida. Por ejemplo, las medidas de resistencia necesitan que paso de corriente eléctrica por los tejidos o la sangre, donde se los mide. Algunos transductores, incluso generan una pequeña cantidad de calor debido al paso de corriente. En la mayoría de los casos, este nivel de energía es tan bajo que su efecto es insignificante. Sin embargo, al tratar con células vivas, se debe tener mucho cuidado para evitar la posibilidad de crear concentraciones de energía que puedan dañar a las células o afectar a las medidas.

h) Aspectos de seguridad

Tal como se ha mencionado anteriormente, los métodos empleados para medir variables en un ser humano vivo, no deben poner en peligro la vida o el funcionamiento normal de los sistemas y órganos del individuo. El hacer énfasis en la seguridad, sugiere que se tome toda precaución necesaria en el diseño de cualquier sistema de medida con el objetivo de proteger al usuario. De un modo parecido, el realizar una medida, no debería producir un dolor, trauma o incomodidad excesivos a menos, que sean necesarias dichas condiciones para salvar la vida del ser humano.

i) Factores internos: Fatiga muscular y estrés

La fatiga muscular según Soo et al. [172], se define como aquella condición en

la cual el músculo ya no es capaz de mantener cierto nivel de fuerza. Cuando la longitud de las fibras musculares aumenta, la velocidad de conducción disminuye. Por ello es importante que las longitudes del músculo sean lo más uniforme posible (contracción isométrica). En contracciones dinámicas se recomienda que el músculo retorne a su longitud original, pues esto podría reducir el efecto de la fatiga. El fenómeno de la fatiga puede alterar las señales EMG, por ello Fernández et al. [80] recomiendan que al realizar pruebas en las cuales se realicen contracciones isométricas, la duración de la contracción no debe ser mayor a 20 segundos. Esto también es aplicable a los gestos.

El estrés juega un papel importante al tomar datos de la actividad muscular. Galen et al. [56] concluyen, que el estrés mental y la tensión muscular están relacionados, que el estrés provoca tensión muscular y, la tensión muscular, provoca estados de estrés. Puede observarse entonces que, al existir tensión muscular, se pueden generar cambios en las señales EMG, las cuales, no necesariamente estarán relacionadas con movimientos voluntarios. Muchos trabajos de investigación han centrado su atención en la búsqueda y determinación de las causas del estrés y como lograr su alivio. Podemos decir entonces que, para lograr lecturas adecuadas, se debe evitar las contracciones dinámicas repetitivas exageradas y que involucren grandes esfuerzos, evitando así la fatiga muscular, que pueden llevar a situaciones de estrés en el individuo.

j) Factores externos: Posición y tamaño de los electrodos

Uno de los puntos más críticos es quizá, la cercanía de los puntos de prueba entre sí y la interferencia generada entre ellos comúnmente llamada “Cross Talk” [173]. El fenómeno de “Cross Talk”, hace que se vean reflejadas señales que no corresponden a los puntos de análisis del músculo. Otro aspecto que perturba la toma de datos en señales EMG, es la presencia de aislantes en la piel, tales como el sudor y el vello corporal, presentes en ella [117]. Para controlar perturbaciones de este tipo, el SENIAM (Surface Electromyography for noninvasive Assessment of Muscles) ⁶ recomienda se tome en cuenta los siguiente:

- *Selección del electrodo EMG.* Los electrodos más usados son aquellos de *Ag/ClAg* (plata/cloruro de plata). Esto se confirma con los datos presentados en la Tabla 3.1.

⁶www.seniam.org. El proyecto SENIAM de electromiografía superficial para la evaluación no invasiva de los músculos (Siglas en inglés Surface ElectroMyoGraphy for the Non-Invasive Assessment of Muscles) es una acción concertada europea en el Programa Biomédico de Salud e Investigación (BIOMED II) de la Unión Europea.

- *Características de los electrodos.* Existen varias características a tomar en cuenta, entre ellas la forma del electrodo y el tamaño del mismo. En el proyecto SENIAM [117] se hace referencia a dos tipos: el circular (Figura 3.5.a) y el rectangular con extremos redondeados (Figura 3.5.b).
- *Distancia entre electrodos.* La distancia entre electrodos se define como aquella distancia existente entre cada uno de los centros de cada una de las áreas conductoras. Las recomendaciones del SENIAM indican que, no puede ser mayor a $1/4$ la longitud de la fibra muscular objeto de estudio. Esta distancia se trata de ilustrar en la Figura 3.5b.
- *Preparación del voluntario.* Según las recomendaciones antes indicadas, se debe preparar la piel del voluntario para la toma de datos de señales EMG y evitar posibles interferencias. Se recomienda limpiar la piel, afeitar la zona y limpiar con alcohol, permitiendo que se vaporice, antes de realizar cualquier toma de muestras. Esto depende en gran manera del material y tipo de electrodo a ser usado en la toma de datos.

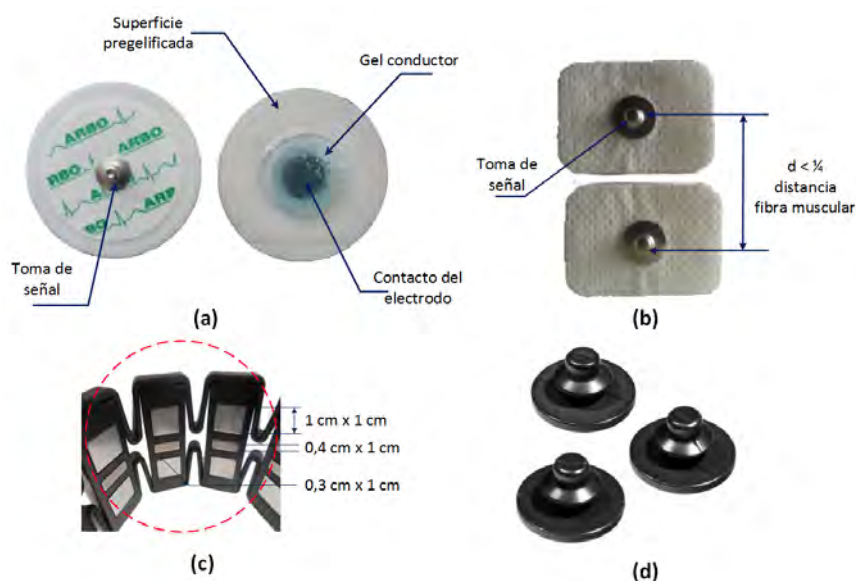


Figura 3.5: Características y tipos de electrodos sEMG.

Sin embargo, en el transcurso de esta investigación pudimos encontrar fabricantes que han lanzado al mercado electrodos de características notables y geometría diferente. Así, por ejemplo, la Figura 3.5.c muestra el sensor sEMG, de acero quirúrgico, incorporado en la pulsera gestual MYO, cuyas características

están disponibles en ⁷, mientras que en la Figura 3.5.d podemos observar electrodos de Ag/ClAg-SA9327 del fabricante Thought Technology Ltd.⁸ para ser usadas con correas de muñeca.

Todos ellos, enfatizan que la superficie de contacto del sensor o electrodo debe estar directamente sobre los músculos, de los cuales se desea tomar información o analizar su señal. El tamaño de la superficie de contacto se conoce con el nombre de *tamaño del electrodo*. En aplicaciones clínicas se observan tamaños entre $1mm^2$ hasta pocos cm^2 . Pero, se recomienda que el tamaño de los electrodos sea de máximo $10mm^2$ ⁹.

3.3. El dispositivo de captura gestual MYO

La idea de alcanzar una comunicación inalámbrica rápida y eficiente, ha sido plasmada en algunos dispositivos wearables, los cuales permiten al usuario independizarse de las ataduras físicas (cables) para envío y recepción de datos. Uno de estos dispositivos es la pulsera gestual MYO [®](Thalmic Labs). MYO incorpora un total de ocho sensores sEMG que capturan, amplifican y filtran la señal EMG para posteriormente, ser convertida a una señal digital. Cuenta además, con una Unidad de Medición Inercial (Inertial Measurement Unit, IMU) de 9 ejes. Posee una tasa de muestreo a 200 Hz y su IMU trabaja a 50 Hz. La Figura 3.6 muestra a MYO, sus componentes y características externas.



Figura 3.6: Pulsera gestual MYO y componentes externos. (a) Distribución de sensores. (b) Dimensiones de los sensores sEMG.

Los sensores sEMG se encuentran distribuidos radialmente en el antebrazo del

⁷<http://blog.thalmic.com/prototype-final-design-evolution-myo-armband/>

⁸<http://www.medicalexpo.es/>

⁹www.seniam.org. Recomendaciones de electromiografía de superficie para Europa, 1999.

usuario y logra la captura de las señales EMG, que se generan en un movimiento asociado al gesto. Esto por las propiedades de la señal EMG revisadas anteriormente. Por su ubicación, en el antebrazo, recoge todas las señales producidas en cada gesto al realizar el movimiento muscular que lo determina. Por lo tanto, garantiza en todo momento, capturar todas las señales EMG que posibiliten la identificación del gesto.

El uso de MYO, no requiere la aplicación de gel, ni limpieza de la piel, y por su geometría, garantiza una lectura adecuada de la señal y reducción del *cross-talk*. Los datos obtenidos por la pulsera son procesados y enviados hacia un dispositivo receptor (host) por medio del protocolo Bluetooth de bajo consumo y alto rendimiento (Bluetooth Low Energy, BLE).

Los sensores sEMG de Myo incorporan, cada uno de ellos, un cuádruple amplificador operacional de alta ganancia. Las características de un dispositivo de este tipo fueron descritas en la Sección 3.1.1). La Figura 3.7 nos permite observar el sensor desde varios puntos de vista.

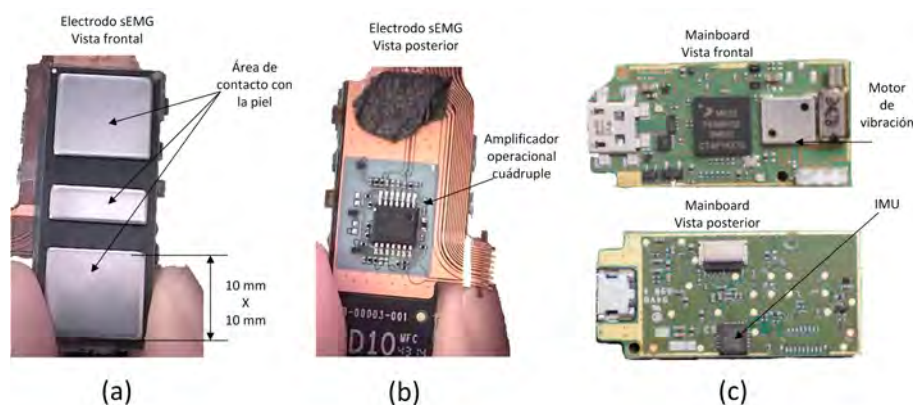


Figura 3.7: Sensor sEMG de MYO desde varias perspectivas.

- La Figura 3.7.a realiza una vista frontal del sensor sEMG y de sus superficies de contacto con la piel. Dichas superficies, son de acero quirúrgico y no requieren limpieza de la piel ni aplicación de gel para su uso. Su geometría es paralela a la fibra muscular, lo cual es muy ventajoso para garantizar la precisión de la medida y evitar el CrossTalk. Sus dimensiones se ajustan a las recomendaciones dadas por el SENIAM (Sección 3.2.4).
- La Figura 3.7.b visualiza la parte posterior interna de uno de los sensores sEMG de la pulsera y, además, en ella se puede apreciar, más claramente, el

amplificador operacional asignado al sensor y sus conexiones.

- Por último la Figura 3.7.c muestra el empotrado, principal componente de procesamiento de la pulsera gestual MYO. El mainboard, soporta un Cortex-M4 a 120 *mHz* e incorpora una unidad de comunicación Bluetooth NRF51822 de bajo consumo (BLE). En esta placa principal, se encuentra también un IMU de 9 ejes (acelerómetro, magnetómetro y giroscopio) de alta precisión, tipo MPU-9150-9-Dof. Se ha destacado también el motor de vibración que dispone de tres niveles (bajo, medio y alto).

Cada sensor esta agrupado en barras paralelas como se aprecia en la Figura 3.6, las cuales rodean el antebrazo del usuario y están separadas entre si por una banda flexible que permite unirlos, los sostiene y cubre los elementos de conexión. La banda Flex, es ajustable y su diámetro va desde los 19 cm (mínimo) hasta 34 cm (máximo) de diámetro. Cada sensor sEMG tiene una altura de 1,143 cm.

El fabricante del ARM Cortex-M4 es Freescale Semiconductor Inc. ¹⁰, el cual detalla las principales características:

- Bajo consumo de energía, lo cual lo hace ideal para trabajo prolongado y procesamiento rápido de datos.
- Es un microprocesador de alto rendimiento y con capacidades de procesamiento de señal.
- Trabaja a 32 bits, con baja potencia dinámica y alta eficiencia energética, debido a los modos integrados de descanso (sleep) controlados.
- Incorpora una unidad de punto flotante (FPU, siglas en inglés) que lo hace mucho más rápido y preciso en operaciones matemáticas de punto flotante.
- Incluye soporte para IP y compatibilidad con protocolo de comunicación Bluetooth de bajo consumo.

Finalmente, en el interior de la pulsera gestual encontramos, dos baterías de Ion-Litio de 260 mA a 3.7 V. En nuestras pruebas iniciales de uso, la pulsera se mantuvo operativa por más de 16 horas de uso continuo (el fabricante garantiza 24 horas).

¹⁰<http://www.nxp.com>

3.4. Características y reconocimiento de patrones

Como hemos mencionado anteriormente, la señal EMG obtenida del usuario al realizar un determinado gesto, encierra patrones ocultos a simple vista, que deben ser descubiertos, para posteriormente, ser aplicados en el reconocimiento gestual. Para alcanzar este objetivo, es necesario realizar varias consideraciones debido a la naturaleza de la señal EMG. Dicho esto, una señal EMG recogida usando electrodos sEMG requiere de una pre-amplificación, necesaria para diferenciar las pequeñas señales de interés. Posteriormente, las señales serán amplificadas, filtradas y digitalizadas usando equipos EMG, especialmente contruidos para ello (por ejemplo, equipo médico NEUROWERK ¹¹) y, en el caso de MYO, finalmente transferidas a un controlador. Todo este proceso ha sido definido por Asghari-Oskoei & Hu [7] y lo detallamos a en las siguientes etapas:

- **Segmentación de datos:** En general la segmentación de datos comprende varias técnicas y métodos, que se usan antes de la extracción de características. Su uso esta justificado para tratar de mejorar la exactitud de clasificación y mejorar el tiempo de respuesta.
- **Extracción de características:** Esta fase es quizá una de las más importantes, pues se encarga de considerar aquellas características intrínsecas, en el Raw (datos en bruto) la señal EMG, que podrán ser empleadas para lograr su adecuada identificación. Básicamente existes tres tipos de selección de características, a saber: en el dominio del Tiempo (Time Domain, TD), en el dominio de la Frecuencia (Frequency Domain, FD) y aquellas en el dominio del tiempo-frecuencia (Time Frequency Domain, TFD). El uso de características, hace que el clasificador incremente su exactitud y por lo tanto disminuya el error de clasificación. La selección de características es llamada también, extracción de características.
- **Clasificación:** Es la fase en la cual, se reconoce los patrones y se los clasifica en categorías predefinidas. Por la complejidad de la señal EMG, ésta fase en ocasiones es influenciada por las condiciones fisiológicas y físicas, por lo cual es recomendable que el clasificador sea lo suficientemente robusto e inteligente. En este sentido, una forma de garantizar el funcionamiento del clasificador a largo plazo, es explotando al máximo la información de entrenamiento offline y online disponible.

¹¹www.abmedic.com

- **Controlador:** Es aquel, que genera los comandos de salida basado en la identificación de patrones y esquemas de control establecidos. Un ejemplo de ello, sería el trabajo presentado por Chen y Wang [27] para reconocimiento de patrones de gestos numéricos, basados en un sEMG de superficie. Otro ejemplo, el trabajo realizado por Wheeler & Jorgensen [192] en el cual se reemplaza al joysticks y teclados por gestos de la mano.

3.4.1. Segmentación de datos

Un segmento, es un intervalo (slot) de tiempo considerado para la extracción de características. De acuerdo a [7], por restricciones de tiempo real, la longitud del segmento adyacente más el tiempo de procesamiento, hasta generar las órdenes de control, debe ser menor o igual a $300ms$. Un segmento demasiado pequeño, degradaría el rendimiento de la clasificación pues el sesgo y la varianza aumentan a medida que disminuye el segmento. Dimensiones de segmentos entre $32ms$ y $250ms$ según Englehart et al. [47] no disminuyen considerablemente la exactitud de clasificación. Sin embargo para su aplicación debe considerarse que una señal EMG comprende dos estados: un **estado transitorio** resultante de una ráfaga de fibras, cuando un músculo va del reposo al nivel de contracción voluntaria; y, un **estado estacionario**, que emana durante una contracción constante mantenida en un músculo.

De acuerdo con Englehart et al. [47] los datos de estado estacionario se clasifican en forma más exacta que los datos transitorios. En longitudes de segmento más cortos, la exactitud no se ve afectada en datos estacionarios, mientras sucede exactamente lo contrario con datos transitorios. **En control, los robots no podrían responder a estados transitorios debido a la existencia de estados indeterminados durante la transición** [7], lo cual confirma nuestro enfoque de uso de EMG orientado a comandos.

Un tercer punto a ser considerado, además de la longitud del segmento y la transición de los datos, es la técnica de **Ventanas de Datos** (Windowing Technique, WT). En la literatura consultada, se han definido dos técnicas de ventana de datos que son: Ventanas adyacentes y Ventanas superpuestas.

- **Ventanas adyacentes**, son aquellos que usan segmentos disjuntos adyacentes con una longitud predefinida, previa a la extracción de características.
- **Ventanas superpuestas**, son aquellas en las cuales un nuevo segmento se desliza sobre el actual, con un tiempo de incremento menor que la longitud

del segmento. Esto podría ser mayor que el tiempo de procesamiento, porque el procesador debe calcular el conjunto de características y generar una decisión, antes de que llegue el siguiente segmento. Farina y Merletti et al [50] descubrieron que segmentos superpuestos simplemente aumentan el tiempo de procesamiento sin mostrar una mejora significativa en la exactitud de clasificación. Y una longitud de segmento inferior a $125ms$ conduce a una alta varianza y sesgo en las características de dominio de la frecuencia.

Un esquema de **clasificación o segmentación continua** fue presentado por Englehart & Hudgins [46], en el cual un flujo denso de decisiones se produce utilizando segmentos superpuestos. La segmentación continua, depende tanto de datos transitorios como estacionarios. Los métodos post-procesamiento, están diseñados para administrar la salida clasificada excesiva y mejorar el rendimiento del sistema. Englehart & Hudgins aplicaron el Voto Mayoritario (MV, siglas en inglés) en el post-procesamiento. Mayores detalles sobre esta técnica pueden encontrarse en [46]. Una conclusión importante que presentan Englehart & Hudgins en este trabajo, es que los resultados obtenidos sugieren que **la exactitud se basa en el tamaño de la muestra, lo cual hace más estable al sistema de clasificación.**

3.4.2. Selección de características

Como hemos dicho anteriormente, la señal EMG posee componentes aleatorios y temporales que, al ser entregados directamente a un clasificador provocarían una baja exactitud. Por lo tanto, es recomendable que la secuencia sea asignada a un vector de dimensiones más pequeñas al cual se llama vector de características. Dicho vector, representa entonces al Raw de datos para la clasificación, por lo que la selección y extracción de características es muy importante. Las características se dividen en tres categorías: en dominio del tiempo, en el dominio de la frecuencia (espectral) y dominio de la escala de tiempo (tiempo-frecuencia). Además, existen dos enfoques para la evaluación de características: enfoques estructurales y enfoques fenomenológicos.

- Enfoques estructurales, son aquellos en los cuales las características se evalúan a partir de modelos físicos y fisiológicos. En este enfoque las características pueden evaluarse utilizando señales sintéticas generadas por modelos matemáticos. En este enfoque se pueden medir el sesgo, la varianza, el nivel de sensibilidad de ruido.

- Enfoque fenomenológico, es aquel que interpreta de forma aproximada la señal estocástica a pesar de su estructura generadora. Las características se evalúan sobre la base de una tasa de desempeño de clasificación y su robustez. También se lo llama enfoque empírico.

Características en el dominio del tiempo

Las técnicas del dominio de tiempo son las más usadas y se basan en la amplitud de la señal. Considerando a la señal EMG como una señal estocástica de media cero, la amplitud puede definirse como la desviación estándar (Standard Deviation, SD) variable en el tiempo de una señal, que es proporcional al número de unidades motoras activas y la velocidad de su activación. La amplitud, indica la energía de la señal, el nivel de activación, la duración y la fuerza. La relación señal-ruido (SNR, siglas en inglés) es una buena opción para juzgar la calidad de las características de la amplitud. A mayor SNR mejores características. Cuando la fuerza o la postura están cambiando, esta ya no es una medida útil. Entre las más usadas tenemos: Valor Medio Absoluto (MAV), integración EMG (Integrated EMG, IEMG), Raíz media cuadrada (Root Mean Square, RMS), cruce por cero (Zero Crossings, ZC), cambios de signo inferior (Slope Sign Changes, SSC), longitud de onda (Waveform Length, WL), Varianza de EMG (VAR), Integral cuadrada simple (Simple Square Integral, SSI), etc.

Características en el dominio de la frecuencia

Este tipo de características, análisis espectral o dominio de la frecuencia, se usan principalmente para estudiar la fatiga muscular e inferir cambios en la Unidad Motora (Motor Unit, MU). El espectro está influenciado por dos factores: la tasa de disparo (Firing Rate, FR) de la MU en el rango de la baja frecuencia (debajo de $40Hz$) y la morfología del potencial de acción que se desplaza a lo largo de una fibra muscular en un rango de alta frecuencia (mayores a $40Hz$). Es una variante del tiempo y depende directamente de la fuerza de contracción, la fatiga muscular y la distancia entre electrodos. **Una señal mioeléctrica se puede suponer estacionaria en aplicaciones en tiempo real, incluso si tiene características espectrales variantes [7].** Entre las características más usadas tenemos: densidad espectral de potencia (Power Spectral Density, PSD), Magnitudes de poder espectrales (Spectral Power Magnitudes, SPM), transformada de Fourier de corto tiempo (Short-time Fourier Transform, STFT), transformada wavelet (Wavelet Transform, WT), transformada de paquete wavelet (Wavelet Packet

Transform, WPT), transformada wavelet continua (Continuous Wavelet Transform, CWT), análisis de componente principal (Principal Component Analysis, PCA), análisis discriminante lineal (Linear Discriminant Analysis, LDA).

Características en el dominio del tiempo-frecuencia

Es la combinación de las características de tiempo y de frecuencia para lograr una mejor clasificación. Un ejemplo de ello, es el trabajo realizado por Vuskovic & Du [185] para clasificación de patrones EMG, en el cual se usó el cuadrado de los patrones EMG sin procesar y luego se los pasó a través de un filtro móvil FIR (Finite Impulse Response filter) con una función de ventana Hamming de tamaño de 300ms. En este trabajo, se demostró empíricamente que la característica aplicada dio una buena tasa de clasificación, pero se señala que un tamaño más grande del segmento podría ser un inconveniente para esta característica. Otros varios trabajos de combinación de características de tiempo y frecuencia han sido realizados, tratando de buscar la mejor clasificación en el menor tiempo.

3.5. Métodos de clasificación

3.5.1. Revisión científica

La clasificación es una etapa donde, las características extraídas deben clasificarse en clases distintas para el reconocimiento de los patrones gestuales deseados. Hay que mencionar, que debido a la naturaleza de las señales EMG es posible obtener grandes variaciones en el valor de una característica en particular. Además, hay factores externos que influyen, tales como: cambios en la posición del electrodo, fatiga y sudor. Todos ellos producen cambios en un patrón de señales con el tiempo. Un clasificador, entonces, debe soportar la variación de manera óptima, para este tipo de señales.

La clasificación debe ser lo suficientemente rápida y exacta para garantizar respuesta en limitaciones de tiempo, como sucede en aplicaciones de tiempo real. Un clasificador debe ser eficiente rápido al clasificar nuevos patrones y el entrenamiento en línea debe mantener un desempeño estable a largo plazo. Por estas razones, muchos trabajos han sido presentados aplicando y combinando diferentes características y dominios, así como múltiples clasificadores.

En este sentido, la eficacia del reconocimiento gestual utilizando señales EMG consigue una exactitud que es altamente competitiva en comparación con otras tecnologías (por ejemplo, cámaras). Por dar un ejemplo, Hsieh & Liou [69] logran

un reconocimiento gestual con un 95,37 % de exactitud utilizando cámara, mientras que Leon & Leija [100] alcanzaron un 99,81 % utilizando la señal EMG. Chen & Wang [27], incluso utilizan electrodos sEMG inalámbricos para capturar la señal EMG, logrando una buena exactitud de reconocimiento en tiempo real (por encima del 90 %). En todos estos casos se analizaron gestos o posturas estáticas. A continuación presentamos una revisión científica de trabajos en los cuales se han utilizado diferentes características, algoritmos y grupos de sensores EMG con el objetivo de determinar el algoritmo que mejores resultados haya obtenido, la combinación de características más eficientes, el número de sensores usados, y la principal contribución lograda. Esto, con el fin de validar mejor aún nuestra implementación. La Tabla 3.2 resume dicha revisión.

Xun Chen et al. [27] en su trabajo orientado al reconocimiento de gestos usando sensores sEMG y un sistema de comunicación inalámbrico, analiza los efectos de tres características (*Hudgins' time-domain features(TD)*, *Autocorrelation and cross-correlation coefficients (ACCC)*, *Spectral power magnitudes (SPM)*) usando cuatro algoritmos de clasificación y su efecto sobre la exactitud de reconocimiento fuera de línea (*off line*), concluyendo que es posible alcanzar hasta un 97.93 % de exactitud al usar un método de aprendizaje de kernel múltiple (Multiple Kernel Learning, MKL) basado en Maquinas de Vectores de Soporte (Support Vector Machines, SVM), superando a clasificadores como *k*-NN (*k*-Nearest Neighbor), LDA (Linear Discriminant Analysis) y QDA (Quadratic Discriminant Analysis), comúnmente utilizados para este tipo de señales. Para el reconocimiento en línea (*on line*), Xun Chen et al. emplearon características basadas en el dominio del tiempo (TD) con un clasificador QDA. En tiempo real, se lograron resultados de reconocimiento entre 90 % y 95 %.

La variación de la exactitud entre los gestos analizados bajo el mismo método es igualmente notable en varios trabajos. Por ejemplo, según la Tabla 3.2, León et al. [100] reportaron variaciones entre 97,23 % y 99,81 %, para 9 gestos o posturas analizadas, mientras que Al-Timemy et al. [3] reportaron variaciones entre 79 % y 98,25 % en 12 gestos. Sin embargo, también es necesario indicar varios aspectos que hacen que nuestra investigación sea más importante. Así tenemos:

- Hsieh y Liou [69] implementan un clasificador basado en SVM con núcleo RBF para lograr un sistema de reconocimiento de gestos basado en visión con un conjunto de datos logrado con 5 usuarios. Este trabajo no prueba con otros tipos de kernel para medir su impacto en la exactitud. Además, el conjunto de datos obtenido es escaso. El clasificador se ejecuta en un equipo

Tabla 3.2: Revisión bibliográfica de aplicación de características y algoritmos de ML

Primer autor	Ds	MC	LG	ACC%	Principal contribución
Hsieh [69]	Cam	SVM	6	95.37%	Reconocimiento robusto del gesto tanto dinámico como estático.
Artemiadis [6]	EMG	SVM+varios	SIG	>82%	Presentó una metodología de control para un robot antropomórfico usando señales EMG.
Alavi [4]	IMU	SVM+ANN	6	>99%	Reconocimiento del gesto basado en SVM y ANN
Ibañez [72]	Cam	DTW1+HMM	7	>98%	Presenta una herramienta de reconocimiento gestual como ayuda para desarrolladores para incluir el gesto en la interfaz
Benatti [13]	EMG	SVM	4	92.36%	Reconocimiento de patrones para control de prótesis usando un sistema empostrado
Chen [27]	EMG	k-NN+LDA+QDA+SVM	10	>97%	Presenta resultados con cuatro algoritmos diferentes para reconocimiento gestual
León [100]	EMG	SVM,ANN,LDA	9	>99.81%	Muestra resultados de aplicar clasificadores en el dominio de la frecuencia y mejorar su exactitud
KIM [89]	EMG	k-NN+Bayes	4	94%	Muestra su sistema de reconocimiento seguro del gesto usando solamente un electrodo
Kulkarni [93]	Cam	SVM, k-NN, NN	5	91%	Expone un sistema de reconocimiento gestual basado en SVM y en tiempo real
Zhai [202]	EMG	CNN+SVM	SIG	87.65%	Muestra un sistema basado en C-NN con buen rendimiento y eficiente entrenamiento
Kakoty [83]	EMG	RBF+SVM	SIG	97.5%	Presenta resultados aplicando PCA como característica FD
Menon [115]	EMG	LDA	7	% de error	Presenta un análisis de longitud de la ventana de análisis de acuerdo al número de electrodos

Ds, Dispositivo usado; MC, Método de clasificación; LG, Lenguaje gestual; ACC%, Exactitud y porcentaje.

Pc.

- Artemiadis y Kyriakopoulos [6] utilizan SVM sin indicar el núcleo empleado. En su investigación, cuatro sujetos participaron, lo que limita el conjunto de datos. Sin embargo, fueron capaces de demostrar que el método de seguimiento propuesto en su estudio no se ve afectado por los cambios en la señal EMG con respecto al tiempo.
- Alavi et al. [4] aplica SVM utilizando un núcleo lineal para clasificar seis gestos diferentes, que implica el movimiento de los brazos, antebrazos y manos. Once sujetos (4 mujeres y 7 hombres) participaron en el estudio. Sin embargo, menciona que hubo problemas de reconocimiento atribuibles a la muestra utilizada, lo que implica que un mayor número de participantes en la formación del algoritmo es aconsejable.
- Ibañez et al. [72] presenta una herramienta (EasyGR) basada en algoritmos de aprendizaje, concretamente combinando Alineamiento Temporal Dinámico (Dynamic Time Warping, DTW) y modelos ocultos de Markov, para ayudar a reducir el esfuerzo en el reconocimiento de gestos de los usuarios. Para su estudio y propuesta usaron a Kinect, con modelos de articulación y esqueléticos, que involucraba a siete gestos, obteniendo una tasa de reconocimiento mayor al 99%. Los movimientos, involucraban a brazos y manos, pero no a dedos y su modelo se basaba en el aspecto corporal completo. En este estudio participaron 10 personas.
- Benatti et al. [13] utilizan SVM para el reconocimiento de siete gestos de la mano en cuatro usuarios. El entrenamiento de SVM con kernel RBF fue hecho solo con el 10% del conjunto de datos lo cual que limita las posibilidades de generalización del algoritmo. Benatti también menciona que su sistema utilizó el protocolo Bluetooth 2.1, que tiene una tasa de transferencia mucho más baja que el estándar 4.0. Benatti menciona en sus conclusiones que lograron una precisión del 92% pero con sólo 4 gestos, y utilizando los siete gestos propuestos, lograron una precisión del 89,2%. Esto indica que el sistema propuesto por Benatti no soporta una carga gestual amplia como la que nosotros proponemos. No se discute ninguna técnica de extracción de características.
- Chen y Wang [27] proponen un sistema de reconocimiento de 10 gestos con la participación de seis sujetos. Utiliza cuatro canales y su clasificador se

basa en SVM con kernels RBF y Polinomial y alcanzó una precisión media de 97,77%. También implementaron k-NN, que alcanzó una precisión media del 64,7%. Chen y Wang utilizaron la extracción de características en el dominio del tiempo. Sin embargo, a pesar de sus buenos resultados, utilizan otro clasificador (QDA) en su implementación. El sistema de reconocimiento de gesto implementado requería la alta potencia de procesamiento de un ordenador. Los electrodos se distribuyeron en el antebrazo para capturar la señal EMG de los músculos involucrados.

- León et al. [100] utilizó SVM con núcleo RBF para clasificar nueve gestos utilizando un conjunto de datos logrado con 12 sujetos. Se utilizaron características en el dominio del tiempo y la frecuencia. En las pruebas, SVM logró una precisión media del 97,23%. A pesar de los buenos resultados con SVM, León no investigó el comportamiento de otros kernels. Se utilizaron cuatro canales y los electrodos sEMG se colocaron dispersos sobre el antebrazo y directamente sobre los músculos. Finalmente, León concluye que SVM es la mejor opción para el análisis de las señales EMG.
- Kim et al. [89] utilizaron k-NN y Bayes combinados en un nivel fusionado de decisión para clasificar 4 gestos de las manos. Se utilizó un solo canal con un electrodo colocado directamente sobre el músculo. El sistema de reconocimiento implementado dependía de la alta potencia de procesamiento de un ordenador y estaba sujeto a constantes cambios de calibración, probablemente debido al limitado conjunto de datos logrado con un solo canal.
- Kulkarni & Urabinahatti et al., [93] comparan SVM con otros algoritmos de clasificación muy usados, a saber: k-Vecino más cercano (*k*-Nearest Neighbors, *k*-NN) y Redes Neuronales Artificiales (Artificial Neural Network, ANN). Kulkarni et al., logran implementar un sistema de reconocimiento del gesto en tiempo real, en el cual el principal problema era la complejidad del reconocimiento, conjuntamente con la complejidad de los algoritmos de clasificación. De las pruebas realizadas concluye que, de los algoritmos estudiados, SVM alcanza un 91% de exactitud en un sistema de reconocimiento en tiempo real.
- Zhai et al. [202] a pesar de utilizar una red neuronal convolucionada (convolucional neural network, CNN) logró establecer una comparación con SVM utilizando el kernel RBF. Se utilizó la base de datos del proyecto NinaPro, la cual se obtuvo de 40 sujetos intactos, utilizando 12 canales y realizando 49 tipos de movimientos de manos. En este trabajo no se indica la ubicación

de los sensores sEMG. SVM alcanzó una precisión del 77,44 % y CNN lo superó en apenas un 1,13 %. Se ensayaron varias características basadas en el dominio de la frecuencia. Sin embargo, la exactitud alcanzada se mantuvo por debajo del 80 % para ambos clasificadores. No hay indicación de la existencia de una implementación en tiempo real y debe tenerse en cuenta que la potencia de cálculo utilizada era alta (CNN se ejecutó en NVIDIA CUDA® y SVM utilizando 4 núcleos Intel i5).

- Kakoty et al. [83] utilizaron dos canales para capturar señales EMG en 80 sujetos sanos (60 varones y 20 mujeres) realizando seis tipos de gestos de sujeción. El núcleo utilizado para SVM fue RBF. Los electrodos sEMG se localizaron en el antebrazo y la muñeca, es decir, directamente en los músculos involucrados en el movimiento. Se utilizaron características de dominio del tiempo (TD) y frecuencia (FD), obteniendo una tasa de reconocimiento promedio de 76 % y 81 %, respectivamente. La potencia informática utilizada no se menciona, pero se deduce que fue un ordenador. Por otro lado, aunque el número de participantes es alto, el uso de pocos canales podría aumentar el error de clasificación [115]. Kakoty señala que la SVM demuestra su validez en este tipo de señales.
- Finalmente, Menon et al. [115] usaron un clasificador LDA para clasificar siete gestos de manos en 13 sujetos (11 hombres y 2 mujeres) con problemas de amputación en sus brazos. Analizó las características de tiempo y frecuencia de la señal EMG y fue capaz de determinar que el uso de una técnica de ventanas superpuestas no tiene ningún efecto directo en la exactitud de clasificación. El procedimiento, similar al de Kakoty, consistió en extraer características del dominio del tiempo de la señal EMG resultante.

A partir del análisis realizado podemos concluir, que el algoritmo más utilizado para el tratamiento de las señales EMG es SVM. Sin embargo, como lo hacen en varios otros trabajos, usaremos k -NN como un algoritmo de contraste para evaluar nuestros resultados. En este sentido, SVM no sólo tiene una estructura más sencilla, en comparación con Análisis Discriminante Lineal (LDA) y los modelos de Redes Neuronales Artificiales (ANN), sino que también es capaz de interpretar la información de manera más eficiente, convirtiéndolo en una buena opción para el análisis de señales EMG [100]. Con respecto a k -NN, sólo para citar un ejemplo, además de la revisión científica anterior, encontramos que Li et al. [103] establecen una comparación entre el algoritmo k -NN y el algoritmo de Neural Ga-

sificado Multicanal Semi-Supervisado (multi-canal semi-Supervised Growing Gas Neural, OSSMGNG), encontrando una diferencia mínima en favor de OSSMGNG (diferencia de 0,2%). El algoritmo OSSMGNG es semi-supervisado, más complejo de implementar y sus resultados son similares a k -NN. Kotsiantis [91] destaca algunas particularidades de los algoritmos k -NN y SVM en comparación con otros clasificadores.

Continuando con el análisis realizado por Kotsiantis, la Tabla 3.3 muestra algunos de sus resultados y a los cuales hemos creído prudente mencionar, aunque en resumen, por estar estrechamente relacionados con la búsqueda del algoritmo ideal para nuestra propuesta.

Tabla 3.3: Características de los algoritmos SVM y k -NN

Gesto	k-NN	SVM
Exactitud general demostrada	BT	MAT
Velocidad de clasificación	MBT	MAT
Tolerancia para valores perdidos	MBT	BT
Tolerancia a atributos redundantes	BT	TM
Tolerancia al ruido	MBT	BT
Permite intentos de aprendizaje incremental	MAT	BT
Permite manejar parámetros del modelo	TM	MBT

MAT, Muy alta tolerancia; TM, Tolerancia media; BT, Baja tolerancia; MBT, Muy baja tolerancia.

De los trabajos de investigación expuestos, es importante resaltar que muchos de ellos no han extendido sus resultados, evitando experimentar con otros tipos de núcleo para SVM. A pesar de esto, en la mayoría de los estudios expuestos se utiliza el núcleo RBF (Radial Basis Functions, en inglés).

A diferencia de otros trabajos nosotros proponemos el análisis de las señales EMG usando dos clasificadores, que son SVM y k -NN. En este análisis, se usarán de ocho canales con ocho sensores EMG (1 canal por sensor), situados radialmente en el antebrazo del usuario. Además, pretendemos trabajar con un dataset (conjunto de datos) mucho más amplio debido a nuestra extensa biblioteca gestual, permitiendo ampliar las posibilidades de aplicación a sistemas más complejos. Por último, proponemos la validación de nuestra investigación mediante la implementación de un sistema que implemente una interfaz de reconocimiento gestual orientada al control de dispositivos de casa. Pretendemos independizar a la pulsera gestual del algoritmo del fabricante y hacerla aún más portable, ejecutando el sistema de reconocimiento, sobre un sistema embebido. Todo esto, logrará establecer nuevas alternativas de interacción natural del usuario de tipo gestual (INU-G) aplicadas

al control de dispositivos.

3.5.2. Algoritmo Support Vector Machines

Las Máquinas de Soporte de Vectores (Support Vector Machines, SVM) han tenido un gran impacto en la comunidad dedicada al aprendizaje automático, creando un entusiasmo similar al experimentado anteriormente por las Redes Neuronales Artificiales (Artificial Neural Networks, ANN). En un principio, los SVM fueron presentados como la técnica más potente para clasificaciones generales (No Lineales), regresión y detección aislada con un modelo intuitivo de representación. Actualmente, las SVM, constituyen un conjunto de métodos de aprendizaje supervisado, usados para clasificación y regresión. Básicamente, SVM permite, dado un conjunto de ejemplos de entrenamiento entre dos categorías posibles, construir un modelo que predice si un nuevo ejemplo pertenece a una u otra categoría[93].

SVM en casos linealmente separables

Las máquinas de soporte de vectores están basadas en el aprendizaje estadístico. La máquina de aprendizaje, implementada por SVM permite encontrar un modelo lineal (la función de decisión o frontera de separación es una recta) que separa los elementos de ambas clases, con un margen máximo, permitiendo la mejor generalización posible. La habilidad de generalización, hace referencia al hecho de que, el clasificador no solo es bueno en realizar la mejor clasificación en los datos de entrenamiento (exactitud), sino que también garantiza una alta exactitud predictiva para datos futuros, a partir de los mismos datos de entrenamiento [194].

En este sentido, un margen puede ser definido intuitivamente como una cantidad de espacio o separación entre dos clases definidas por un hiperplano. Geométricamente, el margen corresponde a la distancia más corta entre los puntos de datos mas cercanos a algún punto en el hiperplano. La Fig. 3.8 ilustra la construcción geométrica del hiperplano óptimo bajo las condiciones dadas sobre un espacio de entrada de dos dimensiones.

Dado que w y b denotan el peso del vector y la tendencia en el hiperplano óptimo respectivamente. El hiperplano esta definido entonces como:

$$w^T x + b = 0 \quad (3.2)$$

La distancia geométrica direccional deseada desde la muestra x hacia el óptimo hiperplano es:

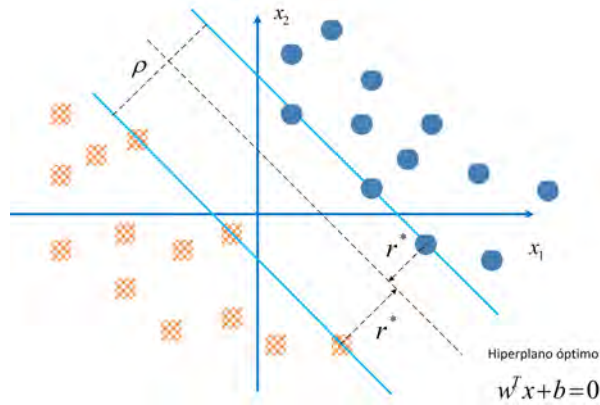


Figura 3.8: Hiperplano óptimo en SVM para un caso separable linealmente.

$$r = \frac{g(x)}{\|w\|} \quad (3.3)$$

donde $g(x) = w^T x + b$ es la función discriminante definida como el hiperplano y también llamada margen funcional dado w y b .

Consecuentemente, el clasificador basado en SVM tiene como fin encontrar los parámetros w y b para un hiperplano óptimo que permita maximizar el margen de separación (ρ) que determina la distancia geométrica más corta r^* desde las dos clases respectivamente, por ello SVM es también llamado *clasificador de margen máximo*. Ahora sin perder la generalidad, fijamos el margen funcional a 1, que esta dado por el conjunto de entrenamiento $\{x_i, y_i\}_{i=1}^n \in R^m \times \{\pm 1\}$, donde:

$$\begin{aligned} w^T x + b &\geq 1 \text{ for } y_i = +1 \\ w^T x + b &\leq -1 \text{ for } y_i = -1 \end{aligned} \quad (3.4)$$

Los puntos de datos particulares (x_i, y_i) para los cuales la primera y la segunda parte de la ecuación(3.5) son satisfechos son llamados *vectores de soporte*, los cuales a su vez son exactamente el espacio óptimo del hiperplano.

El hiperplano óptimo es el que satisface el margen máximo. Entonces puede asumirse que todos los datos de entrenamiento que se encuentran por encima de la línea +1 satisfacen:

$$\frac{w \cdot x_{(+1)} + b}{\|w\|} \geq r^*$$

Entonces, las muestras de entrenamiento por debajo de la línea -1 deben satisfacer

$$\frac{-(w \cdot x_{(+1)} + b)}{\|w\|} \geq r^*$$

Compactando las dos expresiones anteriores en una sola queda la desigualdad:

$$\frac{y_k(w \cdot x_{(+1)} + b)}{\|w\|} \geq r^*, \quad k=1,2,\dots, m$$

donde $y_k \in \{-1, 1\}$ y m es el número de datos de entrenamiento.

El problema de encontrar el hiperplano óptimo se reduce a encontrar el w que maximiza el margen ρ (r^*). Es de especial importancia, el hecho que hay un número infinito de soluciones que se diferencian únicamente en un factor de escala para w . Para limitar las soluciones a una única, se fija la condición $\rho \|w\| = 1$. Por consiguiente maximizar el margen equivale a minimizar la normal de w

Suponiendo un conjunto de entrenamiento linealmente separables con m muestras

$$(x_1, Y_1), \dots, (x_m, y_m), x \in \mathbb{R}^n, y_k \in \{+1, -1\} \quad k = 1, \dots, m \quad (3.5)$$

La frontera de separación de las dos clases va a ser una función lineal de la forma:

$$D(x) = w \cdot x + b \quad (3.6)$$

La cual permite la separación de los datos de entrenamiento sin errores. Todas las muestras cumplen,

$$y_k (w \cdot x_k + b) \geq 1, \quad k = 1 \dots m \quad (3.7)$$

donde la condición de igualdad se satisface para los vectores de soporte. Esta ecuación describe a un hiperplano de separación en función de las muestras de entrenamiento, “incluso sin necesidad de tener que estimar previamente la función de distribución de probabilidad que obedece la fuente de datos”.

Posteriormente, SVM ha sido adaptado para ser aplicado a problemas de clasificación con datos no separables. Finalmente, se extendió la funcionalidad de SVM para poder resolver problemas de regresión.

SVM en casos no separables linealmente

En lo explicado hasta ahora, se ha supuesto que dos clases que forman las muestras de entrenamiento están completamente separadas en el espacio de entrada, espacio de las x , mediante la función de decisión lineal. Sin embargo puede ocurrir dos situaciones. La primera de ellas corresponde al caso en el cual la función de decisión es lineal pero las muestras de entrenamiento no pueden separarse sin errores, Figura 3.9.b. La segunda situación queda reflejada en la Fig. 3.9.a y se pone de manifiesto que las muestras de entrenamiento no pueden separarse en el espacio de entrada mediante una función lineal. Ambos casos serán mencionados, pero describiremos con mayor detalle al segundo por ser el objeto de nuestro estudio.

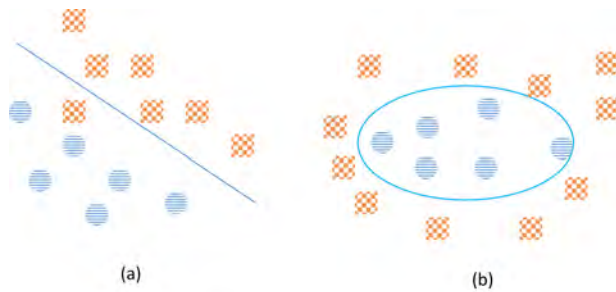


Figura 3.9: Dos casos de separación en SVM. (a) Función de decisión lineal con errores. (b) Función de decisión no lineal.

Función de decisión lineal con errores

En este caso, la frontera de separación de las dos clases es lineal, pero hay al menos una muestra de una de las clases que aparece al otro lado del hiperplano de separación. e busca tanto que el margen sea máximo (hiperplano óptimo) como que el hiperplano que se obtenga deba separar los datos con el menor número de errores.

$$y_k (w \cdot x_k + b) \geq 1, k = 1 \dots m \quad (3.8)$$

Una muestra de entrenamiento es *no separable* cuando no satisface la ecuación (3.8). Este aspecto debe diferenciarse del término "*malclasificada*". Esta última corresponde a la situación en la que la muestra aparece en el lado equivocado del hiperplano (Fig. 3.10).

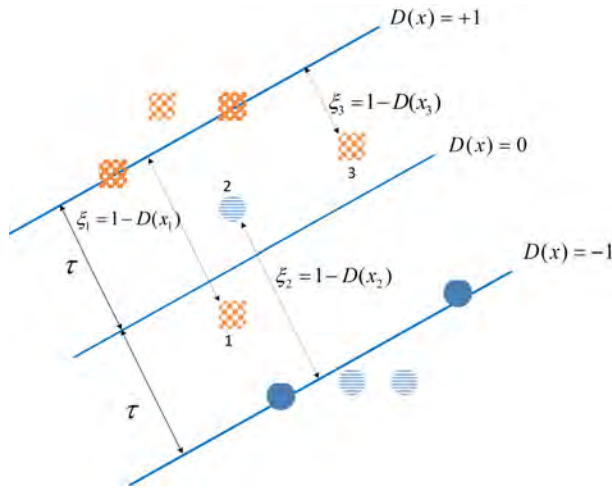


Figura 3.10: Clasificación con SVM, casos *separables*, *no separables* y *mal clasificados*.

En la Fig. 3.10 se puede observar como los puntos x_1, x_2 y x_3 son no separables. Los dos primeros están mal clasificados pero la clasificación del tercero es correcta.

Para lograr una clasificación con un hiperplano que cometa el mínimo número de errores se introduce un conjunto de variables no negativas, una por cada muestra de entrenamiento $\xi_k \geq 0, k = 1 \dots m$. El cálculo del hiperplano de separación a partir de las muestras de entrenamiento (aquí ecuación de muestras lineales), requiere determinar los w y b que minimizan la función siguiente:

$$\Phi_w = \frac{1}{2} \|w\|^2 + C \sum_{k=1}^m \xi_k \quad (3.9)$$

sujeto a las restricciones

$$y_k(w \cdot x_k + b) \geq 1 - \xi_k, k = 1 \dots m \quad (3.10)$$

y donde C es una constante de regularización dada (supuestamente suficiente grande).

La introducción de variables no negativas, Fig. 3.10, ha permitido relajar las condiciones 3.8. Para aquellas muestras de entrenamiento que quedan bien clasificadas, su correspondiente variable $\xi_k = 0$. Si $\xi_k > 0$, la muestra es no separable y si $\xi_k > 1$, entonces la muestra está mal clasificada.

Para los espacios de alta dimensionalidad, el problema es análogo al anterior, es decir para casos linealmente separables. Entonces se trata de encontrar los parámetros $\alpha_k, k = 1 \dots m$ que maximizan la función siguiente:

$$\Phi(\alpha) = \sum_{k=1}^m \alpha_k - \frac{1}{2} \sum_{i,j=1}^m \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) \quad (3.11)$$

sujeto a

$$\left. \begin{aligned} \sum_{k=1}^m \alpha_k y_k &= 0 \\ 0 \leq \alpha_k &\leq C, k = 1 \dots m \end{aligned} \right\} \quad (3.12)$$

La ecuación del hiperplano frontera entre las dos clases es exactamente la misma que la del caso separable y que se reproduce aquí para dar mayor claridad.

$$D(x) = \sum_{\text{Vectores-soporte}} \alpha_k y_k (x \cdot x_k) + b \quad (3.13)$$

Nuevamente, sólo las muestras de entrenamiento que han resultado ser vectores de soporte $\alpha \neq 0$. Además es importante observar que la ecuación 3.13 depende únicamente de los productos escalares entre los vectores de características. Finalmente hay que indicar que este problema de optimización difiere del formulado en el caso linealmente separable en la existencia de una cota superior para los α_k en las restricciones 3.12.

Clasificadores no lineales

Cuando las muestras de entrenamiento no pueden separarse mediante una función lineal, se recurre a transformaciones no lineales que llevan del espacio de entrada a otro espacio de alta dimensionalidad, incluso infinita, en el que los datos son separables linealmente. A este nuevo espacio se le denomina espacio de característica, para distinguirlo del espacio de entrada.

Mediante una adecuada transformación no lineal Ψ a una dimensionalidad suficientemente alta, siempre es posible separar dos clases mediante un hiperplano en el espacio transformado. Se dice entonces que se lleva los datos a un espacio de mayor dimensionalidad y justamente la presencia de dimensiones adicionales permite la separación lineal de las muestras en el nuevo espacio. La Fig. 3.11 trata de mostrar dicha transformación.

Sea $\Psi_j(x), j = 1 \dots M$ un conjunto de funciones no lineales definidas a priori que permiten pasar el espacio de entrada al espacio de características. Estas funciones

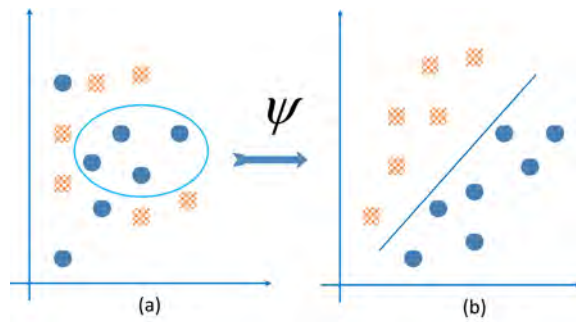


Figura 3.11: Transformación a un espacio de alta dimensionalidad.

transforman cada vector de entrada x en un vector M -dimensional perteneciente al espacio de características. Si consideramos que el espacio de entrada es un espacio bidimensional, es decir, los vectores de entrada tienen dos componentes: $x = (x_1, x_2)$. Y si suponemos que las funciones $\Psi_i(x)$ corresponden a monomios de las componentes de x hasta cierto grado (2 por ejemplo). En este caso el conjunto de funciones de transformación serían:

$$\left. \begin{aligned} \Psi_1(x_1, x_2) &= 1 & \Psi_2(x_1, x_2) &= x_1 \\ \Psi_3(x_1, x_2) &= x_2 & \Psi_4(x_1, x_2) &= x_1x_2 \\ \Psi_5(x_1, x_2) &= x_1^2 & \Psi_6(x_1, x_2) &= x_2^2 \end{aligned} \right\} \quad (3.14)$$

Entonces, dado el vector de entrada $(-2,3)$, su representación en el espacio transformado sería: $(1,-2,3,-6,4,9)$

Si la frontera de separación es lineal en el espacio de características, significa que dicha función puede expresarse como sigue:

$$D(x) = \sum_{j=1}^M w_j \Psi_j(x) = w \cdot \Psi \quad (3.15)$$

donde el número de sumandos depende de la dimensionalidad M del espacio transformado. Esta expresión es equivalente a la ecuación (3.6), donde el término b queda implicado ahora incluyendo una función constante, por ejemplo $\Psi(x) = 1$, en el espacio de características.

Para lograr ubicar la transformación adecuada en un problema, en el caso más general, se debe encontrar el hiperplano de separación equivalente a las ecuaciones (3.9) y (3.10). Es decir, se debe minimizar la ecuación

$$\Phi_w = \frac{1}{2} \|w\|^2 + C \sum_{k=1}^m \xi_k$$

con las restricciones

$$\begin{aligned} y_k [w \cdot \Psi(x_k) + b] &\geq 1 - \xi_k, k = 1 \dots m \\ \xi &\geq 0, k = 1 \dots m \end{aligned}$$

El vector w , considerando los multiplicadores de Lagrange representados por α_k , puede ser escrito como:

$$w = \sum_{k=1}^m \alpha_k y_k x_k \quad (3.16)$$

Luego, la ecuación (3.16), puede escribirse como combinación lineal de los vectores soporte (en este caso transformados)

$$w = \sum_{\text{vector-soporte}} \alpha_k y_k \Psi(x_k) \quad (3.17)$$

Y el hiperplano de separación resulta ser el proporcionado por la expresión siguiente:

$$D(x) = \sum_{\text{vector-soporte}} \alpha_k y_k (\Psi(x) \cdot \Psi(x_k)) + b \quad (3.18)$$

Elegir las funciones $\Psi_j(x)$ es equivalente a fijar una función $K(x, x_j)$, llamada núcleo, tal que:

$$\Psi(x) \cdot \Psi(x_k) = K(x, x_k) \quad (3.19)$$

De esta manera, el cálculo de los productos escalares en el espacio de gran dimensionalidad es efectuado indirectamente mediante la evaluación del núcleo K .

La gran ventaja es que el cálculo del núcleo K se lleva a cabo en el espacio de entrada con las muestras de entrenamiento. La presencia de funciones núcleo permite la transformación de los datos a espacios de altísima dimensionalidad (quizá infinita), pero sin necesidad de manipular las muestras en los espacios transformados.

Algunas de las funciones núcleo más comúnmente usadas son:

Nucleo Polinomial

Genera polinomios de grado d . La función que permite su aplicación es

$$K(x, x') = \left[(x \cdot x') + 1 \right]^d \quad (3.20)$$

Nucleo Radial

Genera núcleos radiales, utilizando funciones de la forma $K(x, x') = K(x - x')$ y cuya función es:

$$K(|x - x'|) = \exp \left\{ -\frac{|x - x'|^2}{2\sigma^2} \right\} \quad (3.21)$$

Nucleo Sigmoidal de redes neuronales de dos capas

Para este núcleo la función es:

$$K(x, x') = \tanh \left[v(x \cdot x' + a) \right] \quad (3.22)$$

Expresando el problema de optimización en su forma dual, los multiplicadores de Lagrange $\alpha_k, k = 1 \dots m$ se pueden obtener conociendo únicamente el núcleo. Para ello, dados los datos de entrenamiento

$$(x_1, y_1), \dots, (x_m, y_m), x \in \mathbb{R}^n, y_k \in \{+1, -1\} k = 1 \dots m$$

Una función núcleo K y un parámetro de regularización C , hay que maximizar la función 3.11, sujeto a 3.12.

Hay que destacar que hasta el momento no existe una teoría para determinar un valor óptimo de C . Típicamente, suele fijarse a un valor muy grande o en potencias sucesivas. Para el caso separable, $C = \infty$.

La función de decisión queda entonces expresada como:

$$D(x) = \sum_{\text{vectores-soporte}} \alpha_k y_k K(x, x_k) + b \quad (3.23)$$

Hay que tener presente que en un problema biclase, la clasificación de una muestra x_c depende únicamente del signo de la función, en la ecuación 3.5.

Problemas multiclase

SVM resuelve problemas biclase, por lo cual un método para resolución de problemas multiclase usando SVM se basa en la combinación de clasificadores binarios. Consiste en descomponer un problema multiclase en pequeños problemas binarios, aplicando después funciones de decisión para unirlos. Estas técnicas son:

Una frente a todas (one-versus-all) *Descompone un problema multiclase con N_c clases en otros tantos problemas binarios, en los cuales cada una de las clases se enfrentan al resto. Así se construyen N_c clasificadores que definen otros tantos hiperplanos que separan la clase i de las $N_c - 1$ restantes. Para que se considere una clasificación admisible es necesario que el elemento a clasificar sólo sea asociado a una clase en un biclasificador, mientras que en todos los demás biclasificadores el elemento debe pertenecer a la categoría del resto.*

Una frente a una (one-versus-one) *Descompone el problema de N_c clases en $N_c \frac{(N_c - 1)}{2}$ problemas binarios, donde se crean todos los posibles enfrentamientos uno a uno entre clases. Así, se obtiene un hiperplano para cada uno de estos problemas binarios. Posteriormente, se somete a cada nuevo elemento a clasificar a todos estos clasificadores, y se añade un voto a la clase ganadora para cada caso, resultando como clase propuesta la que más votos suma.*

Algoritmo SVM

El algoritmo 1 trata de ubicar los límites de una solución ϵ -aproximada y en el cual la dispersión de los datos es independiente del tamaño del conjunto de entrenamiento n . En este algoritmo se busca el plano de separación óptimo usando un número constante de iteraciones y fue propuesto por Joachims [79]. Joachims supone un gran número de muestras n y un gran número de características (features)

N y es una adaptación del algoritmo de entrenamiento SVM estructural.

Algoritmo 1: Algoritmo básico para entrenamiento de SVMs

```

1 Entrada :  $S = ((x_1, y_1), \dots, (x_n, y_n), C, \epsilon)$ , el conjunto de características con
   sus clases de entrenamiento, además de  $C$  y  $\epsilon$ .;
2 Salida :  $w, \xi$  ;
3  $W \leftarrow 0$ ;
4 repeat
5    $(w, \xi) \leftarrow \operatorname{argmin}_{W, \xi \geq 0} \frac{1}{2} W^T W + C\xi$ ;
6   s.t.  $\forall c \in W : \frac{1}{n} W^T \sum_{i=1}^n c_i y_i x_i \geq \frac{1}{n} \sum_{i=1}^n c_i - \xi$ ;
7   for  $i = 1, \dots, n$  do
8      $c_i \leftarrow \begin{cases} 1 & y_i(w^T x_i) < 1 \\ 0 & \text{otherwise} \end{cases}$ 
9   end
10   $W \leftarrow W \cup \{c\}$ ;
11 until  $\frac{1}{n} \sum_{i=1}^n c_i - \frac{1}{n} \sum_{i=1}^n c_i y_i (w^T x_i) \leq \xi + \epsilon$ ;

```

3.5.3. Algoritmo k-Nearest Neighbor

k -NN encuentra un grupo de k objetos en el conjunto de entrenamiento los cuales están más cerca del objeto de prueba y basa la asignación de una etiqueta en el predominio de una clase particular en vecindario (neighborhood) [195]. La clave de este enfoque es:

- El conjunto de objetos etiquetados que se utilizarán para evaluar la clase de un objeto de prueba (esto no necesariamente será el conjunto de entrenamiento completo),
- Una métrica de distancia o similitud que se puede usar para calcular la cercanía De los objetos,
- El valor de k , el número de vecinos más cercanos, y
- El método utilizado para determinar la clase del objeto objetivo sobre la base de las clases y distancias de los k vecinos más cercanos [195].

El enfoque k -NN es también un ejemplo de una técnica de aprendizaje perezoso, la cual es una técnica que espera hasta que la consulta llegue a generalizar, más allá de los datos de entrenamiento [91]. k -NN se basa en el principio de que

las instancias, dentro de un conjunto de datos, por lo general existirán en estrecha proximidad a otras instancias que tienen propiedades similares. Por lo tanto, la mitad de la información disponible en una colección infinita de muestras clasificadas se encuentra en el vecino más cercano [34, 190], como lo muestra la Figura 3.12. Si las instancias son etiquetas para su clasificación, el valor de la etiqueta de una instancia no clasificada se puede determinar observando la clase de sus vecinos más cercanos. k -NN localiza las instancias k más cercanas a la instancia de consulta y determina su clase, identificando la etiqueta de clase más frecuente [91]. Para determinar la distancia relativa entre las instancias más significativas, k -NN utiliza una métrica de distancia. Las métricas más significativas son Mikowsky, Manhattan, Chebychev, Euclidean, Canberra y la correlación de Rank de Kendall.

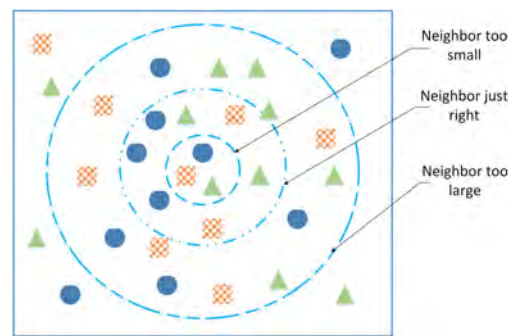


Figura 3.12: Diferentes tipos de valores de k .

Nosotros hemos aplicado la Distancia Euclídea por se fácil de implementar y además garantiza una buena velocidad de cálculo. Su ecuación es:

$$d(x, y) = \left(\sum_{i=1}^m |x_i - y_i|^2 \right)^{1/2} \quad (3.24)$$

Para mayores detalles sobre los algoritmos de aprendizaje perezoso puede revisar el trabajo realizado por Wettschereck et al. [190].

k es un factor clave en el algoritmo k -NN, ya que no debe ser tan pequeño que sea afectado por el ruido, ni tan grande que el vecino más cercano pueda influir en él [91, 34]. Varias cuestiones clave afectan el rendimiento de k -NN de acuerdo a la elección de k (Figura 3.12). Se puede obtener una estimación del mejor valor para k mediante la validación cruzada. En nuestro caso, evitamos el uso de $k = 1$ (demasiado pequeño) porque son muestras lo suficientemente grandes y el uso de k (demasiado grande), pues lo haría insensible al ruido. Es necesario normalizar los atributos para que otros con mucho más rango no tengan más peso que los demás.

El algoritmo 2 no entrega un resumen a alto nivel del métodos de calificación usando k -NN.

Algoritmo 2: Algoritmo básico para k -NN

```

1 Entrada :  $D$ , el conjunto de objetos de entrenamiento,  $\mathbf{z}$  el conjunto de
  objetos de test, el cual es un vector de valores de , y  $L$  es el conjunto de
  clases usadas para etiquetar los objetos.;
2 Salida :  $c_z \in L$ , la clase de  $z$  ;
3 for objeto  $y \in D$  do
4   | calcule  $d(z, y)$ , la distancia entre  $\mathbf{z}$  e  $\mathbf{y}$ ;
5 end
6 Selecciona  $N \subseteq D$ , el conjunto (vecino más cercano) encerrado en  $k$  de los
  objetos de entrenamiento  $z$ ;
7  $C_z = \operatorname{argmax}_{v \in L} \sum_{y \in N} I(v = \text{clase}(C_y))$ ;
8 Dónde  $I(\cdot)$  es una función indicadora que retorna el valor 1 si su argumento
  es TRUE y 0 en otros casos;

```

3.6. Proceso experimental

Para lograr un adecuado reconocimiento del gesto, a partir de la señal EMG recogida por los sensores sEMG, proponemos un conjunto de etapas que deberán seguirse adecuadamente. El flujo de datos que se describe en la Figura 3.13 trata de delinear el proceso experimental seguido. Dos tipos de experimentos fueron conducidos a saber: *Off-line*, o captura y procesamiento información en base a los datos capturados; y, *On-line*, o estudio experimental con un prototipo desarrollado e implementado para el efecto. En ambos casos debemos partir de un protocolo experimental definido tanto para la captura de datos como para su procesamiento. A continuación realizaremos una breve descripción de cada uno de sus componentes.

- **Captura de señales EMG:** Consiste en capturar la señal EMG a través de los sensores sEMG directamente brazo del usuario. Dichas señales podrán ser guardadas para conformar el dataset que a su vez conformará la Biblioteca Gestual (BL) que conforman el conjunto de gestos usados y reconocidos por el sistema.
- **Amplificación y filtrado:** Es parte de la obtención de la señal EMG y corresponde al dispositivo usado para la captura. Por lo general dicha fase la realizan los Amplificadores Operacionales (OpAmp siglas en inglés) y esta

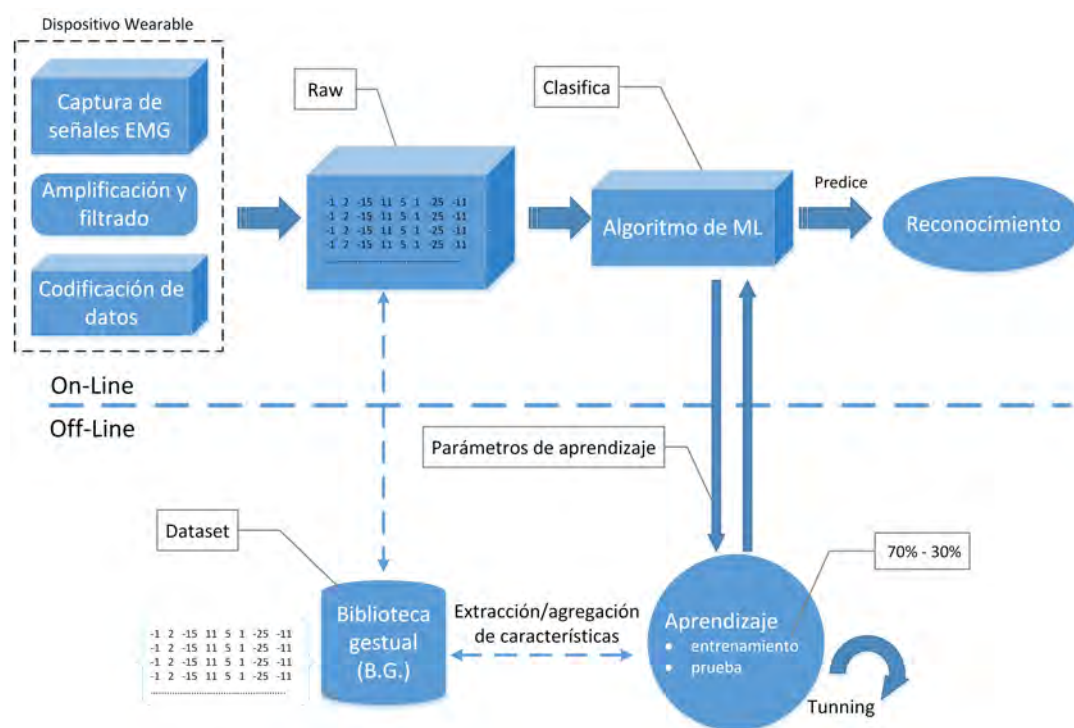


Figura 3.13: Flujo de la información para reconocimiento del gesto.

acompañada de una fase de filtrado (un rango determinado) para eliminación de ruido.

- Codificación de datos:** Esta fase es por lo general implícita en el dispositivo o sistema de captura y permite convertir la señal analógica (voltaje) en datos o componentes digitales de acuerdo a un rango de conversión predeterminado (64, 128 a 256 bits). Esta es dependiente directa del grado de precisión que se desee alcanzar con la señal.
- Raw de datos:** Son los datos que recibimos para procesamiento luego de haber pasado por las fases anteriores.
- Algoritmo de ML:** Este está conformado por el algoritmo de Aprendizaje Automático que mejor se ajuste a la señal y mejor exactitud de clasificación reporte. Este tema fue tratado inicialmente en el Capítulo 11, pero se lo desarrollará con mayor profundidad posteriormente en este mismo capítulo. Pero en breves rasgos es aquel que permite la clasificación del gesto para lograr su predicción o identificación, en base a determinados parámetros de entrenamiento.

- **Reconocimiento:** Esta fase establece la identificación del gesto propiamente dicha. Esta es dependiente de la biblioteca gestual implementada.
- **Biblioteca gestual:** La conforman el conjunto de datos correspondientes a cada gesto usado en este estudio. Cada gesto puede ser identificado por una etiqueta (clase) en concordancia con su característica. El dataset contiene los patrones gestuales que en muchas veces son visibles únicamente aplicando un algoritmo de minería de datos.
- **Extracción/agregado de características:** Son características propias de cada gesto tales como la amplitud alcanzada y los valores de señal EMG alcanzado por uno o varios canales a los cuales se han unido sensores EMG para la captura de la señal. La lectura que registre cada sensor podrá tomarse como una característica del gesto. Si esto no fuese suficiente, Markovitch & Rosenstein [113] recomienda introducir nuevas características implícitas en los datos para lograr una mejor identificación.
- **Aprendizaje:** El aprendizaje está relacionado con la exactitud lograda por el Algoritmo de ML usado. Básicamente comprende el entrenamiento y prueba. La proporción de datos por lo general se establece a tres cuartas partes para el entrenamiento y una cuarta parte para la prueba. Involucra al Tuning o afinamiento de clasificación, el cual se logra estableciendo los mejores parámetros de clasificación logrando la exactitud más alta. Aunque depende de varios factores (los datos, ruido, etc) es la mejor forma de obtener los mejores resultados. Depende en muchos casos del algoritmo de ML usado. Mayores detalles se abordarán más adelante en este capítulo.
- **Parámetros de aprendizaje:** Dependiendo de el grado de exactitud alcanzado y luego del aprendizaje, los parámetros que mejor se ajusten en precisión a los datos deberán ser considerados para lograr la mejor clasificación. En una implementación On-Line (en línea), por lo general en tiempo real, serán los que logren, junto al algoritmo de ML, el reconocimiento gestual.

Varios de los conceptos de aprendizaje automático ya fueron expuestos anteriormente en el Capítulo 1, pero serán ampliados posteriormente en forma pertinente. El Aprendizaje Automático es una ciencia muy amplia y compleja, que se escapa del alcance de esta investigación.

En este análisis, usaremos algoritmos de ML y los aplicaremos, tratando de alcanzar la identificación del gesto a partir de las señales EMG. Por lo dicho, la

Figura 3.13 nos permitirá trazar el camino a seguir para lograr la detección y reconocimiento gestual.

3.6.1. Librería gestual propuesta

Se ha planteado un total de 13 gestos, los cuales conforman una biblioteca gestual orientada a lograr comandos de control gestuales, que podrán ser aplicados en dispositivos eléctricos y electrónicos. Varios de los gestos, a saber gestos (a, b y d), han sido anteriormente validados por su alta precisión y exactitud (Capítulo anterior). Ahora, basándonos en esta nueva librería gestual, mucho más amplia y compleja, implementaremos un sistema de reconocimiento y control gestual basado en señales EMG del usuario. Cada uno de los gestos propuestos, será evaluado en exactitud de clasificación usando los algoritmos de ML propuestos. La biblioteca gestual completa se muestra en la Figura 3.14.

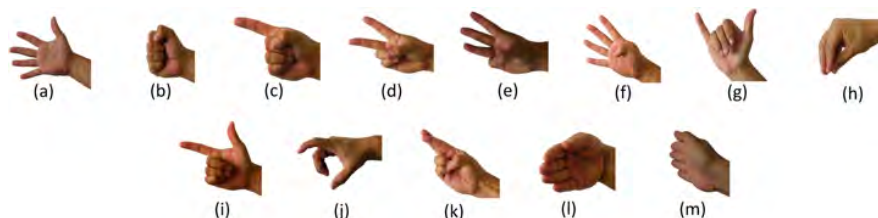


Figura 3.14: Gestos que conforman la librería de gestos para control.

3.6.2. Extracción de características

Para la extracción de características de la señal EMG, se aplicaron aquellas características en el Dominio del Tiempo (DT). Estas características fueron agregadas al Raw de la señal EMG recogida por los sensores [187]. Estas características incluyen la aplicación de la Rectificación de Onda Completa de la señal EMG (FW-REMG) y luego, una Integración EMG (IEMG). Características de este tipo, han sido usadas en otras investigaciones que tratan del estudio de señales EMG [83, 46, 107]. Además, estos valores proporcionan la información sobre los cambios en la señal EMG a lo largo del tiempo y serán agregados al dataset recopilado.

- **Rectificación:** Se prefiere a la aplicación de FW-REMG, el cual se conserva la energía de señal [189, 85]. Esta característica en ocasiones se considera un paso intermedio antes de la fase de integración de la señal. FW-REMG se

implementa convirtiendo todas las amplitudes negativas a amplitudes positivas. Los picos negativos se mueven hacia arriba en el lado positivo del eje de tiempo. En otras palabras, es lo mismo que tomar el valor absoluto de la señal EMG. Se define como:

$$x_3(t) = |x_2(t)| \quad (3.25)$$

donde $x_2(t)$ = Señal filtrada del sensor y $x_3(t)$ =Señal rectificada.

- **Integración:** La IEMG es una característica de la señal EMG que puede calcularse tomando la suma de valores absolutos de la señal EMG digitalizada. Se define por:

$$x_i(t) = \int_{t_1}^{t_2} x_3(t) dt \quad (3.26)$$

donde $x_i(t)$ = Señal integrada y $x_3(t)$ =Señal rectificada.

La extracción de características, permite extraer la información útil que esta implícita y oculta en la señal EMG. También, nos permite eliminar la parte no deseada (ruido), provocado por posibles interferencias. Otras características, ya sea en el dominio del tiempo o de la frecuencia, son válidas, pero nuestro propósito es minimizar la potencia de cálculo necesario para lograr determinar dichas características. Por esta razón las características escogidas fueron las que mejor se acoplaron a nuestro propósito. Esto debido a que nuestro sistema de reconocimiento será ejecutado en un sistema embebido de mediana escala, el cual debe asegurar la identificación del gesto y un tiempo de reconocimiento adecuado, en tiempo real. Entonces, nuestros vectores de características, que servirán de entrada al algoritmo de clasificación, incluyen: la señal Raw EMG, la señal rectificada con FW-REMG y la señal integrada con IEMG. La Figura 3.15 muestra la señal Raw EMG, rectificada e integrada; las tres señales obtenidas para un caso especial de una muestra del canal seis.

3.6.3. Participantes.

En nuestras pruebas participaron veintiún voluntarios cuyos datos se muestran en la Tabla 3.4. Las edades de los voluntarios estaban comprendidas entre 21 a 50

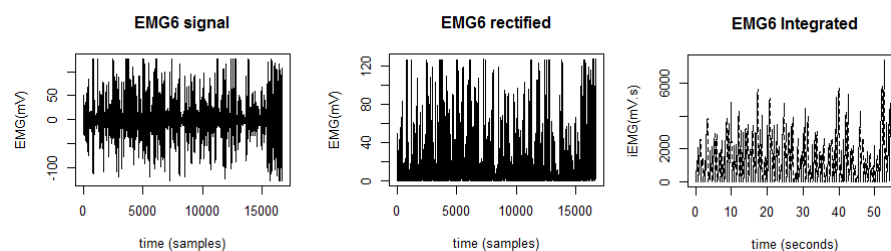


Figura 3.15: Señales EMG al formar el gesto (i) y sus características.

años (5 mujeres y 16 varones) ($Media = 31,57$, $DS = 7,76$). Todos los voluntarios dijeron conocer el manejo de un dispositivo móvil y tenían experiencia en el uso de vídeo juegos. Tres voluntarios eran zurdos. La circunferencia de sus antebrazos alcanzó una media en hombres de $27,85$ ($DS = 1,98$) y en mujeres 26 ($DS = 2,16$). El rango de circunferencia del antebrazo del voluntario es necesario porque, la pulsera gestual debe acoplarse adecuadamente al antebrazo del usuario con todos los sensores distribuidos radialmente.

Calibración y ajuste

En ambos experimentos, *On-line* y *Off-line*, el brazalete gestual fue ubicado y alineado como se indica en la Figura 3.16. Además, debemos indicar que la configuración por defecto del brazalete se mantuvo durante los experimentos debido a que el software del fabricante no fue utilizado para tareas de calibración ni de lectura. En nuestra pruebas, la ubicación del sensor número cuatro (4) fue de extrema importancia, pues mediante aquel, se lograba una ubicación relativamente exacta en el antebrazo, directamente sobre el músculo Extensor Carpi Radialis Longus del usuario. Esto se observa mejor en la Figura 3.16. Después de un gran número de experimentos preliminares, se observó que la mejor ubicación para el brazalete gestual era: Primero el brazo debe ser extendido hacia adelante, paralelo al suelo manteniendo la palma de la mano extendida y vertical al suelo. Posteriormente, el brazalete debía colocarse controlando que el sensor 4 este colocado sobre el músculo Extensor Carpi Radialis Longus, manteniéndolo alineado con el pulgar extendido verticalmente. Esta alineación de brazalete evitó la pérdida de calibración cuando el brazalete se intercambia entre los usuarios. Además, el proceso de alineación manual permitió, evitar la recalibración por software al tener que cambiar de usuario durante las pruebas. La recalibración por software, mediante el utilitario ofrecido por el fabricante, al parecer permite la adaptación de los parámetros máximos y

Tabla 3.4: Características de los voluntarios participantes en esta investigación

Voluntario	Edad	Sexo	Mano	Peso	Estatura	LA	CA
No.	Años	M/F	D/S	Kgs	m	cm	cm
1	40	M	D	91,7	1,71	47	31
2	31	M	D	71,4	1,69	49	25
3	28	M	D	65,9	1,72	45	27
4	24	M	Z	69	1,76	48	31
5	39	M	D	86,7	1,63	46	28
6	23	M	Z	72,5	1,75	45	26
7	39	F	D	72,3	1,49	42	26
8	36	M	D	98,4	1,81	49	29
9	24	F	D	60,1	1,7	43	24
10	23	F	D	73,5	1,76	45	25
11	25	M	D	90,1	1,73	49	29
12	21	M	D	86	1,84	50	26,5
13	24	M	D	85,4	1,76	51	30
14	28	M	D	84	1,69	46,5	29
15	28	M	D	86,4	1,73	50	27
16	30	F	D	62,6	1,56	43	24
17	38	M	D	96,4	1,69	45	28
18	50	M	D	94,6	1,58	42	28
19	39	M	D	82,6	1,65	43	29
20	35	M	D	86	1,73	47	26
21	38	F	Z	89,3	1,69	49	29

LA, Longitud del antebrazo; CA, Circunferencia del antebrazo; Z, Zurdo; D, Derecha; M, Masculino; F, Femenino.

mínimos de amplitud de la señal. Si bien esto ayuda a una mejor identificación del gesto, hemos tratado de obviarla dejando aquella que viene por defecto, y trabajando en nuestros experimentos con la alineación sugerida. Además, esto también permitió mantener el reconocimiento gestual en niveles adecuados independientemente del cambio de usuario.

La ubicación correcta de la pulsera gestual (Figura 3.16) garantiza una adecuada captura del gesto y lecturas lo más similares posibles entre cada usuario. Se busca mediante esta alineación, tratar de mantener la ubicación de los sensores en correspondencia con los músculos del antebrazo en cada usuario. Esta correspondencia muscular, concuerda con lo explicado en la sección (3.2.4), en relación a los posibles problemas de medir señales EMG. Este detalle es clave para lograr medidas y patrones estables, incluso en un mismo usuario. Hacemos referencia a un

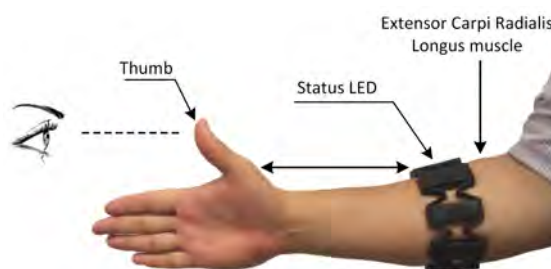


Figura 3.16: Ubicación adecuada de MYO en el antebrazo del usuario.

mismo usuario, pues al sacarse y volverse a poner el dispositivo la misma persona, no deberían verse afectadas las lecturas de la señal EMG provenientes de ella y deberían mantenerse los patrones de identificación del gesto.

3.6.4. Experimentos off-line

En los experimentos fuera de línea, el investigador explicó brevemente (verbal y gráficamente) antes de cada ensayo, al voluntario, el gesto que debía realizar. El área de pruebas consistió en un gran escritorio y una cómoda silla con respaldo. Los voluntarios usaron el brazalete en el antebrazo de la mano dominante (3 personas eran zurdas). El procedimiento para obtener el dataset (conjunto de datos) se realizó de la siguiente manera:

El tiempo de muestreo para cada gesto fue de 10 segundos. El inicio y el final del ensayo estaban controlados por software, aplicación especialmente diseñada para la captura de datos, por medio de las opciones disponibles. Dos o tres segundos antes de iniciar el ensayo, se le pedía al usuario formar el gesto, al igual que mantenerlo hasta uno o dos segundos después de que concluyera la captura de datos. El tiempo de prueba aproximado por usuario alcanzó los 30 min como mínimo, pero podía prolongarse considerablemente por cansancio o molestias, en cuyo caso era recomendable reposar un momento para luego continuar. Un total de 273 gestos fueron analizados (21 usuarios x 13 gestos) alcanzando un dataset total de 581.000 muestras. Al finalizar el test, el software emitía un aviso sonoro (melodía), guardaba la información y preparaba la siguiente toma de datos indicando el gesto a ser realizado. El investigador controlaba la correcta ejecución de cada gesto y su conservación por el tiempo indicado. Este ciclo fue repetitivo, para todos los gestos y reiniciado por cada usuario. Se daba un breve descanso al usuario, entre cada gesto, para no fatigar los músculos.

Los datos Raw EMG obtenidos, fueron capturados usando un programa escrito en lenguaje C. Dicho programa recogía los datos provenientes de la pulsera gestual, a través de sus ocho canales (1 por sensor) y los almacenaba en un archivo .CSV. También, usando el mismo software, fue posible observar la señal EMG (Electromiografía) en todos los canales, como nos muestra la captura de pantalla en la Figura 3.17. Una captura total de todos los canales, realizada con el objetivo de lograr un mejor enfoque y detalle del comportamiento del gesto usando señales EMG, puede ser encontrada en el Apéndice de esta tesis.

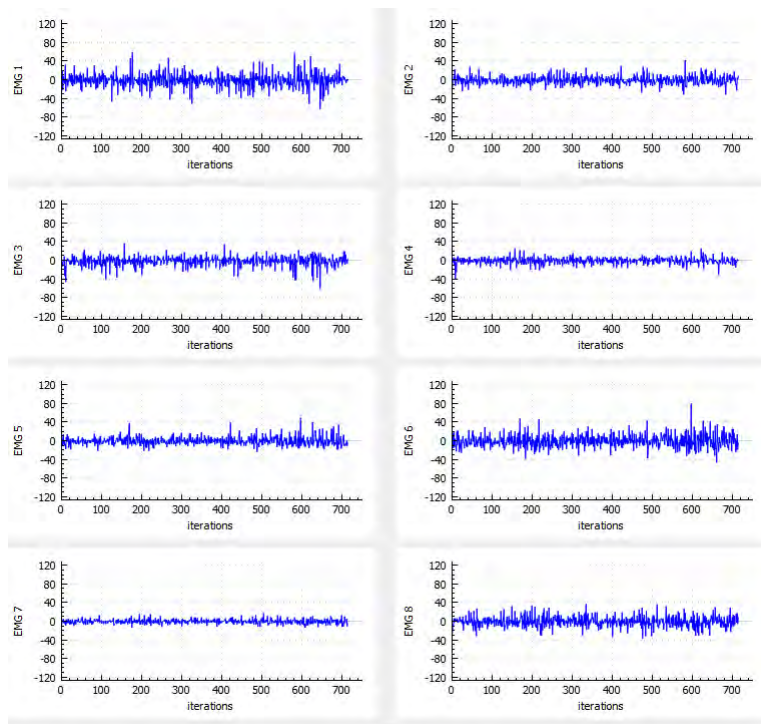


Figura 3.17: Señales EMG, en los ocho canales, obtenidas al formar el gesto (i).

En este sentido, debemos insistir en que el brazalete capta las señales EMG por sus ocho canales EMG_n donde $n = (1, \dots, 8)$ por cada gesto. Los datos Raw (en bruto) capturados por cada gesto en nuestra aplicación, se muestran en la Figura 3.18 desde dos perspectivas diferentes. La Figura 3.18.a muestra el conglomerado de datos en su totalidad agrupados por gesto y su afectación en cada canal. Todos los datos registran una distribución de amplitud, conforme las características del dispositivo. Mientras la Figura 3.18.b, muestra la nube de datos pero cada gesto, agrupando la muestra total para cada gesto en todos los usuarios. Cada nube se ha pintado de un color diferente para lograr una mejor idea de la distribución de

la amplitud del gesto en todos los usuarios.

En la Figura 3.18, en su conjunto, permite apreciar el comportamiento de la señal EMG en todos los sensores distribuidos alrededor del brazo al realizar un gesto. Es notorio entonces, que un sensor capta más información que otro y esto se debe justamente al hecho de que, **la actividad EMG no es la misma en cada músculo ni en cada sensor**. El conjunto de datos completo está disponible en (WEB).

Las señales Raw EMG correspondientes a cada gesto y representadas por cada canal se muestran en la Figura 3.18. Esto permite tener una idea de la similitud y la diferencia entre cada gesto y los datos generados por cada uno de ellos. Todos los electrodos están involucrados, en mayor o menor medida, con los músculos que generan el gesto, en un lugar específico.

Para el estudio numérico, se descartaron dos segmentos de la muestra para cada gesto: a) El primer segmento, equivalente a los 2s iniciales de la muestra de datos fue rechazado para evitar el ruido producido cuando el voluntario apenas está tratando de formar el gesto; b) El segundo segmento, los 2s finales de la muestra evita incluir datos por agotamiento o fatiga muscular. Del segmento de sobrante (6s), se ha tomado un sub-segmento equivalente a 3s (1.281 muestras x gesto) haciendo un total de 16.653 muestras (todos los gestos) para análisis, entrenamiento (70 – 30) y pruebas. Este proceso es similar al aplicado por [13] en un estudio similar.

Las señales EMG además, fueron filtradas por software con el fin de eliminar los artefactos introducidos a la señal y evitar la contaminación (ruido) existente. El filtro aplicado fue, el filtrado de paso alto. La frecuencia de corte del filtro se fijó a $128Hz$ (el rango de valores en el brazalete varía de +128 a -128). Se aplicó la extracción de características en el dominio del tiempo, indicadas en la ecuación 3.25 y ecuación 3.26 indicadas en la Sección 3.6.2, y todas fueron calculadas sobre la señal EMG capturada y agregadas al dataset. Las extracción de características FW-REMG y IEMG se muestran en la Figura 3.19 y fueron calculadas para cada canal (cada sensor). La integración de la señal EMG se calculó con punto de reinicio a 200 muestras. Los vectores resultantes (2 características x 8 canales) se añadieron al conjunto de datos antes de ser enviados al clasificador. En este caso, no hemos utilizado la técnica de ventana superpuesta por las siguientes razones: la ventana adyacente podría introducir un retraso en el sistema de clasificación [47], y la superposición de ventana puede aumentar el tiempo de procesamiento [7]. Además, porque el dispositivo aplica técnicas de amplificación y filtrado. La Figura 3.19

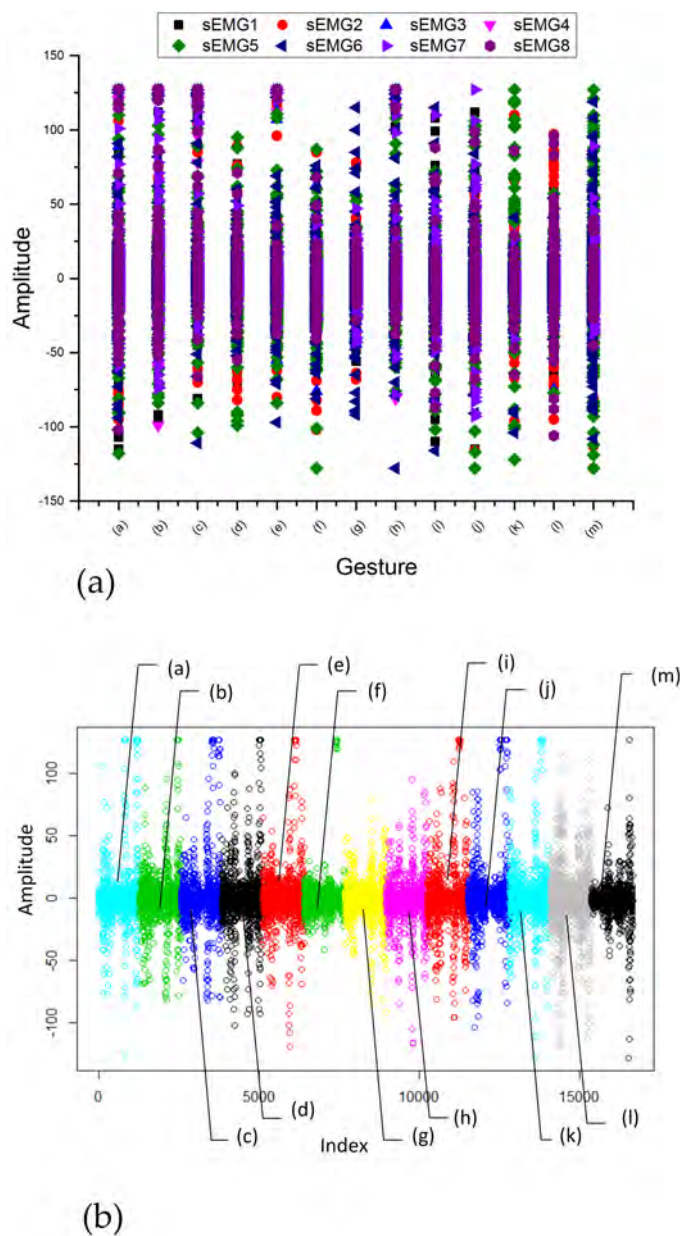


Figura 3.18: Lecturas recibidas por los ocho sensores sEMG.

muestra en : a) el Raw de la señal EMG recogida por el canal 1 para un mismo gesto en toda la muestra analizada; b) La señal EMG aplicada el filtro paso-alto; c) La misma señal rectificada en ancho de banda completo; d) La señal EMG con integración con punto de reset a 200 muestras.

Los clasificadores, SVM y k-NN, fueron entrenados utilizando un conjunto de

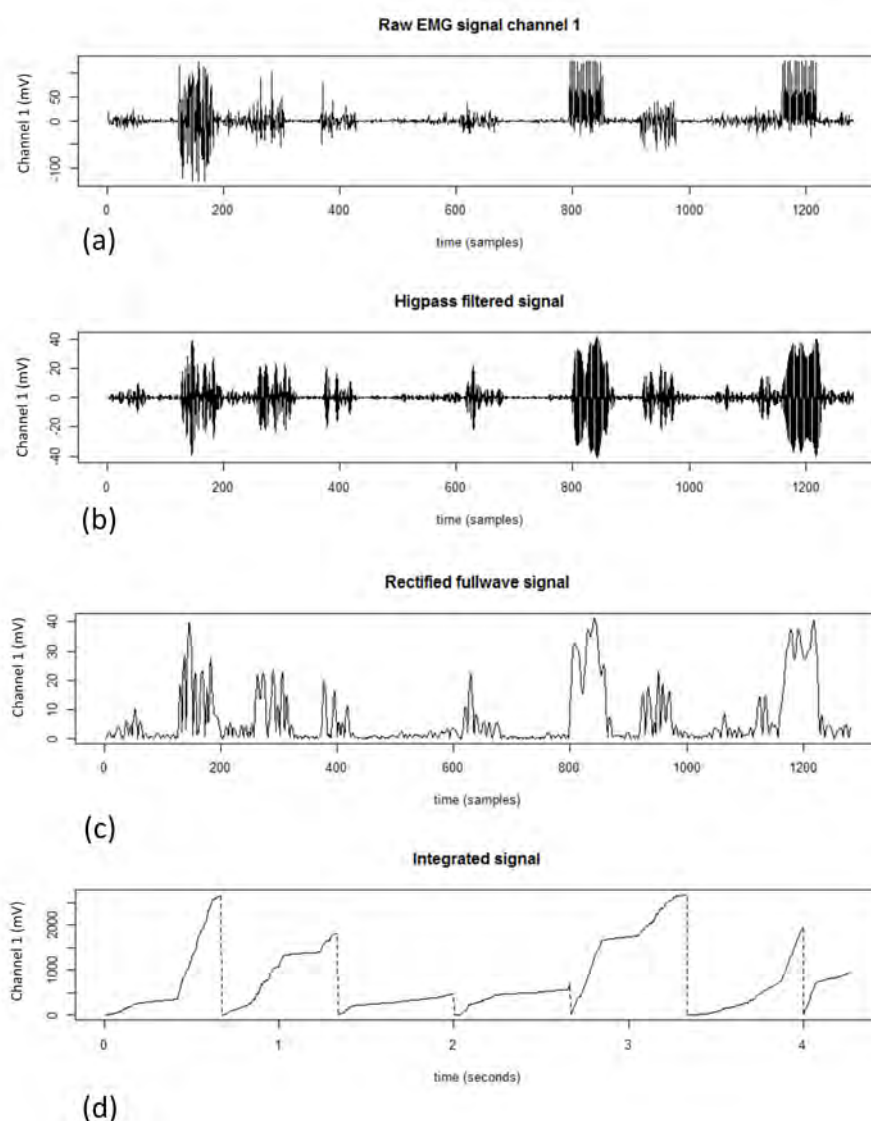


Figura 3.19: Filtrado y selección de características aplicada a la señal EMG.

entrenamiento mientras que un conjunto de validación se utilizó para verificar los parámetros de cada algoritmo. SVM requiere de un conjunto de parámetros de los cuales el parámetro kernel es el más importante debido a que define la estructura del espacio e alta dimensionalidad donde se puede encontrar un máximo margen de separación del hiperplano. En SVM, se usó la función `TUNE()` del paquete `e1071`, la misma que genera un conjunto de hiper-parámetros o métodos estadísticos usando un algoritmo de búsqueda para determinar los mejores parámetros tanto para

gamma γ como para la variable de regularización de costo C . En R el parámetro encontrado para lograr la mejor separación fue con $C = 1$ y el parámetro por defecto de gamma $\gamma 0,5$ [203]. Para poder ubicar la mejor opción de clasificación usando los parámetros encontrados para C y γ y lograr el mejor intervalo para la mejor generalización, se usó la validación cruzada (cross-validation) que en nuestro caso estuvo en 10. Entonces, el conjunto (C, γ) fue verificado para encontrar la mejor estimación de exactitud (accuracy, en inglés). El mejor parámetro usado para el entrenamiento, permitirá general el modelo final. Los diferentes valores de C y γ usados se describen en la Tabla B.1.

Se experimentó con los datos, usando varios tipos de kernel con SVM como algoritmo de clasificación. Se estableció una proporción de datos para entrenamiento y prueba, correspondiendo a un 70% de datos para entrenamiento y un 30% de datos para prueba, siguiendo la recomendación dada por [203]. Así también, la Tabla B.1, permite observar los valores alcanzados para exactitud en cada uno de los kernel usados. Este proceso fue iterativo para cada clasificador, en el caso de SVM para cada kernel, y para k-NN, para los diferentes valores de k. Las proporciones del conjunto de datos fueron 2/3 para el entrenamiento y 1/3 para la verificación de SVM, y un 4/5 entrenamiento y 1/5 para la verificación en k-NN.

El dataset fue procesado en R (Project for Statistical Computing) sobre un servidor con OS Ubuntu 16.14 cuyas características de hardware fueron 26 GB en RAM, Quad Intel Xeon E5-2680 v3, con 12 núcleos cada uno, trabajando en 2.5 GHz. Los clasificadores SVM y k-NN se construyeron utilizando el paquete e1071, EMG biosignal, entre otros disponibles en R, para el procesamiento de datos.

Algoritmo SVM SVM soporta clasificación multi-clase para realizar la clasificación. SVM busca un hiperplano con el mayor margen de separación posible para clasificar los diferentes conjuntos de datos por cada gesto. El clasificador SVM procesó los datos utilizando diferentes funciones núcleo tales como: Función Básica Radial Gausiana (Gaussian Radial Basis Functions, RBF), Polinomial (Polynomial) y Sigmoide (Sigmoid). Cada núcleo alcanzó diferentes niveles de exactitud a partir de los dos parámetros principales: C y γ (gamma). Con el fin de afinar estos parámetros y encontrar aquellos que permiten una mejor generalización, hemos utilizado varios valores de búsqueda en C y γ a través de la validación cruzada por 10 (10-cross-fold validation). Los parámetros se seleccionaron sobre la base de varios valores de $C \in [0.0001, 0.001, 0.01, 0,1, 1, 10, 100, 1000, 10000]$ y $\gamma \in [0.00125, 0.0625, 0.0125, 0.125, 0.25, 0.5, 0.652, 0.75]$. El mejor parámetro se utilizó para la optimización de C en el conjunto de entrenamiento. La Tabla B.1 muestra la

exactitud alcanzada para cada núcleo.

Algoritmo k-NN El conjunto de datos se clasificó utilizando el algoritmo k -NN, con $k \in 1, 3, 5, \dots, 27$. El valor de k no debe ser demasiado grande ni demasiado pequeño, por las razones expuestas en la Sección 3.5.3. Para superar este problema, buscamos encontrar el valor k óptimo a través del análisis cuantitativo, cambiando el valor k de $[1, \dots, 27]$ usando la Distancia Euclídea. Los mejores resultados obtenidos fueron aquellos encontrados para $k = 7$ usando la ecuación 3.24 y algoritmo 2. Sin embargo, la exactitud de k-NN puede sufrir una degradación apreciable debido a su baja tolerancia al ruido como se indica en la Tabla 3.3 y en la Sección 3.5.3.

3.6.5. Experimentos on-line

En los experimentos on-line, se evaluó el sistema de clasificación y reconocimiento gestual. El mismo brazalete utilizado para las pruebas *off-line* se utilizó sin cambios en su configuración. Se realizaron cinco pruebas de reconocimiento para cada gesto. El software creado por los investigadores guardó automáticamente los intentos del usuario y la respuesta de reconocimiento del sistema, asociándolos mutuamente, en un archivo para procesamiento posterior. El tiempo de reconocimiento se calculó en milisegundos y se midió desde el envío de los datos al clasificador hasta obtener la respuesta de clasificación. El brazalete se colocó de acuerdo al método indicado para ubicación del dispositivo, diseñado en esta investigación, y concebido para alcanzar un óptimo reconocimiento. Se siguieron además las recomendaciones dadas en los experimentos *off-line*.

3.7. Resultados y discusión

3.7.1. Experimentos off-line

Algoritmo SVM

Inicialmente se experimentó con el dataset sin incluir la extracción de características, para verificar si el Raw proveniente de la pulsera gestual requiere la introducción características para distinguir adecuadamente el gesto. Los resultados de dicho experimento se pueden observar en el Anexo B de esta tesis. En base a aquellos resultados podemos decir que, el Raw de datos no es suficiente para identificar adecuadamente el gesto.

Posteriormente, y continuando con el estudio del dataset agregamos la extracción de características expuestas en la Sección 3.6.2. Los resultados cambiaron dramáticamente.

SVM con núcleo Polinomial alcanzó una precisión del 98,81 %. El kernel RBF logró una precisión del 96,22 %, superior al 79,36 % logrado en un trabajo anterior [143]. Este aumento tan notorio de la exactitud se debe al hecho de haber incluido características en el dominio del tiempo. La diferencia entre el núcleo RBF y polinomial es de apenas un 2,59 %, lo cual nos presenta dos alternativas de implementación por su pequeña diferencia. En este sentido, la exactitud media obtenida por el clasificador k -NN fue del 83,90 %. Los resultados para el núcleo Sigmoidal se mantuvieron bajos, e incluso inferiores a nuestras expectativas iniciales (apenas alcanzaron el 52,66 %). Esto puede indicar que este tipo de kernel no se ajusta al conjunto de datos utilizado.

Las matrices de confusión de SVM con núcleo polinomial (Figura 3.20.a) y k -NN (Figura 3.20.b) se muestran en la Figura 3.20.

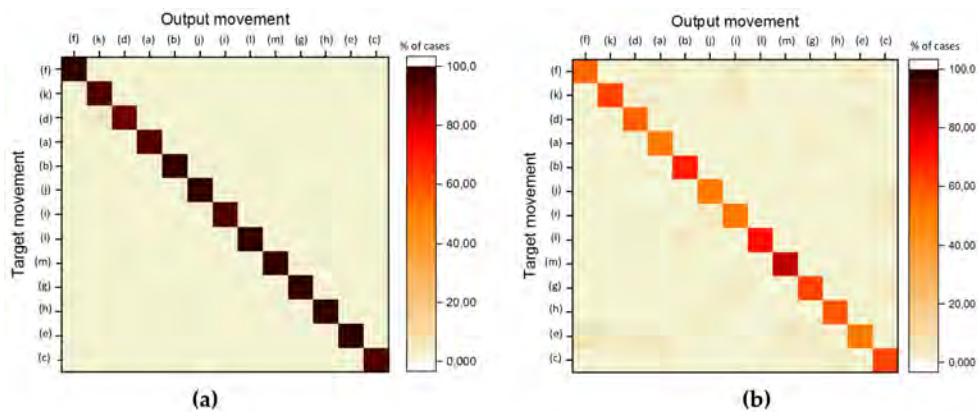


Figura 3.20: Matrices de confusión de los clasificadores usados.

La baja exactitud del algoritmo k -NN se compensa, de alguna manera, por la facilidad de implementación debido a su simplicidad. Además, no requiere un tiempo de aprendizaje significativo. Sin embargo, si se requiere una alta exactitud, como en nuestro caso, el núcleo Polinomial y RBF son las mejores alternativas. La exactitud obtenida por ambos kernels se convierte en un modelo altamente recomendable para un dispositivo wearable como el usado en este estudio. Los resultados obtenidos, basados en características que requieren una carga computacional mínima, nos hacen suponer que es posible la implementación de un sistema de reconocimiento de gestos utilizando un sistema embebido.

La Tabla 3.5 muestra los resultados de los experimentos para los diferentes tipos de kernel de SVM. Como puede observarse, es evidente que el núcleo Polinomial proporcionó un mejor nivel de exactitud sobre que los kernels RBF y sigmoides. Esta Tabla, también muestra los rangos de (C, γ) .

Tabla 3.5: Exactitud media alcanzada con Kernels SVM junto a parámetros C y γ

Kernel	Parámetros	Exactitud media
RBF	$C = [0.0001; 0.001; 0.01; 0.1; 1; 10; 100; 1000; 10000]$, $\gamma = [0.00125; 0.0625; 0.0125; 0.125; 0.25; 0.5; 0.625; 0.75]$	96,22 % (DS=1,16)
Polynomial	$C = [0.01, 0.1, 1, 10]$, $\gamma = [0.125, 0.25, 0.5, 0.625, 0.75]$	98,81 % (DS=0,72)
Sigmoidal	$C = [0.01, 0.1, 1, 10]$, $\gamma = [0.125, 0.25, 0.5, 0.625, 0.75]$	52,66 % (DS=3,10)

La tasa de error, para el núcleo RBF fue de 0,038 mientras que para el núcleo Polinomial fue de 0,012, con los mismos parámetros.

Algoritmo k -NN

El algoritmo k -NN fue probado con varios valores de k ($k \in 1, 3, \dots, 27$), obteniendo la mejor exactitud para $k = 7$ usando el mismo conjunto de datos. El algoritmo k -NN alcanzó un 83,90 % (DS= 4,23) de exactitud media. La tasa de error alcanzó 0,161, con porcentajes de exactitud calculados para cada gesto, como se muestra en la Tabla 3.6. En este sentido, los niveles de precisión son aceptables, ya que todos superaron el 80 %, excepto para el gesto (e). Estos resultados son muy similares a los obtenidos por Kim et al. [88], trabajando con movimientos de muñeca (arriba, abajo, derecha, izquierda y descanso) y también basado en un sistema sEMG inalámbrico pero de dos canales y logrando una exactitud media del 84.9 %. Además, también observamos algunas limitaciones en los resultados de reconocimiento reportados en la matriz de confusión (Figura 3.20 y Tabla 3.6). Varios gestos son relativamente más difíciles de distinguir, probablemente debido a la alta similitud de movimiento, como se muestra en la Figura 3.14 del lenguaje gestual y la Figura 3.18 de densidad de lecturas por cada canal EMG.

La exactitud de reconocimiento (Tabla 3.6) en algunos gestos es buena (por ejemplo, gestos b, l, y m), en otros es baja (por ejemplo, gestos e, f), lo que afecta a la exactitud media. Esta disminución en la exactitud causada por algunos gestos, quizá cuyas lecturas son muy similares (por ejemplo, a, f, e, etc.), introducen ruido en el sistema.

Tabla 3.6: Exactitud alcanzada por algoritmo k -NN y SVM

Gestos		Algoritmos			
Index	Nombre	k-NN	RBF	SVM	
				Polinomial	Sigmoide
(a)	Open hand	81.37	95.94	98.32	50.88
(b)	Close hand	91.77	97.77	99.17	55.80
(c)	One	82.82	95.54	98.41	49.87
(d)	Two	80.77	94.53	97.09	50.77
(e)	Three	76.25	95.24	99.10	50.91
(f)	Four	80.5	95.62	99.20	53.86
(g)	Six	84.76	96.71	99.13	54.23
(h)	Seven	83.66	96.93	99.48	54.48
(i)	Eight	83.24	94.94	98.02	47.64
(j)	Nine	83.5	97.00	99.60	50.78
(k)	Ten	83.01	95.23	98.44	52.48
(l)	Palm inwards	89.11	97.19	99.04	53.04
(m)	Palm out	89.92	98.25	99.56	59.87

Resumiendo, los dos algoritmos de clasificación utilizados, SVM y k -NN, han demostrado ser adecuados para trabajar con la señal EMG procedente del brazalete. SVM con núcleo Polinomial consiguió la mejor exactitud media de 98,81 % con una diferencia mayor de 14,91 % con respecto a k -NN. Además, tuvo un mejor desempeño y logró mejores resultados individuales y globales. El brazalete funcionó sin problemas y capturó todos los datos de los sensores sEMG.

3.7.2. Experimentos on-line

Con los experimentos en línea, podemos analizar las capacidades de la banda gestual MYO el cual se proyecta como dispositivo ideal para la construcción de nuevas interfaces basadas en gestos de la mano. Para lograr la identificación de un biblioteca mucho más amplia que la que provee el fabricante, hemos implementado un sistema de reconocimiento gestual basado en clasificadores SVM. Nuestro sistema además, libera al brazalete MYO de su SDK, ya que los datos brutos proporcionados son capturados directamente por medio del protocolo Bluetooth y gracias a la reciente liberación de su codificación por parte del fabricante.

Arquitectura propuesta para interfaces gestuales

Basado en los resultados del análisis off-line, que demuestran que un sistema de reconocimiento gestual es posible usando clasificadores aplicados directamente al Raw EMG capturado por MYO, proponemos una arquitectura para una interfaz gestual. Nuestra arquitectura se describe en la Figura 3.21 y está compuesta por dos capas principales: Capa de captura de señal (Signal Capture) y Capa de procesamiento de datos (Data Processing).

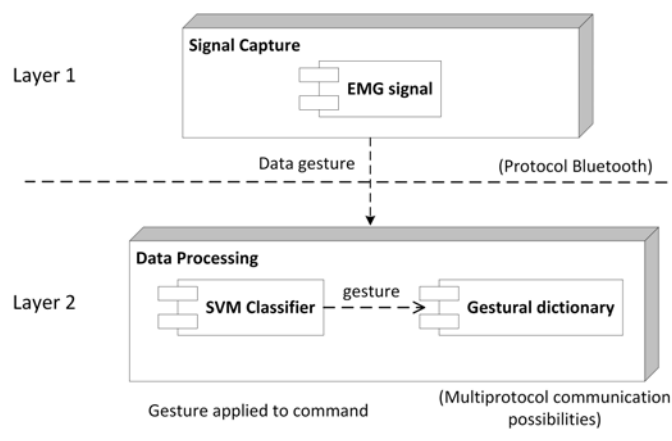


Figura 3.21: Arquitectura para sistemas de reconocimiento gestual basada en clasificadores.

Capa de captura de señal La capa de captura de señal es responsable de capturar las señales EMG. Los sensores sEMG de la pulsera MYO captan las señales por cada uno de sus ocho canales. A continuación, las señales EMG se transmiten a la capa de procesamiento de datos, utilizando un protocolo de comunicación tal como Bluetooth Low Energy (BLE). Los detalles de la pulsera gestual fueron presentados en detalle en la Sección 3.3.

Capa de procesamiento de datos La capa de procesamiento de datos, se encarga de lograr la clasificación y el reconocimiento adecuado del gesto. Esta capa está soportada por un dispositivo informático con una capacidad de cómputo suficiente (sistema embebido o Pc) para implementar el clasificador con una frecuencia de al menos 20 Hz [64]. Una vez que el gesto es enviado al clasificador, éste lo identifica comparando los patrones gestuales en términos de vectores de soporte (SV, siglas en inglés) almacenados en el diccionario gestual.

El sistema de reconocimiento gestual identifica el gesto como un comando, lo cual implica la asociación de un comando o una acción, a cada gesto reconocido (por

ejemplo, una mano cerrada podría significar detener una tarea o un dispositivo). En esta capa, es importante que el tiempo de cálculo sea el más pequeño posible para un óptimo reconocimiento y adecuado funcionamiento.

Implementación

Basado en los resultados alcanzados de los experimentos off-line observados (Sección 3.6.4), implementamos un sistema de clasificación y reconocimiento gestual en tiempo real. Dicho sistema implementaba la biblioteca gestual descrita en la Sección 3.6.1. Para la implementación, hemos utilizado la arquitectura abierta propuesta en esta Sección (Sección 3.7.2), la cual incorpora todos los elementos necesarios para lograr una aplicación en tiempo real que implemente una interfaz gestual basada en los movimientos de la mano. Considerando la complejidad algorítmica y la exactitud alcanzada por cada uno de los algoritmos estudiados, se implementó un sistema de reconocimiento gestual en tiempo real basado en SVM.

El sistema de reconocimiento se ejecuta en un sistema embebido basado en un empotrado de alto poder computacional llamado Raspberry PI 3. Dicho empotrado incorpora un Broadcom BCM2387 SoC (sistema en un chip) con un microprocesador Cortex A53 de cuatro núcleos ARM trabajando a 1,2 GHz, GPU Broadcom VideoCore IV Dual Core y 1 GB de RAM LPDDR2. Este dispositivo inicializa y carga automáticamente el controlador de software. En nuestro caso, hemos instalado Android Things de Google (R) como sistema operativo. El empotrado recibe los datos de las señales EMG enviadas desde el brazalete a través de protocolo Bluetooth, con algunas características propias patentadas por Thalmic Labs. El sistema, toma directamente el Raw, agrega las características y envía los datos al clasificador el cual está programado en Android y, a través de él, se logra el reconocimiento gestual. Para hacerlo, el sistema incorpora los vectores de soporte necesarios, encontrados durante Proceso Experimental.

El software implementado, usando esta arquitectura, es completamente independiente del software proporcionado por el fabricante del brazalete gestual, cuyas características fueron descritas a fondo en la Sección 3.3 de este capítulo. El brazalete logra, con carga máxima de sus dos baterías, una autonomía de hasta catorce horas continuas de uso con breves descansos. Sin embargo, aquello que más nos preocupó fue el hecho de lograr un perfecto ajuste de reconocimiento, independiente del usuario. Para lograrlo se siguieron las recomendaciones de calibración y ajuste dadas en la Sección 3.6.3.

Las pruebas de operabilidad y reconocimiento mostraron un rendimiento ade-

cuado del sistema de reconocimiento de acuerdo a lo previsto. Esta experiencia sugiere que, mediante el uso y adaptación de una interfaz de reconocimiento gestual, usando a SVMs como clasificador, se puede lograr un aprendizaje progresivo y de fácil aceptación por parte del usuario. Sin embargo, en algunos casos fue necesario realizar el gesto con mayor persistencia hasta lograr su reconocimiento (p. Ej., C, d, e, h y j). Esto probablemente, debido a la activación simultánea del mismo grupo muscular en el antebrazo, que se ilustra en la Figura 3.22.a, y que se presenta además, por la geometría del diseño (radial) de la pulsera gestual. Esto se analizó con mayor detalle en el Proceso Experimental (Sección 3.6). No hubo diferencias significativas entre los usuarios diestros y aquellos zurdos, pero si se pudo apreciar un mejor ajuste de la pulsera, en aquellos usuarios cuya Circunferencia del Antebrazo (CA) era mayor a 24 cm $CA > 24cm$. En los voluntarios con CA menor ($CA < 24cm$), se usaron los eslabones de ajuste, para reducción de diámetro de la pulsera gestual, y se mantuvieron todas las demás recomendaciones.

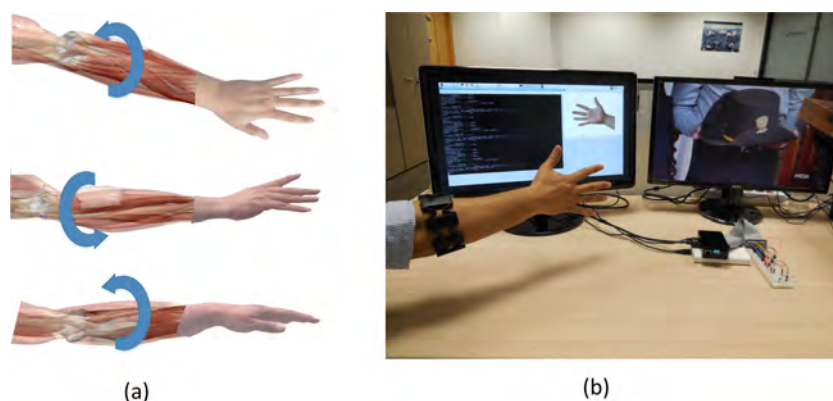


Figura 3.22: Sistema de reconocimiento en operación. (a) Ilustración de los músculos del antebrazo. (b) Pulsera MYO en brazo derecho de usuario.

Debido a la ubicación de los electrodos (que rodean el antebrazo) y a pesar de su geometría tratando de evitar la diafonía, la pulsera gestual, al igual que otros dispositivos que trabajan con bioseñales, también recoge ruido del medio ambiente [12], por lo que el filtro y selección de características son indispensable debido a la amplia librería gestual utilizada.

Experimentalmente, el tiempo promedio empleado en la identificación de gesto fue determinado en 3,8 ms con una variación de 2,5 ms a 7,8 ms. Este margen de tiempo que es muy adecuado para el procesamiento en tiempo real y muy en especial en un sistema de reconocimiento gestual [13]. Esto es un logro significativo, dado que el mayor inconveniente de los clasificadores es el tiempo de cálculo neces-

rio para alcanzar su meta. En este sentido, un tiempo similar aunque quizá un poco más bajo, fue alcanzado por Al-Timemy et al. [3] quienes combinaron el Análisis Discrecional Ortogonal de Vecino Difuso (Orthogonal Fuzzy Neighborhood Discriminant Analysis, OFNDA) y SVM para lograr clasificar 12 tipos de movimiento en un tiempo igual a 2,4 ms. Esto confirma que la solución propuesta en esta investigación es muy aceptable para el diseño y construcción de sistemas de reconocimiento gestual basados en un sistema embebido, que usen el wearable gestual.

Los ensayos determinaron que el alcance máximo de la conexión Bluetooth entre la pulsera gestual y el sistema de reconocimiento gestual alcanzó hasta 8 metros en un área cerrada. Además, el consumo de energía del dispositivo wearable fue mínimo. Benatti et al. [12] estimó el consumo de energía para un dispositivo similar en aproximada 86 mW.

Por último, debemos indicar que el sistema de reconocimiento gestual implementado abre nuevas alternativas de aplicación en Interfaces Naturales de Tipo Gestual basadas en señales EMG desde las cuales es posible controlar cualquier sistema o dispositivo eléctrico y electrónico.

3.8. Conclusiones

En este capítulo hemos realizado un análisis del gesto capturado por sensores, aplicando en este caso, las señales EMG. Hemos revisado los principios anatómicos y formulación científica de la señal electromiográfica, para su aplicación y uso en la captura del gesto. Se ha analizado los procesos de adquisición de señales y los tipos de electrodos existentes, optando por los de tipo superficial por ser los menos invasivos. Hemos revisado las técnicas de machine learning, haciendo un profundo análisis de aquellas más usadas en sistemas de reconocimiento del gesto de la mano usando señales EMG. Se pudo determinar que los algoritmos SVM y k -NN son los más usados en este tipo de señales y con los cuales se experimentará en este análisis. Logramos también determinar las características y principales ventajas de usar un dispositivo wearable de bajo costo conocido como brazaletes gestual MYO, el cual será usado en la captura del gesto. Dicho wearable cuenta con ocho sensores sEMG, demostrando ser el dispositivo portátil más adecuado para capturar señales de EMG superficiales y el que mejor se adapta a múltiples usuarios. Hemos definido un método de calibración, dada en relación a ubicación y alineación, el cual ha sido usado tanto para pruebas *Off-line* como en pruebas *On-line*. No fue necesario realizar una calibración por software y se mantuvo aquella por defecto

proporcionada por el fabricante. La pulsera gestual demostró ser práctica y no demasiado intrusiva para lograr el reconocimiento gestual, logrando aumentar la movilidad y comodidad del usuario, en comparación con dispositivos estacionarios tales como cámaras.

En nuestro análisis, se evaluó una amplia biblioteca gestual correspondiente a trece gestos estáticos con la participación de veintiún voluntarios. Se utilizó un gran amplio dataset mediante el cual se estableció la exactitud de clasificación de SVM y k -NN. El análisis de extracción de características determinó que un adecuado reconocimiento gestual y una mejor clasificación podría lograrse mediante la adición de características intrínsecas de la señal EMG, en el dominio del tiempo, tales como FW-REMG y IEMG. Estas características se han añadido a los datos Raw EMG por defecto recibidos desde el dispositivo de captura gestual portátil para lograr una mejor clasificación y reconocimiento. En nuestros experimentos, el algoritmo SVM logró una exactitud promedio del 98,81 % usando el núcleo Polinomial, el 96,22 % usando el núcleo RBF y el 52,66 % usando el núcleo Sigmoidal. Mientras que el algoritmo k -NN alcanzó una precisión media de 83,90 % y fue usado para contraste.

Se implementó un sistema de reconocimiento y clasificación gestual basado en señales EMG y construido sobre una arquitectura abierta de dos capas, propuesta e implementada en esta investigación. El sistema implementado fue evaluado realizando experimentos en línea con el fin de verificar, tanto sus las capacidades operativas como aquellas de reconocimiento gestual. Los resultados indican que, un sistema embebido de gran potencia es capaz de ejecutar el algoritmo de clasificación, basado en vectores de soporte, y obtener una adecuado reconocimiento en tiempo real. El sistema embebido alcanzó un tiempo de procesamiento promedio de 3,8ms, medido desde el envío de datos al clasificador hasta que aquel entregó una respuesta. Por último, podemos decir que este trabajo evidencia las grandes posibilidades de aplicación de las señales EMG, al diseño e implementación de interfaces naturales de usuario de tipo gestual usando un wearable de bajo costo.

Capítulo 4

Análisis de la Interfaz gestual vs. la pantalla táctil

La superficie de contacto es una forma de interacción que en la actualidad ha logrado excelentes resultados, al amparo de los avances tecnológicos, especialmente en el diseño e implementación de las llamadas *pantallas táctiles* (Touchscreen). Su popularidad fue total, al ser incorporadas en forma masiva como una forma de interfaz nativa en dispositivos móviles como teléfonos inteligentes, tabletas, relojes, etc. Aunque inicialmente fueron concebidas para sistemas industriales, las pantallas táctiles han sido incorporadas a tantos dispositivos, y tal ha sido su impacto, que se considera una forma rápida y efectiva de interacción con sistemas y equipos de cómputo. Pero en muchos casos, las limitaciones que impone (precisión, agudeza visual, definición adecuada de colores, entre otras), contrasta con el objetivo de interacción natural. Se hace indispensable usar el dedo, o dedos del usuario, para reemplazar al toque de tecla o clic del ratón, para lograr una entrada de datos. Si bien la guía auditiva ofrece posibilidades, no es menos cierto que exige mayores facultades (visión-oído) y esta sujeta al lenguaje (idioma) y comprensión del sujeto. Por ello, nuevamente el gesto se impone como lenguaje universal, espontáneo y con menores restricciones.

El teléfono inteligente o Smartphone en la actualidad incorpora tantas aplicaciones que el usuario desconoce, en muchos casos, que aquellas existen y, cuál es su uso. En todas ellas, se aplica la ventaja del toque (touch) sobre una superficie táctil o de contacto. En este sentido, un reciente estudio realizado por Hooper [68] y llevado a cabo sobre 1.333 observaciones, encontró que el 22% de usuario usan su teléfono móvil solo para hacer llamadas, el 18% solo para recibir llamadas y,

el 60 % se dedica a interactuar con el Smartphone para hacer uso de aplicaciones de mensajería instantánea. También en la misma investigación se pudo determinar que, de éste último porcentaje, el 49 % manejaba el Smartphone con una sola mano, el 36 % usaba la una mano como soporte y la otra para hacer touch, y el 15 % restante, usaba las dos manos en la interacción. A la vista de estos porcentajes, es más que notorio, entonces que el uso de las manos es esencial para el manejo del Smartphone.

Sin embargo, a pesar de que el uso del touch es una gran alternativa en la interacción, también ha dificultado en cierta manera la forma de comunicación, debido en muchos casos, a sus reducidas pantallas (con secciones muy pequeñas de toque) lo que hace difícil o imposible su uso en personas adultas mayores, con problemas de salud (visión o movilidad) o impedimentos físicos, ya sean temporales o permanentes. Lo contrario sucede, cuando las dimensiones de los dispositivos exceden las condiciones anatómicas de uso, por ejemplo en Smartphones con pantallas grandes. Imposibles de sostener con una mano y difíciles de llevar junto al usuario. A esto debemos agregar el hecho, muy común hoy en día, en el cual el usuario adopta posiciones físicas anatómicas poco usuales, con el afán de no seguir comunicado. Tal es el caso de la posición acostado, muy común en caso de enfermedad o también por simple comodidad. Esta forma de interacción, por Touchscreen tan ampliamente usada, merece otra alternativa de comunicación con el usuario pero sin descartarla. Es decir, lograr que la una sea complementaria a la otra, para que el usuario sea quien tome la alternativa más apropiada de interacción, dependiendo de su estado físico o de salud, sería lo ideal.

Este capítulo analiza las ventajas potenciales que tiene la interfaz natural de usuario, sin contacto físico alguno, basada en el gesto (Gesture Based-user Interface, GBUI) frente a una Interfaz por contacto o toque (TouchScreen User Interface, TUI), aplicada al manejo de un dispositivo móvil. Este estudio ha sido realizado además, teniendo en consideración diferentes posiciones anatómicas a saber: de pie o llamada también “normal”, acostado o en posición “especial”. Dichas posiciones anatómicas son comunes en usuarios de Smartphone, pero poco se ha dicho al respecto, sobre si aquellas dificultan el manejo del dispositivo y/o afectan a la exactitud de toque en la TUI de un dispositivo móvil. Posteriormente, y teniendo en cuenta los resultados de nuestro estudio, en el cual participaron veinticinco usuarios (voluntarios), se presenta una propuesta de interacción que soporta GBUI para realizar llamadas telefónicas desde teléfonos inteligentes. Este caso de estudio, ha sido elegido por ser el Smartphone aquel dispositivo de comunicación más

ampliamente usado y universalmente aceptado para esta tarea. La propuesta incluye, la presentación de una arquitectura específica para este tipo de interacción junto a una solución software implementada y basada en ella. La solución software propuesta, es independiente de la posición anatómica del usuario, permite gestionar llamadas telefónicas, funciona combinando una agenda telefónica incorporada (propia) y aquella que posea el dispositivo. Además, goza de ayuda por voz (mediante etiquetas de audio) y permite personalizar los gestos implementados en su biblioteca, para lograr las tareas gestionadas.

4.1. El gesto y su ámbito

Para lograr definir la importancia de la interacción gestual en dispositivos móviles y el efecto que causa el uso del gesto en la Interacción Humano-Ordenador (Human Computer Interaction, HCI), hemos revisado la bibliografía tratando de relacionarla con los siguientes aspectos: El gesto orientado a la interacción, el reconocimiento gestual, la aplicación del gesto y la accesibilidad mediante el gesto. Este enfoque resume en algo aquello que hemos dicho anteriormente y trata de situarnos en un ámbito ideal que acredite el uso del gesto y su alcance.

4.1.1. El gesto en la interacción

La interacción multimodal humano-computador (Multimodal Human Computer Interaction, MMHCI) plantea nuevos paradigmas y establece un modelo de comunicación, donde el contexto de la interacción tiene un impacto significativo en aspectos tales como la información que se transfiere, cuando se hizo la transferencia y cómo se la realizó [183]. Este concepto requiere entender tres aspectos: *el sistema* (la tecnología informática y su usabilidad), *el usuario* que interactúa con él, y la forma de *interacción* entre el usuario y el sistema [76]. En este contexto, MMHCI busca combinar varias técnicas para construir un tipo diferente de interfaz natural, donde el usuario interactúa a través de sus sentidos [76]. Sin embargo, aunque el ser humano tiene cinco sentidos, hasta el momento ninguna forma de interacción logra combinarlos todos en su conjunto. Afortunadamente, el gesto ofrece una gran ventaja en la interacción, ya que combina *visión* (imagen del gesto), *toque* y *conocimiento* (significado) [127]. La aplicación del gesto en la comunicación con la máquina no es nueva, y muchos trabajos de investigación han sido orientados para lograr la interacción usando el lenguaje corporal [48, 43, 130]. De hecho, el gesto en el ser humano puede considerarse una forma de comunicación [120]. En este sen-

tido, el gesto es una característica propia de nuestra especie que mejor se adapta para lograr la ansiada interacción natural con la máquina [150].

Los gestos de las manos, en relación con otros generados por el cuerpo humano, pueden ser tan simples que pasan desapercibidos, o tan complicados que incluso causan dolor al ejecutarlos [151]. El gesto puede realizarse en una superficie (por contacto) o en el aire (sin contacto). Además, dependiendo de si el gesto implica movimiento o no, se puede clasificar como dinámico o estático, respectivamente [120]. La capacidad de expresión del gesto manual es tal, que varía desde simples bibliotecas (uno o varios gestos) hasta lenguajes gestuales completos, como los denominados lenguaje de signos. El lenguaje de señas (Signals Language, LS) más utilizado es el Lenguaje de Signos Americano (America Signs Language, ASL) y sus variantes para diferentes idiomas y regiones.

4.1.2. El reconocimiento gestual

Se han utilizado varios métodos para capturar e interpretar el gesto, algunos de ellos combinando hardware y software, utilizando guantes, imágenes, superficies, dispositivos basados sensores de múltiples y variados tipos [142, 48, 87]. Estos métodos han sido impulsados, en gran medida, por el desarrollo significativo experimentado en tecnologías de procesamiento de señales y modelos matemáticos, admitiendo en éstas técnicas el refinamiento logrado en Inteligencia Artificial (Artificial Intelligence) [148]. Todo esto ha permitido identificar y clasificar el gesto de una manera más rápida y eficaz [128, 2, 97, 27, 143]. Dentro de este ámbito, el reconocimiento gestual basado en señales electromiográficas (EMG) ha obtenido muy buenos resultados [191, 27, 143]. Las señales EMG presentes en la actividad muscular (brazo y antebrazo) permiten el reconocimiento del gesto de la mano. Las señales EMG son detectadas por electrodos (superficiales o intramuscular) [64], cuyo rango de medida varía desde microvoltios hasta los milivoltios. Estos rangos dependen directamente de la intensidad de la señal, además de la ubicación y el tipo de electrodo [3, 71]. Los resultados en la identificación de este tipo de señal, utilizando electrodos sEMG, han demostrado ser altamente eficaces [138]. Por ejemplo, al utilizar una matriz lineal de electrodos [191], un conjunto inalámbrico sEMG [27] o un conjunto sEMG colocado alrededor del brazo [101]. En este sentido, Al-Timemy et al. [3] logra una identificación correcta del movimiento usando sEMG en personas con miembros amputados. En la actualidad, varios wearables (pulseras, relojes, bandas deportivas, etc.) incorporan sensores y electrodos EMG, además de unidades de medición inercial (IMU), que capturan las bioseñales y envía sus datos

por Bluetooth o por Wi-Fi [25, 143].

4.1.3. La aplicación del gesto

Junto con el avance tecnológico y en sintonía con la búsqueda de interfaces nuevas y mejores, el gesto se ha aplicado en múltiples contextos, desde la ejecución de ciertas acciones (por ejemplo, movimientos en juegos) hasta soluciones completas de gestión y control de dispositivos (por ejemplo administración de dispositivos de casa) [142, 145]. Los ejemplos incluyen dispositivos de control [142, 116, 60], en la ayuda del diagnóstico médico [21, 57], en el diseño y aplicación en prótesis para reemplazo de extremidades [3, 148, 149], entre otros.

Varias investigaciones presentan el uso del toque en superficies de contacto para interactuar con dispositivos Smartphones. Estos trabajos utilizan los recursos computacionales que ofrecen los Smartphones, explotan su potencia de cálculo y su rica funcionalidad, aplicándolas a tareas muy variadas. Entre todas estas funcionalidades (agenda, notas, fotos, vídeos, etc.) aquella dedicada a la comunicación (llamadas, mensajes, etc.) es quizás la función más relevante como lo determinó Jung et al. [81]. También se ha utilizado el teléfono inteligente en otras áreas, como en el control remoto de aparatos domésticos [92], punto de venta en aplicaciones de comercio electrónico [18], dispositivo de supervisión en plataformas U-health [118], asistentes en navegación y ubicación geográfica [28, 40], guía de turismo [186], etc. Todas estas soluciones, interactúan con el usuario a través del gesto por contacto (“touch”) [81], usando uno o más dedos, sobre una superficie o touchscreen (resistiva, capacitiva, etc.) en un área específica [78, 133]. El *touch* puede ser realizado en forma estática o dinámica. Por ejemplo, en la mencionada guía turística implementada por Wang et al. [186] usan el *touch* dinámico para operaciones de acercamiento-alejamiento (gesto *zoom*) de áreas en mapas de visita. Kuhnel et al. [92] presentaron una forma de control de dispositivos domésticos en base a movimientos del Smartphone combinándolo con el gesto *touch* estático y dinámico.

Con respecto al uso de una o dos manos para realizar el gesto, Lee et al. [99] determinaron que, en la mayoría de los casos el usuario utiliza sólo una mano. Esto provoca fatiga, a veces dolor, y aumenta el estrés. Si bien la pantalla táctil permite al usuario interactuar directamente con la información que se muestra, no es menos cierto que requiere precisión en el gesto *touch* (estático y dinámico) [74, 133].

4.1.4. La accesibilidad gestual

El grado de comodidad y exactitud del usuario tiene un papel importante en el diseño de interfaces gestuales [150], y su aplicación ha sido notable tanto en el hogar [142, 143], como en aplicaciones industriales [128]. Sin embargo, la afectación o pérdida de agudeza en los sentidos (vista) puede hacer que un simple gesto, como el toque, sea difícil de realizar con exactitud [99, 76]. Del mismo modo, las habilidades motoras en los seres humanos pueden verse afectados, ya sea por causas accidentales o como resultado de un problema degenerativo (enfermedad) en el sistema nervioso central [40, 71]. Del mismo modo, pérdidas de exactitud en el movimiento y la coordinación significativas, se produce debido a la edad de una persona o se relacionan con su posición anatómica en algún momento [99, 76]. Todos estos aspectos comprometen, temporal o permanentemente, la motricidad fina de las manos en una persona [152, 53, 132], causando errores en el manejo de dispositivos que requieren algún grado de exactitud.

Para ayudar a superar estos problemas, se han propuesto muchas soluciones de accesibilidad (por ejemplo, reconocimiento de voz, manos libres, reconocimiento de imágenes, etc.), algunas de ellas incorporadas en dispositivos y Sistemas Operativos (SO). Un SO que ha incorporado soluciones de accesibilidad (por ejemplo, el zoom del teclado para facilitar el toque) en Smartphone, es Android. Sin embargo, estas soluciones de accesibilidad no han tenido un impacto significativo en el grado de aceptación de los usuarios, especialmente en los adultos mayores [205, 133]. Por lo tanto, la incorporación de una opción de interacción gestual basada en el gesto de la mano para la gestión de teléfonos inteligentes sería una buena alternativa. De hecho, aquellas personas que, en un momento dado (enfermedad o accidente), padecen problemas de movilidad, pérdida de visión, sordera, etc., tendrían mayores oportunidades de comunicación y uso de un teléfono inteligente. Para las personas sanas, la interacción gestual con la mano puede ser útil para resolver ciertas circunstancias (ruptura de la pantalla, manos ocupadas, posición especial, etc.), lo cual es muy importante pues el teléfono inteligente se ha convertido en un medio de comunicación universal.

4.1.5. Revisión científica

La Tabla 4.1 resume todos los trabajos de investigación analizados en el contexto de aplicación que hemos expuesto anteriormente. Como puede observarse, y como ya se dijo en capítulos anteriores, han sido varios los trabajos de investigación que usan la cámara o llamados dispositivos ópticos para lograr el reconocimiento

gestual. Sin embargo, estas soluciones deben manejar problemas de oclusión, iluminación y enfoque. Ahora, hacemos un nuevo planteamiento, analizando aquellas propuestas que utilizan gestos de toque (*touch*) a través de una pantalla táctil, obligando al usuario a tener exactitud en el *touch* y un buen reconocimiento o capacidad visual. Revisamos además, aquellos que utilizan electrodos sEMG y ofrecen una mejor garantía de reconocimiento gestual a costa de ser, un poco o moderadamente, intrusivos. Por lo tanto, las aplicaciones de teléfonos inteligentes basadas en electrodos sEMG, que a su vez implementen comunicación inalámbrica para captar dichas señales, podrían convertirse en un sistema integral de soporte para el usuario. Las soluciones de este tipo deberían integrarse en el sistema operativo proporcionado por el fabricante, para lograr superar en gran medida las limitaciones del uso de la pantalla táctil que se presentan día a día en los teléfonos inteligentes.

El uso de Touchscreen se han vuelto muy común en los dispositivos móviles en especial en los Smartphone. Los trabajos que se fundamentan en ella, suponen que el usuario esta en uso de todo su potencial sensitivo, motoro y psicológico, haciendo enfoques novedoso de aplicación. Además, son especialmente atractivas, porque ahorran espacio en el dispositivo y permiten el acceso a sus funcionalidades combinando imagen, animación y color, en áreas de entrada de datos, logrando alcanzar una forma de interacción y control del dispositivo, muy dinámica a través de botones o interruptores (switches). Esto provee altos niveles de inmediatez en la interacción al tocar, presionar o recorrer (trazar) un área con los dedos. Sin embargo, durante la revisión realizada, no se ha encontrado ningún trabajo previo que proponga aplicaciones para teléfonos inteligentes que usen señales EMG para el reconocimiento gestual. Además, no se han encontrado estudios previos que analicen el índice de dificultad y precisión, del gesto con y sin contacto, en este contexto. Sin embargo, esto es esencial para garantizar la utilidad del gesto. En consecuencia, este capítulo intenta abordar ambas deficiencias detectadas.

En conclusión, sería de gran ayuda una solución basada en gestos sin contacto, utilizando una biblioteca gestual personalizada y adaptada al dispositivo móvil, que permita la gestión de aplicaciones o servicios en un teléfono inteligente.

4.2. TUI, evolución y características

La Interfaz Gráfica de Usuario (Graphical User Interface, GUI) es uno de los principales propulsores de la conquista informática a nivel mundial. Todos los aspectos de la vida humana involucran el uso, manejo o acceso a una aplicación apli-

Tabla 4.1: Revisión de estudios que usan el gesto

Primer autor	Interacción	Contacto	Dispositivo de reconocimiento	Varios	Principal contribución
Sanna [162]	Corporal	No	Cámara, Smartphone	Estáticos	Control de Drón
Francese [54]	Corporal	Si/No	Varios	Dinámicos	Control de juegos
Yang [198]	Facial	Si	EMG	Dinámicos	Teclado virtual
Hasan [65]	Manual	No	Cámara	Estáticos	Reconocimiento usando redes neuronales
Kiliboz [87]	Manual	Si/No	Polhemus Patriot	Dinámicos	Técnica de detección y reconocimiento
Heo [66]	Facial, Manual	Si/No	Cámara, Guante de datos, EMG	Ambos	Control de juegos
Pomboza [143]	Manual	No	EMG	Estáticos	Control de dispositivos de casa
Erdem [48]	Manual	No	Cámara, PIR	Dinámicos	Control de dispositivos
Dinh [43]	Manual	No	Cámara	Ambos	Control de dispositivos
Olm-Bar [130]	Manual	Si/No	Cámara	Ambos	Control de dispositivos autónomos
Chen [25]	Corporal, Manual	No	Cámara, acelerómetro	Ambos	Framework fusionado de reconocimiento
Kittinel [92]	Manual	Si	Smartphone, IMU	Dinámicos	Control de dispositivos con movimientos de las manos
Premaratne [145]	Manual	No	Cámara	Dinámicos	Control de dispositivos
Blazquez Cano [18]	Manual	Si	Smartphone	Ambos	Tecnología Touchscreen
De Waard [40]	Visual/Auditiva	No	Smartphone, GPS cámara	Ninguno	Mapa electrónico de navegación
Este trabajo	Manos/auditiva	Yes	Smartphone, sEMG	Estáticos	Análisis de TUI vs. GBUI, software SICLLE

Ds: Dispositivo usado, MC: Método de clasificación, LG: Lenguaje gestual, ACC: Exactitud y porcentaje.

cación informática que mediante el uso de su GUI logra el acceso a información, procesamiento de datos o resultados esperados en diferentes fases de la actividad humana. Para el desarrollo de interfaces gráficas ha sido necesaria la aplicación de leyes que permitan su diseño y mejora. Una de esas leyes es la llamada Ley de Fitts, la cual ha sido adaptada, ampliada y mejorada, a diferentes ámbitos orientados a renovar el diseño y rendimiento de las GUI.

El uso de GUI requiere de un dispositivo apuntador (generalmente mouse) para lograr el acceso a una determinada acción, opción o menú en un programa o software que la implemente. La GUI ha sido integrada a las llamadas pantallas táctiles (Touchscreen) para lograr una mejor interacción y eliminar el uso de dispositivos apuntadores ajenos al usuario y admitiendo el uso de los dedos del usuario como punteros para registrar el toque sobre dichas superficies, naciendo de esta manera las llamadas Interfaces de Usuario por Touchscreen (TouchScreen User Interface, TUI). A todo ello, se suma el hecho de poner a disposición del usuario teclados en pantalla (On-Screen Keyboards) como una forma de asistencia al usuario [14]. Las Touchscreen, de acuerdo a la tecnología que implementan [139, 158, 14], puede ser resistivas, capacitivas, por superficie de onda y óptica-infrarroja.

- **Resistivas**, son más baratas y mucho más resistentes a agentes externos como agua, polvo e inclusive agua salada. Pueden ser usadas con un puntero o con el dedo. Inicialmente escasas de brillo, en la actualidad han sido mejoradas en dimensiones, peso y brillo. Muchas de ellas lograr exponer una imagen apreciable a pesar de estar expuestas a la luz directa del sol. Su bajo brillo (15 %) y su grosor ha provocado que sean reemplazadas por otras, en los dispositivos móviles.
- **Capacitivas**, son una mejora excepcional con relación a las anteriores, pues permiten el uso de más de un dedo a la vez en la superficie (multitouch). Su funcionamiento se basa en sensores capacitivos que combinados a la característica de conductancia del cuerpo humano, logran una distorsión del campo electrostático por el cambio de capacitancia (capacidad eléctrica). La ubicación de la posición de toque en la pantalla, puede ser realizada de múltiples maneras para luego ser enviada al controlador para su procesamiento. Su calidad de imagen es mucho mejor y tienen una mayor respuesta, sin embargo son más caras y no pueden usarse con un puntero normal, sino uno especial para pantallas capacitivas.
- **Superficie de onda (Surface Wave Acoustic, SAW)**, son aquellas que

usan ondas ultrasónicas que pasan por el panel de la pantalla táctil. El cambio en las ondas al producirse el toque (absorción de onda) es la que registra la posición de toque. Esta información es enviada al controlador para su procesamiento. Es una tecnología más reciente sujeta aún a problemas y elementos externos, pero ha dado buenos resultados al mejorar la calidad en imágenes y al obtener una excelente durabilidad, usándola especialmente en cajeros automático, controladores de tráfico, aplicaciones militares, etc.

- **Optica-infrarroja**, se basa en arreglos de led infrarrojos y detectores de luz que forman una malla (cuadrícula) a través de la pantalla. Cada toque provoca una caída de voltaje que es detectada por la ausencia de luz y el controlador calcula las coordenadas de toque. Es muy robusta y recomendada para ambientes hostiles por su calidad óptica y porque, puede ser escalable. El principal problema es la alineación de los detectores/emisores.

De los diferentes tipos de Touchscreen descritos, las más usadas en dispositivos móviles son las de tipo capacitivo, por su bajo costo, alta gama de color, aunque la durabilidad sigue siendo un inconveniente. Pero aún resta aprovechar al máximo todas las ventajas que las touchscreen multi-touch permiten, además de lograr mejores aplicaciones que exploten las ventajas que pueden dar al usuario. Sin embargo, la capacidad de trabajar con ambas manos en la interacción es la manera más común de uso [14].

4.2.1. Ley de Fitts

El modelo de ley de Fitts, que fue publicado originalmente en 1954, se utiliza para estimar el tiempo para completar una tarea de selección o un movimiento. El modelo de ley de Fitts ha demostrado ser uno de los modelos más robustos y exitosos para modelar una tarea motora humana y está definido por:

$$T = a + b \left(\frac{A}{W} + 1 \right) \quad (4.1)$$

donde T es el tiempo medio que toma el completar un movimiento. A es la distancia desde el punto inicio hacia el centro del punto objetivo. W es el ancho del objetivo. a y b son coeficientes de pendiente y de intersección, típicamente derivados, usando regresión lineal sobre los datos obtenidos del test experimental.

La ley de Fitts ha tenido varias revisiones que han sido aplicadas y verificadas en HCI, mejorando la predicción bajo condiciones particulares, las cuales incluyen

diferentes dispositivos en tareas de apuntar a un objetivo, modalidades de interacción, contextos de uso y configuraciones experimentales [109]. A partir de ella, basándose en la expresión logarítmica de Shannon se determinó el *índice de dificultad* (ID) para una tarea motora es el doble de la distancia y la anchura del objetivo final [52], y viene dada por:

$$ID = \text{Log}_2 \left(\frac{2A}{W} \right) \quad (4.2)$$

Debido a que A y W son ambas medidas de distancia, la relación en el logaritmo se da sin unidades. Debe usarse la Base 10 para el caso de trabajar con dígitos, pues inicialmente estaba pensado para trabajar con bits (base 2).

4.2.2. Características de TUI

El tamaño de la pantalla en un dispositivo móvil (Smartphone, Smartwatch, tablets, etc) es muy importante, pues en dichos dispositivos la pantalla táctil es su principal punto de interacción con el usuario [105]. Varios estudios han sido realizados tratando de encontrar una relación entre la dimensión de la pantalla, el tamaño de los botones y la exactitud de toque por parte del usuario. Así tenemos, el trabajo realizado por Hara et al. [63] sobre dispositivos Smartwatch en el cual, logra determinar que, el tamaño que tiene un botón (objetivo) en la pantalla tiene un efecto importante sobre el rendimiento de la acción de apuntar o tocar con el dedo a dicho botón. Este estudio demostró que, tanto el tiempo para completar tareas como la tasa de error, fueron altas para botones de tamaño pequeño. Similares resultados fueron encontrados por Trudeau et al. [181] en un estudio sobre el rendimiento del tareas de apuntar usando el dedo de una sola mano, especialmente sobre dispositivos pequeños como Smartphones. Además, se determinó que el tamaño de los dedos del usuario ocuyen a otros botones cuando se trata de tocar a botones pequeños y que la exactitud de tales movimientos es limitada (debido al tamaño de los dedos del usuario). Para intentar lograr una estimación del tiempo y tamaño óptimo en una tarea de selección de un botón o también llamado “objetivo”, se ha usado la llamada Ley de Fitts [52].

En este sentido, Jin y Kiff [78] usando la ley de Fitts en su forma original (ecuación 4.1) logran determinar el tamaño y separación óptimo entre objetivos (botones) más recomendable para personas adultas mayores, concluyendo que, si se va a trabajar con adultos mayores con relativa normalidad de destreza manual el tamaño del objetivo debía ser de 16,51 mm y un espacio de entre 3,17mm a

6,35mm. Sin embargo, para adultos mayores con destrezas manuales pobres, el tamaño del objetivo debía estar en 19,05mm y la longitud de separación entre 6,35mm y 12,7mm. Se pudo determinar además, que el **tiempo de reacción** (Reaction Time, RT) en una persona, disminuía con relación al tamaño del objetivo, siendo mejor en aquellos casos en los cuales el objetivo era más fácilmente visible. En este mismo estudio se pudo determinar que, si estamos diseñando botones u objetivos separados entre ellos y admitimos un tiempo de reacción de alrededor de 1,400ms es aceptable, entonces el tamaño del botón u objetivo podría estar en 11,43mm. Se menciona también que, si la interfaz usa botones u objetivos separados en una misma superficie, una separación entre 3,17mm y 12,7mm era la más adecuada, aunque las personas adultas prefieren un espacio de 6,35mm. Hay que considerar además, que mayores espacios redundarían en tiempos de búsqueda del objetivo en la pantalla. Un espacio equivalente a cero (0) no afecta a la velocidad pero si a la exactitud.

4.2.3. Android en los Smartphone

Esta subsección trata de realizar un rápido y efectivo enfoque sobre Android, sus componentes y entorno de desarrollo. No pretendemos adentrarnos demasiado en el estudio de Android, por alejarse de nuestro objetivo, pero es necesario revisar algunos conceptos, bajo los cuales se diseñaron e implementaron las aplicaciones que se usan en este análisis.

Android es un entorno de ejecución de aplicaciones para dispositivos móviles que incluye un sistema operativo, un frame de aplicaciones (Application Framework) y aplicaciones principales (core applications). Las aplicaciones se escriben en lenguaje Java y C++, basándose en las API proporcionadas por el Kit de desarrollo de software para Android (Software Development Kit, SDK). La base de la pila de software de Android es el Kernel de Linux. Android utiliza Linux para sus controladores de dispositivos, gestión de memoria, gestión de procesos y redes. Por lo general, un desarrollador de aplicaciones no debería programar directamente en esta capa. El nivel siguiente contiene las bibliotecas nativas de Android. Estas bibliotecas están escritas en C y C++ y son utilizadas por varios componentes del sistema de las capas superiores. La incorporación de estas bibliotecas en las aplicaciones de Android, se logra a través de interfaces Java. Esta capa contiene una biblioteca C personalizada, un motor de base de datos SQL, bibliotecas gráficas 2D y 3D, un motor de navegador web nativo (WebKit) y codecs multimedia (por ejemplo, MPEG4 y MP3). El siguiente es el *Android Runtime*, que consta de la

Máquina Virtual Dalvik y las bibliotecas principales. Dalvik ejecuta archivos .dex (Dalvik Executable) que son más compactos y eficientes en manejo de memoria que los archivos de clase Java. Esta es una consideración importante, para los dispositivos basados en batería o pilas con memoria limitada. Las librerías de núcleo están escritas en Java y proporcionan un subconjunto sustancial de los paquetes de Java 5 SE (por ejemplo, librerías estándar de E/S, redes, utilidades, etc.) así como librerías específicas de Android, las cuales son necesarias para acceder a las capacidades que ofrece el hardware, el sistema operativo y las bibliotecas nativas.

La capa de *Application Framework*, escrita completamente en Java, incluye herramientas proporcionadas por Google, así como extensiones o servicios propietarios. Un componente importante del framework es el *Activity Manager*, que gestiona el ciclo de vida de las aplicaciones. La capa superior es la capa de *Aplicaciones* necesaria para implementar aplicaciones tales como un teléfono, un navegador web, un cliente de correo electrónico y otras. Cada aplicación en Android es empaquetada en un archivo .apk (*Paquete Android*) para su instalación. El .apk es similar a un archivo Java jar estándar, el cual contiene todos los recursos (por ejemplo, imágenes) para la aplicación. El paquete de Android es una colección de componentes. Los componentes en un apk, están aislados de uno a otro apk y, sólo pueden comunicarse entre sí y compartir datos a través de los medios proporcionados por el sistema. Cada apk, está asociado con un proceso primario en el que se ejecutan todos los componentes de la aplicación (tales como: Actividades, Servicios, Receptores Broadcast y Proveedores de contenido). Los componentes de esta aplicación, junto con las capacidades, los permisos y los requisitos deben aparecer en el archivo *AndroidManifest.xml*. Un detalle muy particular es que el fabricante siempre pone nombres de dulces a cada una de sus versiones. Mayores detalles de Android pueden encontrarse en ¹.

Android es el OS que más se ha difundido, llegando a ocupar casi el 89 % del mercado en España [159]. Por ello, hemos escogido a Android como aquel sistema operativo sobre el cual desarrollaremos nuestro estudio.

Nuestro principal interés se ha enfocado a aquellas opciones encargadas de la gestión de llamadas en diferentes teléfonos inteligentes, las cuales manifiestan una estructura similar incluyendo áreas comunes para múltiples fabricantes, a pesar que cada Smartphone posee una capa de personalización específica o una versión particular de Android (Lollipop y Marshmallow). Por otro lado, se determinó que el usuario sigue una secuencia de navegación equivalente, en la interacción con las

¹<https://www.android.com/>

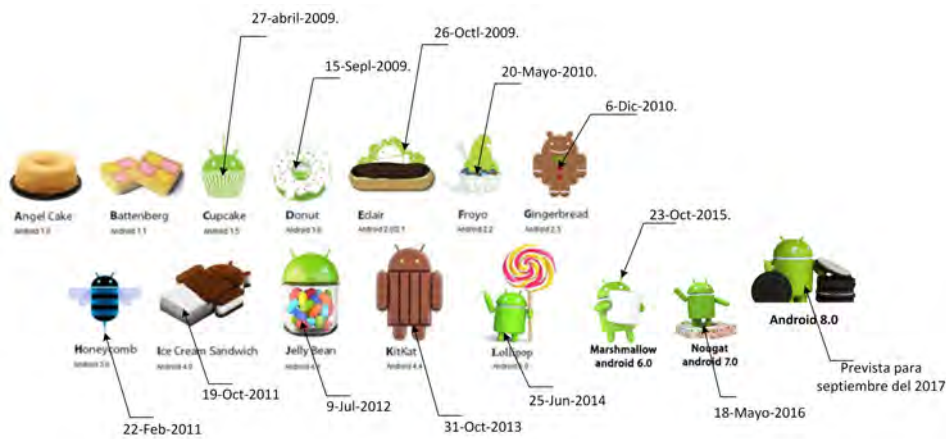


Figura 4.1: Diagrama evolutivo de Android.

aplicaciones. No obstante, no se activaron opciones personalizadas ni de accesibilidad para realizar este estudio.

4.3. Características del Smartphone y su interfaz

Los teléfonos inteligentes, como es obvio, cubren un gran segmento del mercado móvil orientado a realizar “llamadas telefónicas”. La interacción con el usuario, en la última generación de teléfonos inteligentes, se gestiona mediante el uso de una Interfaz de Usuario por pantalla táctil (TouchScreen User Interface, TUI). La pantalla está construida por una pantalla multi-touch, la cual permite la presión de uno o más dedos. La interacción es percibida cuando un dedo presiona un área específica de la pantalla. En general, esta área está representada gráficamente por un widget o un botón, para facilitar su identificación por el usuario y la consecuente acción que la aplicación debe realizar a continuación. Para el diseño de los experimentos en esta tesis, se realizó un análisis cuyo objetivo fue determinar las características promedio, que el teléfono inteligente debe cumplir y además, un estudio de la interfaz que cada fabricante incorpora en el Smartphone para hacer llamadas telefónicas. En ambos casos, se analizaron doce Smartphones de diferentes fabricantes (Samsung Galaxy S6, Sony Xperia Z5, Motorola Moto E, Bq Acuario A4.5, Sony Xperia M2 y otros), agrupados en tres categorías (alta, media y baja).

Luego de realizado nuestro análisis, fue posible determinar que un dispositivo de gama media con un tamaño de pantalla entre 4 y 5 pulgadas, con un peso entre 130 y 155 g, y una resolución de 960×540 píxeles, sería más apropiado para llevar a cabo los experimentos. En consecuencia, hemos optado por usar un Smartphone

Tabla 4.2: Análisis de dispositivos móviles

Gama	Fabricante	P-DM	TP	R-CF
Alta	Samsung Galaxy S6	138	5,1	5
	OTC One 9	157	5	4
	Apple iPhone 6s	143	4,7	5
	Sony Xperia Z5	156,5	5,2	5
Media	Bq Acuaris M5	144	5	5
	Motorola Moto G	155	5	5
	Sony Xperia M4	136	5	5
	Alcatel One Touch	141	5,5	8
Baja	Motorola Moto E	145	4,5	2
	Bq Acuaris A4	115	4,6	5
	MyWigo Magnum 2	112,5	5	5
	Lumia 640	145	5	0,9
Promedios		140,666	4,966	4,575

P-DM: Peso del dispositivo móvil en gramos, TP: Tamaño de pantalla dado en pulgadas, R-CF: Resolución de cámara frontal.

Sony Xperia M2 con un tamaño de pantalla de 4,8 pulgadas, con un peso 148 g, una resolución de 960×540 píxeles y dimensiones (largo, ancho, profundidad) de $139,9 \times 71,1 \times 9,6mm$ para nuestras pruebas.

4.3.1. Zonas de importancia

Hemos encontrado determinadas zonas comunes entre los frameworks analizados, a las cuales hemos llamado “zonas de importancia”, justamente porque son zonas en las cuales el usuario realiza la gestión de llamada. Dichas zonas son usadas para ubicar una llamada recibida, para aceptar o rechazar una llamada entrante, hacer una llamada (accediendo directamente a los contactos) y por último, una zona de aceptación de la llamada entrante. Sea cual sea el fabricante, estas zonas siempre han estado presentes.

Las *zonas de importancia* antes indicadas, han sido definidas en el dispositivo móvil que servirá para las pruebas, cuyas características físicas fueron dadas anteriormente, en la Sección 4.2, y que se describen en la Figura 4.2.

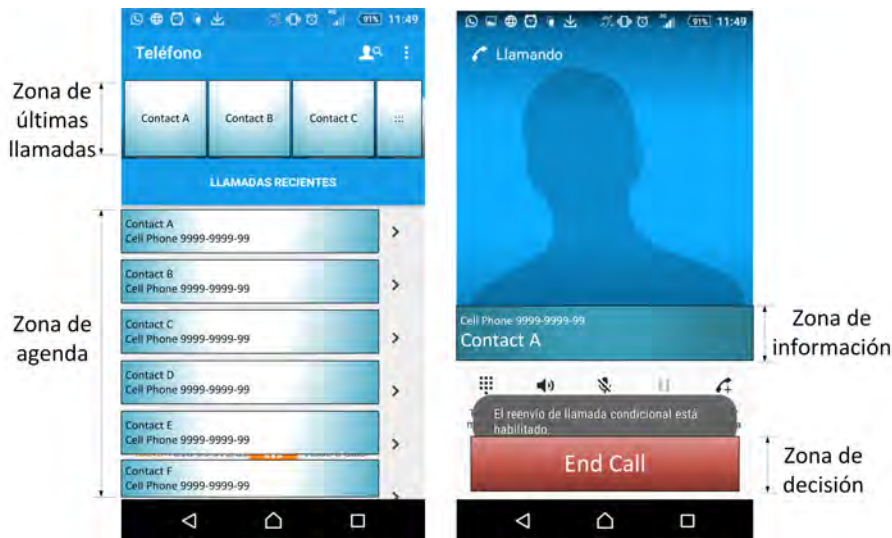


Figura 4.2: *Zonas de importancia* en un Smartphone con SO Android.

Zona de llamadas recientes

La zona de llamadas recientes o últimas llamadas pretende ayudar al usuario al realizar una llamada desde un listado de las llamadas, generado por aquellas llamadas recibidas (aceptadas o no) por el usuario del dispositivo móvil. Las llamadas se ordenan por fecha y hora de recepción, ubicando la más reciente al inicio y permitiendo navegar en el listado en caso necesario. Una llamada desde un número telefónico registrado a nombre de un contacto existente, será mostrado por el nombre del contacto y demás características asignadas, mientras que un número no registrado, será mostrado simplemente.

Zona de contactos

Esta zona permite al usuario navegar directamente entre los contactos, desde la aplicación de control de llamadas, usando la agenda de contactos del dispositivo móvil. Es decir, el acceso a los contactos registrados en la agenda del dispositivo es posible porque la aplicación de llamadas tiene acceso o permiso a la misma. Los permisos en aplicaciones principales para Android están dados por defecto, en cambio en caso de un apk, es el usuario quien los concede admitiendo los riesgos que ello implica [94]. Realizar una llamada directamente desde esta lista es posible y puede ser realizada en forma directa, independientemente del hecho que un mismo contacto tenga varios números registrados, la llamada se realizará a aquel número del cual provino la llamada entrante.

Zona de información

Es aquella zona que se encarga de informar al usuario si una llamada, entrante o en curso, corresponde al contacto seleccionado. Básicamente muestra información del contacto al cual se está llamando o del cual proviene la llamada. Por lo general es textual a menos que haya configurado información adicional en la aplicación de contactos del dispositivo tal como foto del contacto u otros.

Zona de decisión

En esta zona podemos aceptar o finalizar una llamada telefónica. Para aceptar una llamada telefónica, es necesario tocar la zona indicada en la GUI, y realizar el gesto de toque indicado. El gesto de aceptación de llamada, en muchos casos es personalizado por el fabricante del dispositivo móvil pero por lo general consiste en: tocar con el dedo un área determinada de la GUI o mantener presionada dicha área mientras se desliza, de izquierda a derecha o de arriba hacia abajo, sobre la superficie de contacto. Para finalizar una llamada telefónica, es necesario realizar un toque en esta área.

Para una interfaz TUI, la forma natural de interactuar con una *Zona de Importancia* es tocar consecutivamente estas zonas en un orden determinado el cual sigue una secuencia específica. Cuando una interacción por toque es percibida por el sistema, éste reacciona ejecutando una acción o proceso asociado, mientras la interacción se produzca en dicha zona.

Para una interfaz GBUI, la interacción se realiza de una manera diferente porque no hay contacto con el dispositivo. El sistema requiere entonces una interfaz que le permita reconocer el gesto que el usuario ha realizado o está ejecutando. Para lograr capturar el gesto usaremos la pulsera gestual MYO junto a un sistema de reconocimiento del gesto. La pulsera gestual usa sensores sEMG para capturar las señales EMG que genera el usuario al realizar el gesto, permitiendo su identificación y reconocimiento. Los detalles técnicos de la pulsera gestual fueron discutidos a profundidad en la Sección 3.3 del Capítulo 3 y un estudio minucioso se ha expuesto en el Apéndice B. La interacción es percibida por el sistema a través del gesto realizado e identificado usando una biblioteca gestual que incorpora el sistema propuesto, y a partir del cual, el software reacciona ejecutando una acción específica.

4.4. Definición del Test de interacción

El test de interacción ha sido definido de acuerdo a las *Zonas de Importancia* y su secuencia ha sido establecida como se muestra en la Figura 4.3. Consiste en un conjunto de toques en áreas predeterminadas, para el caso de la interfaz TUI, y un conjunto de gestos establecidos, para el caso de la interfaz GBUI. En ambos casos se acudirá a un *punto de interacción*, el cual será tratado de forma diferente en cada interfaz. En una interfaz TUI, el *punto de interacción* tiene lugar cuando el usuario toca un *objetivo* en una *Zona de Importancia* definida. En cambio, en una interfaz GBUI, el *punto de interacción* tiene lugar cuando el sistema detecta un gesto realizado por la mano del usuario en una secuencia preestablecida y relacionada con un *objetivo*.

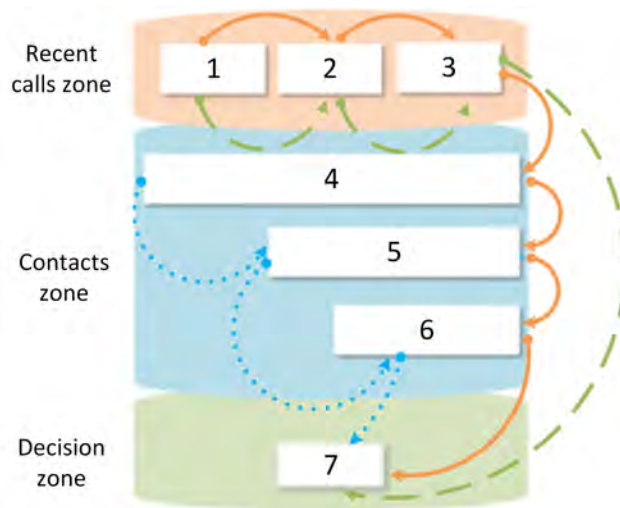


Figura 4.3: Secuencia de interacción a través de las *Zonas de Importancia*.

El test de interacción se compone de tres etapas. Cada etapa comprende una secuencia de *puntos de interacción* que deben alcanzarse siguiendo un orden establecido. El acierto sobre cada *punto de interacción* es la meta u objetivo a ser alcanzada por parte del voluntario. El *punto de interacción* entonces se convierte en un *objetivo* (un ejemplo de ello es el clásico “tiro al blanco”) que puede ser representado gráficamente por nodos o una secuencia de nodos. Entonces, podemos definir los *toque-objetivo* y los *gesto-objetivo* que son los dos tipos de *objetivos* que el usuario debe satisfacer con respecto a cada interfaz de usuario.

La Figura 4.3 muestra un ejemplo de la correlación entre *objetivos* y *Zonas de Importancia* que contiene las *objetivos* con respecto a la interfaz TUI. Así, tenemos:

- Etapa 1, en la que se cumplen todos los objetivos (línea continua marrón);
- Etapa 2, en la que se alcanzan los objetivos 1, 2, 3 y 7 (línea verde discontinua);
- Etapa 3, en la que se alcanza un subconjunto de 4, 5, 6 y 7 objetivos (línea punteada azul).

4.4.1. Definición de los parámetros de evaluación

Los parámetros a evaluar en el test de interacción de las interfaces son el **índice de dificultad, (ID)** que se halla definido en la Sección 4.2.1 mediante la ecuación 4.2 y la **exactitud de la interacción (ACC)**.

Para lograr la métrica de *ACC* o medida de la exactitud en el toque a un objetivo, hemos procedido de forma similar a [142] y tomando en consideración lo descrito en la ecuación 1.2 que permite medir el rendimiento de un modelo por su exactitud. Básicamente es una relación entre los aciertos (casos positivos) y los no aciertos (casos negativos). *ACC* entonces viene dada por:

$$ACC = \left(\frac{Positivos}{Positivos + Negativos} \right) \quad (4.3)$$

donde *ACC* es la exactitud. *Positivos* son todos los intentos realizados por el usuarios que hayan coincidido con la ubicación del objetivo o aquellos gestos que hayan sido reconocidos positivamente por el sistema. *Negativos* son aquellos intentos del usuario que no hayan sido realizados en el área del objetivo o aquellos gestos no reconocidos por el sistema.

El *índice de dificultad* de interfaz sólo es aplicable a la interfaz TUI, mientras que la *exactitud de la interacción (ACC)* es aplicable a ambas interfaces.

4.4.2. Software para test

Para estimar la validez del gesto en interfaces de GBUI en oposición a interfaces del tipo TUI es necesario realizar un test de interacción que sea evaluado sobre los dos tipos de interfaz. Para lograrlo dos aplicaciones fueron diseñadas, una por cada tipo de interfaz (TUI y GBUI). Ambas aplicaciones fueron desarrolladas para dispositivos que ejecuten el SO Android en sus versiones 5.1 y superiores. En cada uno de ellas, el usuario sigue una secuencia de *puntos de interacción* para lograr un objetivo, en nuestro caso relacionado con la gestión de llamadas.

TUI

El *toque* es un gesto definido por el usuario para la interacción táctil que requiere un contacto directo del dedo con la superficie táctil (Touchscreen). Los puntos de interacción se activan cada vez que se realiza un toque (touch) dentro de una *Zona de Importancia* específica, como se muestra en la Figura 4.4.

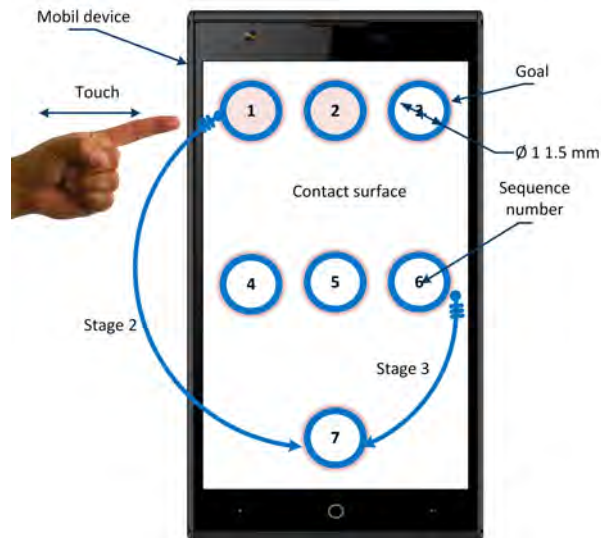


Figura 4.4: Distribución de objetivos (iconos) para el test de interacción TUI.





Además, la distribución de las *Zonas de Importancia* es esencial para lograr un rendimiento óptimo de las interacciones entre el usuario y el dispositivo móvil. Para el diseño de la prueba de interacción, las recomendaciones de Google para el diseño de interfaces gráficas en dispositivos móviles se tuvieron en cuenta. La separación y el tamaño de los iconos (11,5 mm) intentaron ajustarse al estudio realizado por Jin et al. [78] quienes verificaron que tamaños más grandes de iconos no producen una pérdida significativa de tiempo de reacción (RT) del usuario y por lo tanto no afecta su rendimiento en interfaces TUI. Cada objetivo táctil está numerado y puede tener dos estados: activo e inactivo. En el estado activo, el objetivo es visible y está a la espera de registrar una acción. Cuando el estado es inactivo, el objetivo se oculta y no se puede realizar ninguna acción sobre él.

La Figura 4.4 también muestra la distribución de los iconos a través de la pantalla en las pruebas de interacción.

GBUI

En este caso, el usuario puede realizar los gestos sin tocar ninguna superficie del Smartphone. El gesto será reconocido, siempre que el gesto se haya almacenado previamente en el catálogo gestual. La identificación del gesto es el elemento significativo que activa un punto de interacción en el dispositivo móvil. Esto se consigue, en nuestro caso, mediante un proceso de reconocimiento a partir de las señales EMG recogidas por los sensores sEMG de la pulsera gestual MYO (Capítulo 3 Sección 3.3). Por esta razón, cada gesto tienen asociado un *punto de interacción* (y consecuentemente un *gesto-objetivo*) con la acción correspondiente cuando se logra el objetivo. Los gestos predefinidos en este estudio se describen en la Tabla 4.3.

Tabla 4.3: Gestos usados en la interfaz GBUI

Gesto	Nombre	Descripción
	Mano abierta	Todos los dedos extendidos fuera de la palma.
	Palma dentro	Dedos extendidos junto a la palma. La mano, manteniendo la posición y antebrazo firme, hace un giro a la izquierda.
	Palma fuera	Dedos extendidos junto a la palma. La mano, manteniendo la posición y antebrazo firme, hace un giro a la derecha.
	Mano cerrada	Todos los dedos juntos en la palma de la mano.

A diferencia de TUI, los gestos-objetivos pueden tener lugar en cualquier momento, cuando se identifica un gesto válido; pero no se toman en consideración si la *Zona de Importancia* no está activa al igual que sus puntos de interacción. Sin embargo, el sistema debe controlar que los puntos de interacción ocurren siguiendo una secuencia específica para ser considerados válidos, tal como sucede en la interfaz TUI. La secuencia y los gestos usados por cada escenario son mostrados en la Figura 4.5. Para facilitar el seguimiento de la secuencia gesto-objetivo en GBUI, hemos implementado un **Árbol de Control Gestual** (Gestural Control Tree, GCT) cuya idea nació en el trabajo presentado por Pomboza & Holgado [142] y que ha sido implementado también en esta investigación. El GCT organiza la identificación del gesto en niveles para saber cuál es el siguiente gesto válido en

relación al gesto válido anterior, dentro de la secuencia. Es decir, cuando un gesto es reconocido en un nivel específico, el sistema sólo puede identificar un gesto del siguiente nivel. De esta manera, un gesto no puede tener lugar al mismo tiempo porque los niveles de GCT imponen restricciones en la secuencia. El GCT utilizado para la prueba de interacción GBUI se muestra en la Figura 4.5.

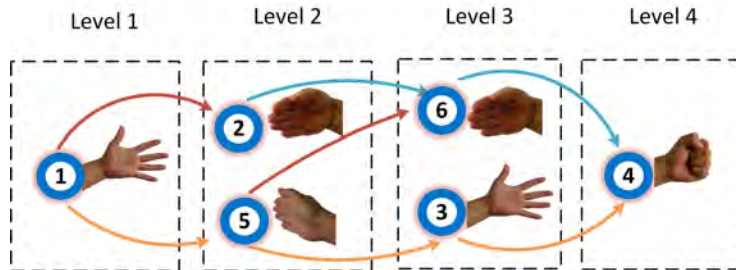


Figura 4.5: Secuencia de gestos-objetivos por cada escenario en interacción GBUI usando GCT.

4.5. Probando TUI vs. GBUI

4.5.1. Participantes

En nuestros experimentos para comparar la interacción TUI con la interacción GBUI participaron veinticinco voluntarios, los cuales utilizaron el software diseñado para este efecto y presentado en la Sección 4.4.2. Todos los voluntarios tenían experiencia en informática y un sólido conocimiento en el manejo de dispositivos móviles. Las edades oscilaron entre 20 y 50 años (seis mujeres y diecinueve varones) ($Media = 33,44, DS = 8,7$). Todos eran propietarios de un teléfono inteligente y tenían experiencia con el uso de "manos libres".

4.5.2. Protocolo experimental

Las posiciones utilizadas en este estudio fueron *normal* y *especial*. En la posición *normal* el usuario estaba de pie, con sus dos pies apoyados en el suelo. En la posición *especial*, el voluntario estaba acostado en un sofá de pruebas, totalmente reclinado, conservando la cabeza sin apoyo (almohada) y manteniendo una vista directa de la pantalla del Smartphone a nivel de sus ojos.

Se realizaron un total de cinco ensayos por participante, para cada posición (*normal* y *especial*). Cada ensayo tenía una duración de 10 segundos controlada por software. Durante este tiempo, el usuario realizaba la prueba de interacción

y el sistema registraba cada intento realizado. Un intento era considerado válido cuando un punto de interacción era alcanzado dentro de una secuencia válida; en todos los demás casos, se anota como inválido.

Para calcular el *ACC* en TUI y GBUI, todas las secuencias de toque definidas debían ser realizadas sin omitir ninguna. En el cálculo se consideraban los aciertos como positivos y los errores como negativos.

4.5.3. Pruebas con TUI

En las pruebas con interfaz TUI los usuarios ejecutaron la prueba de interacción utilizando la aplicación software diseñada e implementada para este tipo de interfaz. El uso de las manos durante los ensayos fue el siguiente: se utilizó una mano para sujetar el dispositivo móvil y la otra mano (mano dominante o mano de gesto) para realizar el toque usando el dedo índice. No era permitido cambiar de mano una vez que se daba inicio al ensayo. El tiempo era controlado en forma automática por la misma aplicación, emitiendo un mensaje y mostrando los totales alcanzados en cada prueba, al mismo tiempo que todos los datos eran guardados en una base de datos para su posterior procesamiento. La prueba daba inicio cuando el usuario tocaba el objetivo 1 del primer escenario y luego continuaba tocando los siguientes objetivos en la secuencia indicada, como se muestra en la Figura 4.6. Para mayor claridad, las secuencias de la prueba de interacción se describen a continuación:

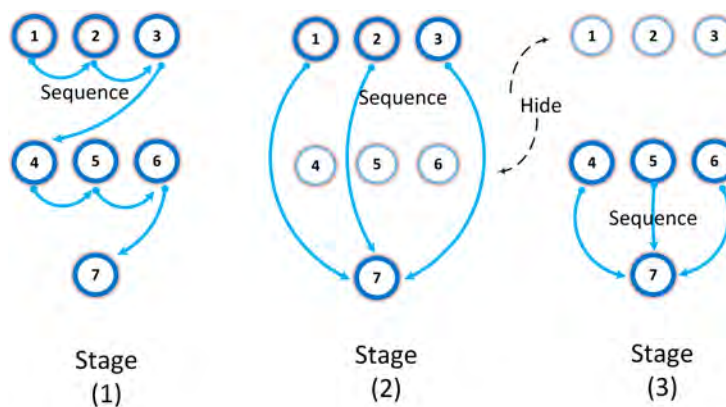


Figura 4.6: Escenarios de la interacción TUI (obtenida desde el test de interacción de la Sección 4.4).

- **Escenario 1:** El voluntario toca los objetivos del 1 al 7 en la secuencia indicada por los números (todos los objetivos deben tocados);

- **Escenario 2:** El voluntario toca los objetivos de acuerdo a la siguiente secuencia: 1 a 7, 2 a 7, y 3 a 7, correspondientes a este escenario. Los objetivos 4, 5 y 6 no son tocados porque se hallan ocultos para este escenario;
- **Escenario 3:** El voluntario toca los objetivos en esta secuencia: 4 a 7, 5 a 7, y 6 a 7. En esta etapa, los objetivos 1, 2 y 3 no son tocados porque se hallan ocultos para este escenario.

Los datos necesario para el cálculo de ACC , fueron obtenidos de la siguiente manera: Si el toque se hizo de acuerdo con la secuencia esperada, el intento era admitido y se incrementaba la cuenta de intentos positivos (+1). En otros casos, los sumaba el intento a los intentos negativos. No era permitido ir a al siguiente escenario, si toda la secuencia de toques de la etapa anterior no era completada. En cada prueba, la cuenta de los intentos positivos y negativos se reiniciaba. Los intentos totales, positivos y negativos, se almacenaban en un archivo de base de datos para su posterior procesamiento y análisis.

4.5.4. Pruebas con GBUI

En las pruebas con interfaz GBUI se utilizó la aplicación diseñada para el efecto, la cual implementaba el software de reconocimiento gestual. A diferencia de la prueba anterior, el usuario toca la Touchscreen del Smartphone durante los ensayos. Los objetivos en este caso, eran los gestos-objetivo, es decir, el voluntario tenía que ejecutar el gesto indicado en la secuencia establecida en cada etapa. Para capturar los gestos, se utilizó la pulsera gestual MYO, la cual fue descrita y estudiada en profundidad en la Sección 3.3 del Capítulo 3. Para este experimento, hemos adaptado el sistema de reconocimiento gestual a al dispositivo Android, fusionando en una sola aplicación el reconocimiento gestual y el control de interacción del experimento.

El dispositivo gestual se colocó en el antebrazo del voluntario, ya sea derecho o izquierdo, dependiendo si era diestro o zurdo. No era permitido cambiar el brazalete de un brazo a otro, una vez iniciado el test. La calibración por defecto se mantuvo y se aplicó la técnica indicada por Pomboza & Holgado [142] para evitar la pérdida de calibración en cada usuario. Esta técnica de calibración también es tratada en esta investigación en el Capítulo 3 en la Sección 3.6.3. Las secuencias de toque fueron realizadas utilizando el GCT diseñado para esta prueba y expuesto en la Figura 4.7. El GCT de la prueba descompone las secuencias siete o cuatro niveles al admitir al gesto unitario en cada nivel, el cual es una abstracción para seguir

una secuencia establecida, donde cada gesto es considerado un nivel. Las secuencias realizadas por los voluntarios se describen a continuación:

- Escenario 1: El voluntario realiza el gesto de inicio (gesto configurado para objetivo 1) y ejecuta los gestos para los gesto-objetivo del 2 al 6, para finalizar en el gesto-objetivo 7. Todos los gesto-objetivo son realizados.
- Escenario 2: Se ejecutan los gesto-objetivo 1, 2, 3 y 7.
- Escenario 3: Se realizan los gesto-objetivo 4, 5, 6 y 7.

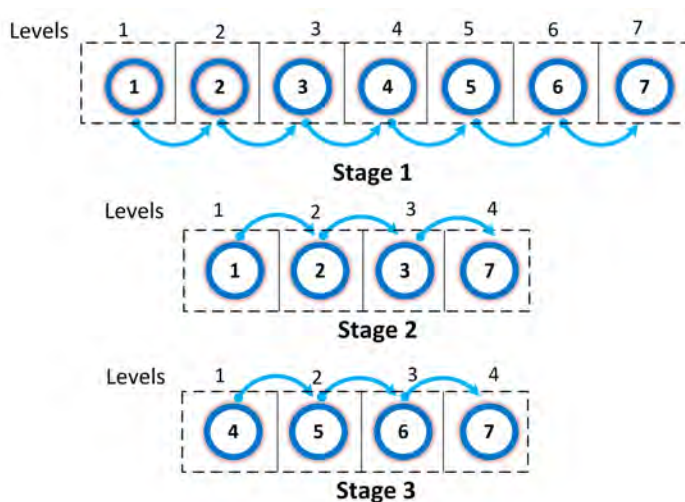


Figura 4.7: Escenarios y niveles del test de interacción por GBUI (obtenida desde el test de interacción de la Sección 4.4).

Para este tipo de interfaz, el voluntario también recibe notificaciones a través de mensajes de audio para cada gesto-objetivo. Si el gesto fue aceptado, el sonido de "moneda.^{er}a reproducido. De lo contrario, el sonido "plop.^{er}a reproducido. En caso de éxito (objetivo alcanzado en el orden esperado), el Smartphone emitía el mensaje de audio asociado y, se ponía a la espera del siguiente gesto-objetivo. Esta forma de trabajo, evitó la supervisión visual del dispositivo móvil por parte del usuario, avanzando en forma rápida y eficaz en la secuencia. Se estableció un tiempo de espera de máximo de 10 segundos en caso de no detectarse gesto alguno. Cumplido el plazo y al no detectarse gesto alguno o al no ser alcanzado el último objetivo (7), se cancelaba la lectura y se iniciaba otra secuencia. El cálculo de *ACC* fue similar a aquel realizado para la interfaz TUI.

4.6. Resultados

Aplicando el protocolo experimental descrito en la secciones anteriores, se realizaron las pruebas pertinentes en ambos tipos de interfaz (Sección 4.5), en las que participaron cada uno de los voluntarios. Posteriormente, el análisis comparativo de los dos tipos de interfaz fue realizado usando cada una de las posiciones anatómicas previstas en esta investigación. Los resultados se presentan a continuación.

4.6.1. TUI

Procediendo de acuerdo con la ecuación 4.2 hemos sido capaces de determinar el ID para la TUI (véase la Tabla 4.4, donde también se incluyen los valores utilizados para A y W). Como se observa, el ID tiene una mayor variación en el escenario 2 (Figura 4.6), porque la distancia entre cada dos objetivos requiere un mayor movimiento (por ejemplo, el movimiento del objetivo 1 al objetivo 7 en el escenario 2, alcanzó un $ID = 3,94$). Esto nos indica entonces que, la dificultad para alcanzar un objetivo puede aumentar cuando el Smartphone tiene un tamaño de pantalla más grande. Algo que era de esperarse y que confirma la validez de la Ley de Fitts. Entonces, debe evitarse una excesiva separación de objetivos con el fin de reducir el ID . Pero, el ID puede verse afectado también por el tamaño del objetivo cuando el Smartphone tiene una alta resolución con un pequeño tamaño de pantalla. En este caso, los puntos de interacción serían demasiado pequeños para poder aplicar el toque con el dedo de la mano. En este sentido, se han proporcionado algunas soluciones basadas en visualización usando un acercamiento (zoom) en áreas que quedan comprometidas por su resolución y la exactitud de toque del usuario. Es decir, el dispositivo acerca el área para que el usuario pueda tocar la opción deseada [28]. Esto también puede indicar que diseños triangulares con vértices al inicio y al final de la superficie táctil deben ser evitados.

En la Figura 4.8 podemos observar la media de intentos realizados por los voluntarios: los positivos o aciertos (+) al momento de asignar el objetivo; y, los negativos o errores (-) cometidos en cada ensayo. Los éxitos dan una idea clara de la exactitud de los voluntarios en posición normal (+) y especial (+). Por ejemplo, el voluntario 5 tuvo problemas para localizar el objetivo en posición normal. Sin embargo, en posición especial, alcanzó niveles similares a otros voluntarios. En el caso del voluntario 15, se observó que ocurrió lo contrario. También puede notarse una ligera disminución en el número medio de intentos en la posición especial (+) respecto a la posición normal (+). Esto significa que la mayoría de los usuarios

Tabla 4.4: ID alcanzado por la interfaz TUI

Objetivos	A (mm)	W (mm)	ID
1 a 2, 2 a 3, 4 a 5, 5 a 6	20	11,5	1,80
5 a 7	45	11,5	2,97
6 a 7, 4 a 7, 6 a 7	50	11,5	3,12
3 a 4	55	11,5	3,12
2 a 7	85	11,5	3,89
1 a 7, 3 a 7	88	11,5	3,94

ralentizan su destreza manual en posición especial. Sin embargo, en el caso de los voluntarios 3 y 6 ocurrió todo lo contrario, logrando un alto grado de éxito en posición especial. Algunos voluntarios informaron haber experimentado fatiga en la mano que sostuvo el dispositivo móvil y, en consecuencia, molestias al tratar de tocar el dispositivo en posición especial.

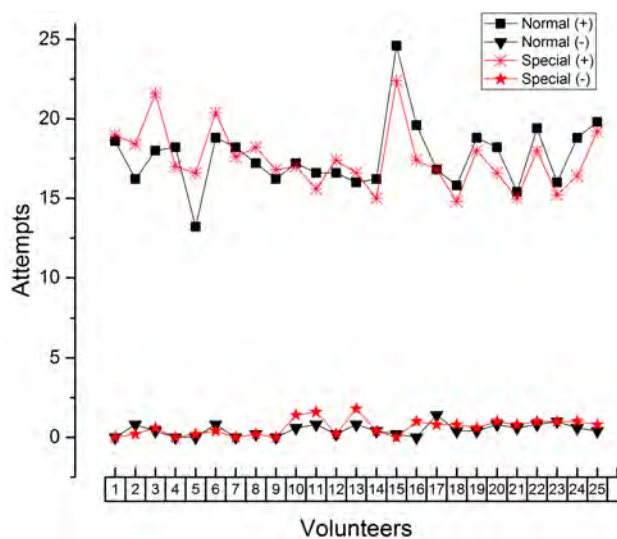


Figura 4.8: Intentos positivos y negativos usando TUI.

El promedio de intentos positivos en la posición normal alcanzó 17,616 ($DS = 2,131$) en contraste con el promedio de los intentos en la posición especial que alcanzó 17,48 ($DS = 1,936$). Así, a pesar de que el voluntario en algunos mejoró la posición de comodidad durante la interacción (al tomar el Smartphone con

una mano), los resultados no mejoraron. El promedio de intentos negativos (-) alcanzó 0,464 ($DS = 0,377$) en posición normal, y 0,632 ($DS = 0,522$) en posición especial. En este sentido, algunos voluntarios (2, 6 y 17) destacan en posición normal, y otros (10, 11 y 13) en posición especial. En cuanto a la ACC en la interfaz TUI en las posiciones estudiadas, la Figura 4.9 muestra una pérdida, aunque mínima, de ACC en la posición especial que alcanzó 0,965 ($DS = 0,029$) en comparación con la posición normal que alcanzó 0,974 ($DS = 0,021$). Excepcionalmente, algunos voluntarios alcanzaron una exactitud, similarmente alta, en las dos posiciones. Algunos voluntarios lograron una exactitud absoluta o total (a decir los voluntario 1, 4, 7 y 9), en ambas posiciones con este tipo de interfaz (ver Figura 4.9)

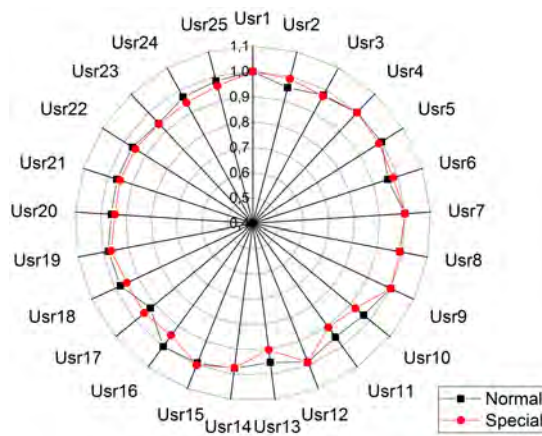


Figura 4.9: Niveles de ACC alcanzados usando TUI.

4.6.2. GBUI

La Figura 4.10 muestra el promedio de intentos realizados por cada voluntario al intentar alcanzar el gesto-objetivo en la interfaz gestual, para cada una de las posiciones estudiadas. Como en el caso anterior, se muestran los aciertos (+), en el momento de acertar con el gesto-objetivo y, los errores (-) cometidos en cada prueba. Se puede observar que, en este tipo de interfaz, el voluntario logró un mayor número de éxitos en la posición especial, en comparación con la posición normal. Se alcanzó un promedio de 7,416 ($DS = 1,696$) en la posición normal, en contraste con un promedio de 7,8 ($DS = 1,667$) en la posición especial. Además, podemos argumentar que el 60 % de los voluntarios mejoran los aciertos en posición especial, lo cual es notable para la interfaz GBUI. Además, como puede observarse

la variación es mínima entre una posición y otra. Tal vez entonces la mayor ventaja de este tipo de interfaz es que, no depende de la distancia o el tamaño de la *Zona de Importancia* (objetivo), y además, es independiente del contacto; podemos asumir entonces que, ajusta su comportamiento a cada usuario. Esto indica además, que el gesto se adapta mucho mejor a las necesidades de interacción del usuario sin causar fatiga excesiva. El promedio de intentos negativos (-) alcanzó 0,84 ($DS = 0,775$) en posición normal, y 1,008 ($DS = 1,038$) en posición especial y se mantuvo bajo excepto para los voluntarios 7, 8, 14 y 16 en posición especial.

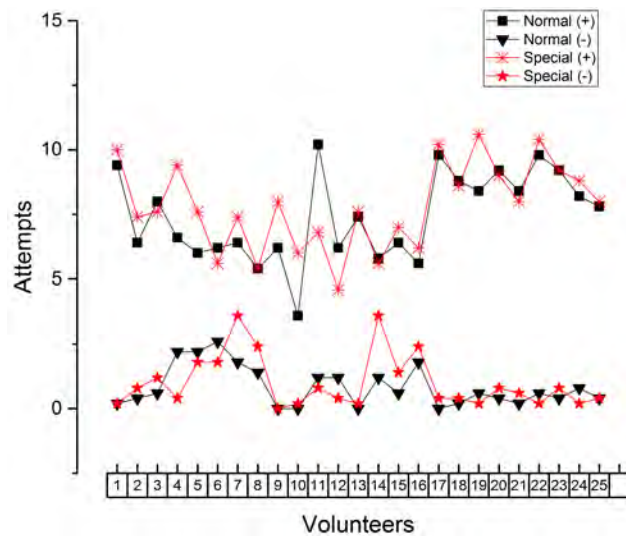


Figura 4.10: Intentos positivos y negativos usando GBUI.

Los niveles de ACC de la interfaz GBUI se muestran en la Figura 4.11. El ACC fue muy similar tanto en la posición normal ($ACC = 0,898$, $DS = 0,095$) como en posición especial ($ACC = 0,885$, $DS = 0,112$).

Para varios voluntarios la mejora en la exactitud fue evidente (tales como 4, 5 y 12) en posición especial, mientras en otros casos, la exactitud permaneció muy similar, tal vez independiente de la posición anatómica del voluntario. La mayoría de voluntarios, no tuvieron mayores problemas con este tipo de interfaz y se adaptaron directamente al uso de los gestos. Sólo el voluntario 9 fue capaz de alcanzar niveles absolutos de exactitud en las dos posiciones, mientras que sólo el voluntario 14 presentó una pérdida significativa de exactitud en posición especial. Esto indica, en cierta manera, que GBUI es en cierto grado independiente de la posición (ya sea normal o especial) adoptada por el voluntario para realizar el gesto.

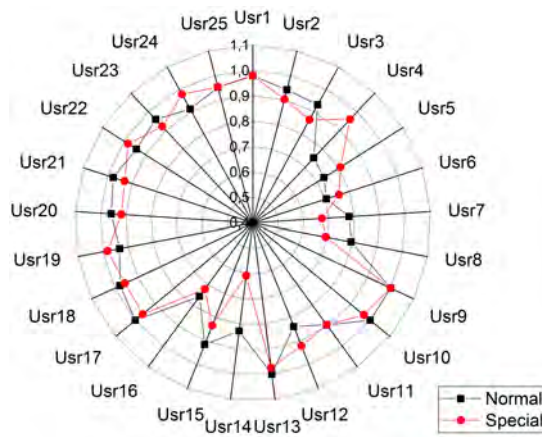


Figura 4.11: Niveles de ACC alcanzados usando GBUI.

4.6.3. Comparación TUI vs. GBUI

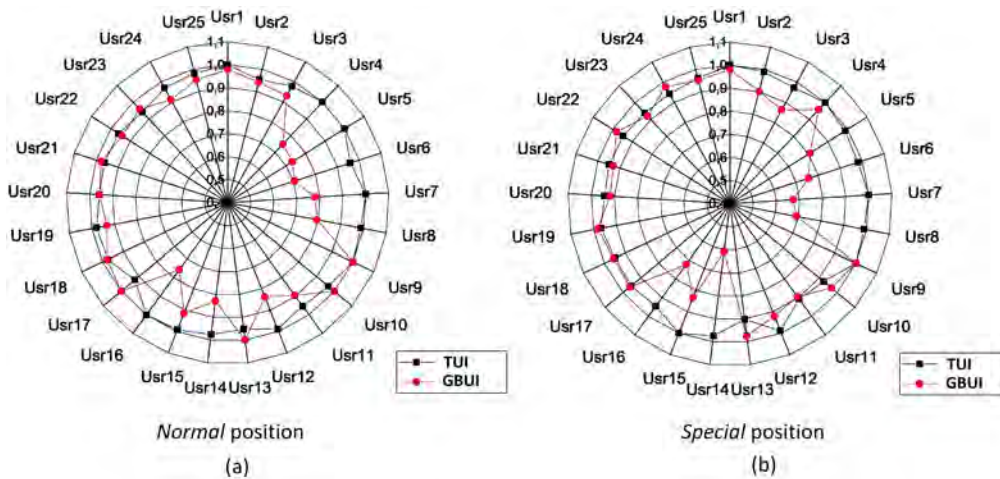


Figura 4.12: Comparación de niveles de ACC alcanzados por las interfaz TUI y GBUI.

Realizamos una comparación entre los dos tipos de interfaz estudiadas para dar un panorama general de su comportamiento. La Figura 4.12 nos permite observar el comportamiento de exactitud tanto en la interfaz TUI como en la interfaz GBUI. Cada interfaz ha sido comparada tomando en consideración la posición anatómica de uso sobre las cuales hemos orientado este estudio, es decir en posición *normal* como en posición *especial*. La Figura 4.12.a muestra que la exactitud alcanzada por los voluntarios que utilizan la interfaz GBUI en la posición *normal* es muy similar en muchos casos y excede en otros, a la exactitud lograda con la interfaz TUI en

las pruebas realizadas. En los voluntarios 4, 5, 6, 7, 8, 11, 12, 14 y 16 también se observó una variación notable de la exactitud (pero por debajo de 0,9) cuando se utilizó la interfaz GBUI, ya que para ellos la interfaz TUI fue más precisa. Los demás voluntarios parecen estar adecuadamente adaptados al uso de la GBUI en la posición normal, manteniendo siempre una similitud relativa. La interfaz TUI alcanzó una exactitud media de 0,974 ($DS = 0,021$) mientras que la interfaz GBUI alcanzó una exactitud media de 0,898 ($DS = 0,095$) en esta posición.

La figura 4.12.b muestra cómo la interfaz GBUI supera o coincide en exactitud con la interfaz TUI en posición *especial* (voluntarios 9, 10, 13, 17, 18, 19, 22 y 24). Hay excepciones (por debajo de 0,9), como los voluntarios 3, 5, 6, 7, 8, 11, 14, 15 y 16, en los cuales la interfaz GBUI muestra una disminución de la exactitud. Sin embargo, los demás voluntarios muestran un mejor acoplamiento con la interfaz gestual e incluso mejoran su exactitud en la posición *especial* (voluntarios 10, 13, 17, 18, 19, 22 y 24). La interfaz TUI alcanzó una exactitud media de 0.964 ($DS = 0,029$) mientras que la GBUI alcanzó una exactitud media de 0.885 ($SD = 0,112$) en esta posición.

4.7. Propuesta experimental

El análisis realizado nos ha llevado a la conclusión de que una interfaz GBUI puede tener la misma facilidad de uso y eficacia que la interfaz TUI, en especial si es aplicada a la gestión de llamadas telefónicas. Por lo tanto, una implementación software, basada en una interfaz GBUI para gestión de llamadas telefónicas puede ser muy adecuada. Con este objetivo, hemos diseñado e implementado una solución software basada en GBUI orientada a Smartphones a la cual hemos llamado SICLLE (Sistema Interactivo de Control de Llamadas de Emergencia). Esta aplicación puede ser particularmente útil, para personas que requieren interacción gestual por conveniencia o en cualquier necesidad especial (por ejemplo, enfermedad, limitación física, problemas visuales, etc.). En esta implementación, hacemos uso de todos los conceptos analizados y expuestos en las secciones anteriores.

4.7.1. Arquitectura de interfaz GBUI

La Figura 4.13 muestra la arquitectura utilizada para implementar SICLLE. Esta arquitectura ha sido pensada para ser usada en el desarrollo e implementación de una aplicación para gestión de llamadas telefónicas con interfaz GBUI en un Smartphone. El reconocimiento gestual es logrado usando señales EMG. Su diseño

se basa en capas y se describe a continuación:

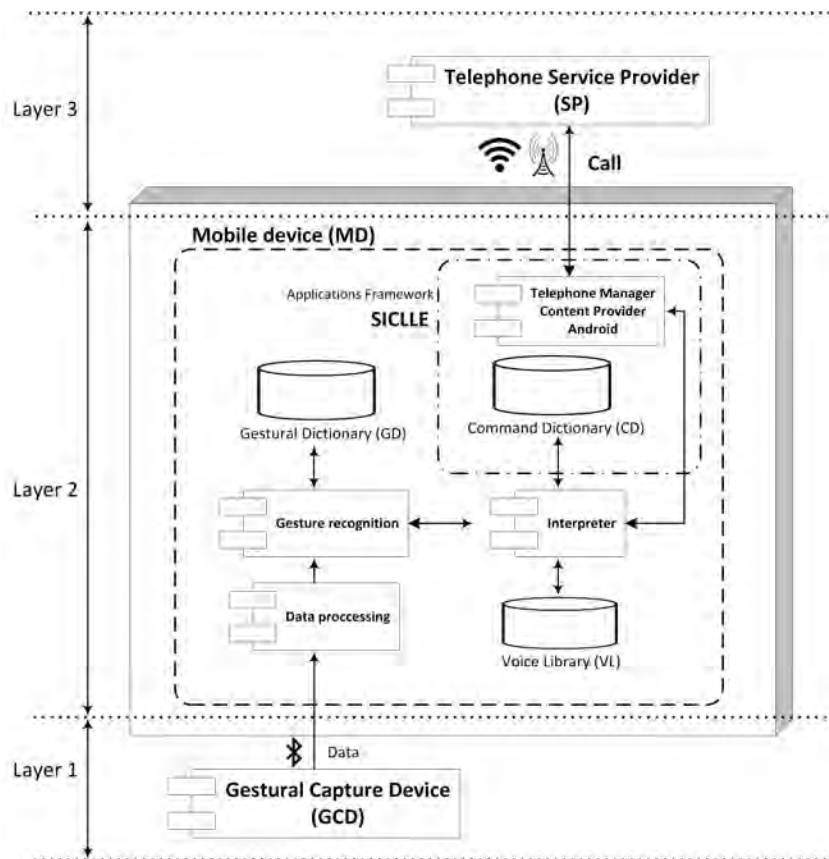


Figura 4.13: Arquitectura usada en el sistema SICLLE.

- Capa de Datos:** Esta capa es responsable de capturar los gestos del usuario. En SICLLE, esta capa se instancia utilizando la pulsera gestual o brazaletes MYO. Este dispositivo captura el gesto usando sus sensores sEMG dispuestos radialmente en el antebrazo del usuario. Las señales son capturadas, amplificadas, filtradas y codificadas para posteriormente ser enviadas usando protocolo Bluetooth (BLE) hacia la siguiente capa. Los detalles técnicos de la pulsera gestual fueron expuestos a profundidad en la Sección 3.3 del Capítulo 3.
- Capa de aplicación:** Esta capa se encarga del reconocimiento gestual y del procesamiento de los comandos. Esta capa está compuesta por el Smartphone o cualquier un Dispositivo Móvil (Mobile device, MD) que pueda recibir y hacer llamadas, compatible con Android. Dicho dispositivo recibe los datos

enviados por la capa inferior. En SICLLE, se eligió un Smartphone de gama media con procesador Qualcomm quadcore MSM8926 de 1,2 MHz, con 1 Gb de RAM, memoria flash de 8 Gb, compatible con Bluetooth 4.0, OS Android 5.1. Este dispositivo cumple con los requisitos de tamaño, peso y resolución establecidos en la Sección 4.3.

La interacción, es percibida por el sistema a través del gesto realizado por el usuario e identificado por el sistema. El gesto, se halla contenido en una biblioteca gestual y a partir de su definición SICLLE reacciona ejecutando una acción específica.

El MD, recibe los datos enviados por la Capa de Datos. Luego, los datos se procesan a través de varias etapas. Después de aplicar un proceso de reconocimiento gestual, el MD logra identificar un gesto al compararlo con todos los patrones gestuales almacenados y previamente definidos en una Librería o biblioteca gestual (Gestural Dictionary, GD). Una vez identificado el gesto, se le asocia un comando desde el Diccionario de Comandos (Command Dictionary, CD) para ser ejecutados finalmente por el SO del dispositivo móvil. El comando puede ser realizado antes, después o durante la llamada, todo depende del usuario. Además, se ha implementado una biblioteca de mensajes de voz (Voice Library, VL), cuyo principal objetivo es, almacenar etiquetas de audio para notificar a los usuarios de una acción seleccionada (opción). Varios mensajes de voz informan al usuario de todas las acción elegidas, así como cualquier otra en curso.

- **Capa de proveedor de servicio telefónico:** Esta conformada por el Proveedor de Servicios (Service Provider, SP) de telefonía móvil, al cual esté sujeto el dispositivo Smartphone por contrato suscrito por su propietario. El SP, es el responsable de establecer la llamada a través de la red telefónica, ya sea fija o móvil, y tiene en cuenta variables adicionales tales como: balance, coste, protocolo, etc., para proporcionar el servicio requerido. Esta capa no es administrada por SICLLE, por lo cual una llamada puede ser establecida o no dependiendo de la calidad del servicio, entre otras circunstancias, que podrían afectarla como llamada en espera, transferencia de llamada, etc.

4.7.2. Librería gestual y comandos

Los comandos implementados en SICLLE a través de gestos se describen en la Tabla 4.5 (gestos similares se han utilizado en [142, 175, 160, 126, 145]). Cada gesto

es un comando que está asociado con una acción específica dedicadas a lograr la gestión de llamadas telefónicas en un Smartphone.

Tabla 4.5: Comandos gestuales usados en SICLLE

Comando	Descripción de la acción
Mano abierta	Inicializa la interacción. En el siguiente nivel del GCT realiza una llamada urgente al contacto seleccionado como contacto por defecto.
Palma dentro	Cancela un comando. En modo agenda permite a los usuarios moverse de un contacto a la izquierda
Palma fuera	Acepta un comando. En modo agenda permite al usuario moverse un contacto a la derecha.
Mano cerrada	Cierra la llamada actual. En modo agenda asigna un contacto como contacto por defecto para una llamada urgente.

SICLLE, una vez instalado y ejecutado, se comporta como un programa residente en memoria (Terminate-and-stay-resident program, TSR) del dispositivo, siempre a la espera del gesto de apertura. Tan pronto como se inicia la interacción, el software está a la espera de un comando gestual para la gestión de llamadas o manejo de la guía telefónica (SICLLE también incorpora su propia agenda telefónica).

4.7.3. Diseño de SICLLE

El diseño de una aplicación basada en interfaz GBUI requiere resolver dos interrogantes principales:

- Establecer un conjunto de gestos de control para especificar la forma de administración y manejo de llamadas usando los gestos reconocidos.
- Diseñar las GUIs a mostrar, tanto para configuración como para manejo de la aplicación, aprovechando los recursos de una TUI, la cual se ejecuta sobre el SO del Smartphone.

La interacción entre el usuario y la GBUI debe realizarse de acuerdo al Árbol de Control Gestual (GCT) mostrado en la Figura 4.14. GCT indica las secuencias

válidas de los gestos que son permitidos por la interfaz GBUI de SICLLE.

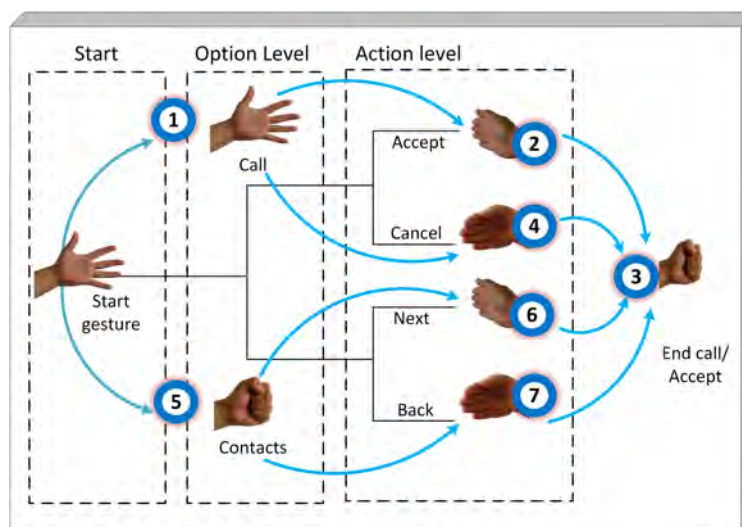


Figura 4.14: El Árbol de Control Gestual (GCT) definido en el sistema SICLLE.

El GCT definido en SICLLE tiene el siguiente modo de operación:

- Escenario 1: El usuario inicia la interacción con SICLLE con el comando *Mano abierta*. Luego en el nivel de opciones ejecuta el comando (*Mano abierta*) para realizar una llamada al contacto definido por defecto. En el nivel de acción acepta (*Palma fuera*) o cancela (*Palma dentro*) dicha solicitud de llamada.
- Escenario 2: El usuario da inicio a la interacción con el comando *Mano abierta*. En el nivel de opción, activa el uso de la agenda con el comando *Mano cerrada*. Posteriormente en el nivel de acción, navega hacia la izquierda *Palma dentro* o hacia la derecha *Palma fuera* entre los contactos y asigna al nuevo contacto por defecto con el comando *Mano cerrada*.
- Escenario 3: El usuario cancela una llamada en curso o cualquier otra acción asociada a la aplicación con el comando *Mano cerra*.

SICLLE, como se indicó anteriormente, incluye y aprovecha también algunas GUI sobre TUI que, facilitan su configuración y pueden ser gestionadas usando la pantalla táctil o usando comandos gestuales. La Figura 4.15 muestra algunas capturas de pantalla de las opciones principales con que cuenta SICLLE, las cuales brevemente describimos:

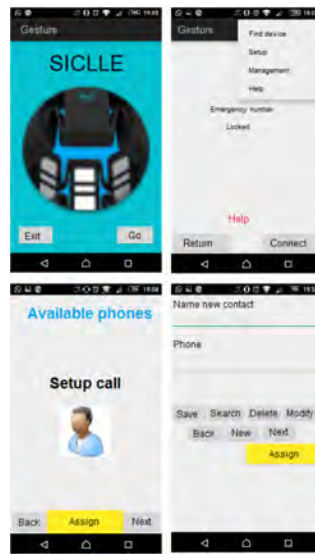


Figura 4.15: Capturas de pantalla de la aplicación SICLLE.

- **Find device:** Esta opción permite al usuario vincular el dispositivo de captura gestual con el Smartphone. Si hay varios dispositivos de captura gestual, sólo uno se asociará a la vez. La aplicación habilita la comunicación Bluetooth del dispositivo móvil si es necesario.
- **Setup:** Permite asignar un gesto a una acción o comando, teniendo en cuenta los gestos implementados e incluidos en el diccionario gestual. SICLLE permite al usuario personalizar su interacción gestual.
- **Management:** Esta opción permite acceder a la guía telefónica del teléfono inteligente, el cual comparte su agenda telefónica con la guía telefónica de SICLLE. El usuario puede agregar, modificar y eliminar contactos en las dos guías telefónicas. Una configuración adecuada, logra establecer la asignación de un número de contacto como pre-seleccionado o contacto por defecto, para una llamada emergente. SICLLE muestra la foto del contacto, si aquella está disponible.
- **Help:** Muestra en la pantalla del Smartphone todos los comandos gestuales (gestos) disponibles en la aplicación, incluyendo una descripción de la función de cada uno de ellos. Los mensajes o etiquetas de voz están relacionados con cada comando.

4.8. Conclusiones

La interacción hombre-máquina se transforma y junto a ella la tecnología utilizada para lograrlo. Así, los avances tecnológicos en la interacción natural están enriqueciendo la experiencia del usuario en aplicaciones muy diversas. El gesto y la interacción gestual, usados como una forma de interfaz natural de usuario, ha demostrado su competencia en diferentes contextos, tanto para controlar como para administrar dispositivos eléctricos y electrónicos. El presente estudio muestra la gran facilidad de uso, exactitud y eficacia que una interfaz de usuario basada en gestos (GBUI) puede llegar a alcanzar, en contraste con la tradicional TUI. Las dos ejecutadas y comparadas usando un teléfono inteligente. Consecuentemente, hemos logrado dos aportaciones principales: la primera, el análisis comparativo de ambas interfaces (GBUI vs. TUI) y la segunda, una aplicación basada en GBUI para la gestión de llamadas telefónicas llamada SICLLE, diseñada para Smartphones.

El análisis de los dos tipos de interfaz, GBUI y TUI, ha sido realizado con la participación de veinticinco voluntarios y, considerando dos posiciones básicas de uso, muy comunes entre los usuarios de teléfonos inteligentes, que son: la llamada posición “normal” que es aquella en la cual el individuo está de pie; y la llamada posición “especial” que es aquella en la cual el individuo está acostado. La primera posición es común en individuos sanos, mientras que la segunda es muy común en personas enfermas, discapacitadas o adoptada por simple comodidad.

Este análisis evidenció la alta exactitud media que la interfaz GBUI puede alcanzar ($ACC = 89,8\%$), en contraste con la mayor exactitud media demostrada por la interfaz TUI ($ACC = 97,4\%$), las dos en posición “normal”. Mientras que en la posición “especial”, GBUI logró un ACC promedio de $88,5\%$, en contraste con TUI que alcanzó un ACC promedio de $96,5\%$. Si bien los porcentajes de ACC promedio de GBUI, aunque no son comparables con la alta tasa de exactitud alcanzada por la TUI, son muy razonables para esta nueva forma de interacción. GBUI, garantiza una buena respuesta, independientemente de la posición anatómica que adopte del usuario (sólo $1,3\%$ de diferencia entre las dos posiciones).

Por otro lado, GBUI tiene propiedades adicionales con respecto a TUI, como el hecho que en GBUI no hay oclusión, es independiente de la luminosidad, libera al usuario del seguimiento visual de control y ofrece grandes posibilidades de accesibilidad. Además, GBUI es independiente del objeto a controlar y no requiere ningún contacto durante la interacción. Todo esto nos sugiere que usuarios con impedimentos visuales podrían obtener muchos beneficios con una GBUI instalada en su dispositivo móvil. Si tomamos en cuenta la arquitectura propuesta, GBUI

tiene ventajas únicas que sólo la interacción sin contacto (utilizando sEMG y comunicación Bluetooth) y gestos libres de la mano pueden ofrecer, mejorando la experiencia de usuario y logrando personalizar el contexto de aplicación.

El GCT implementado en SICLLE permite reutilizar el gesto en sus diferentes niveles y además, facilita el seguimiento y control de las acciones, lo cual nos hace pensar que la idea es aplicable a contextos de control basados en interfaces gestuales. En este mismo sentido, la arquitectura GBUI presentada en este documento, puede ser extensible a otros tipos de aplicaciones en dispositivos móviles, así como, los resultados de la comparación entre GBUI y TUI puede ser utilizados como una referencia en otros contextos, incluyendo investigaciones de ayuda o soporte para las personas con necesidades especiales.

Finalmente, todos estos hechos parecen indicar que la tecnología en interacción gestual avanza a paso firme, y nos hace imaginar una gran variedad de posibilidades de uso y aplicación. Más aún, si agregamos el hecho de que actualmente, que existen dispositivos wearables que incorporan capacidad computacional, movilidad y múltiples protocolos de comunicación, con los cuales, múltiples y variadas soluciones pueden ser implementadas.

Capítulo 5

El gesto como lenguaje de control

La Figura 5.1 trata de mostrar los aspectos principales tratados en esta tesis, hasta llegar a realizar un nuevo enfoque que define al gesto como un lenguaje, sujeto a variables y restricciones. Las características del lenguaje gestual son parte del confort asociado a aspecto físico humano sujeto a la capacidad de procesamiento y algoritmos que se ejecutan en un controlador.

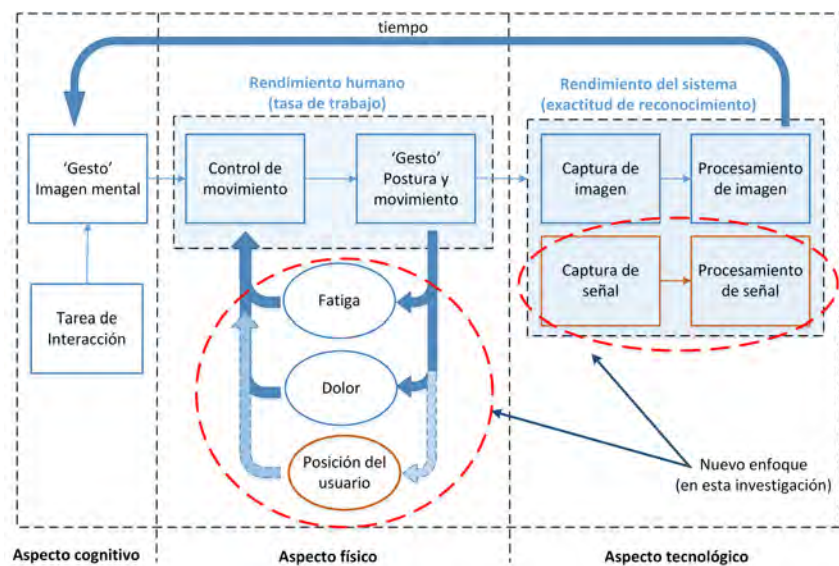


Figura 5.1: Aspectos principales del gesto usado como un lenguaje.

La captura del gesto del usuario ha sido lograda usando muchas y extensas

técnicas, unas externas al usuario (imagen, guantes, etc.) y otras junto a este (dispositivos wearables). El gesto del usuario capturado usando señales electromiográficas (EMG) generadas por el mismo usuario al formar un gesto, quizá es la que genera mayores expectativas como describe Wheeler et al. [192, 191]. A la fecha, varias investigaciones que capturan el gesto a través electrodos sEMG (EMG de superficie) han sido presentados, obteniendo óptimos resultados de reconocimiento aplicados a diferentes propósitos [201, 90, 71]. Sin embargo, muchos trabajos han sido expuestos bajo el enfoque del gesto de la mano como un comando simple (un gesto-un dispositivo-una funcionalidad) de control [142, 126, 145]. Otros trabajos exponen el uso del gesto (corporal, facial, manual y otros) en ámbitos de ayuda o soporte al usuario [205, 45, 167]. En ambos casos, la complejidad del lenguaje gestual está relacionada directamente con el comando asignado a cada gesto, lo cual no es suficiente, ya que el sistema tiene que administrar en muchos casos dispositivos domésticos distintos y con diferente funcionalidad. Por esta razón, un comando complejo (un gesto-varios dispositivos-distinta funcionalidad) es requerido. Además, lograr un lenguaje gestual de características confortables, de fácil aprendizaje, que maneje gestos intuitivos, de fácil memorización, y sin restricciones físicas es el nuevo objetivo [128, 87].

En la mayoría de trabajos de investigación, se realizan implementaciones enfocadas u orientadas a establecer el reconocimiento gestual relacionado a una acción en concreto, formando un conjunto de órdenes unitarias y preestablecidas, imposibles de cambiar o modificar en su contexto. Por ejemplo un gesto como *mano cerrada* indicaría *Alto* y no podría ser usado para otro contexto, en estos enfoques. Un sistema con este planteamiento, requerirá de otro gesto como por ejemplo *mano abierta* relacionado a otra acción para que se ejecute la orden *Continuar*. Como puede observarse, un gesto mantiene un enfoque unitario, en relación al comando pensado para cada uno de estos casos. Por otro lado, la interacción usando interfaces gestuales aplicadas al control de dispositivos de casa no es muy común porque aumenta la complejidad y el tamaño del sistema [126]. Para compensarlo, la aparición de dispositivos wearables que integran sensores más complejos y precisos, hace pensar en una integración del usuario a ambientes de interacción, inteligentes y controlados [4, 25, 104, 96, 166]. Nosotros proponemos una nueva idea, la cual logra la definición e implementación de un lenguaje gestual, que partiendo de gestos atómicos (simples), consigue formar parte de un protocolo gestual (complejo), especialmente pensado para lograr una interacción total con varios dispositivos usando incluso el mismo gesto. De tal manera que, si tomamos el ejemplo anterior, cuando

el usuario desee dar una orden de *Alto* podría usar *mano cerrada* pero, el mismo gesto en otro contexto podría significar *Encender* un dispositivo en particular, a más de *Continuar* en otro.

El potencial que se logra a través del protocolo propuesto es muy amplio, establece el inicio de una forma de comunicación orientada a optimizar el gesto y maximizar su aprovechamiento con el menor número de gestos posibles. A este protocolo gestual lo hemos llamado *Árbol de Control Gestual* (Gestural Control Tree, GCT). Un trabajo previo de los autores fue presentado en [142] en el cual se mostraban resultados preliminares. En esta estructura, el contexto de cada gesto, está expresado por la jerarquía de control que define un protocolo, para formar posteriormente un comando, del cual el gesto es parte principal. Esta estructura servirá para gestionar el control de varios dispositivos y sus funcionalidades, universalizando su aplicación. Por ejemplo, un comando visto desde esta perspectiva, podría ser concebido como un conjunto de gestos orientados a conseguir un control total de todas las funcionalidades de un dispositivo de casa, algo similar a control remoto universal. En este estudio, se hace uso del reconocimiento gestual basado en señales electromiográficas, usando técnicas de aprendizaje computacional expuestas anteriormente y que nos han dado excelentes resultados.

Este capítulo plantea una forma diferente de concepción del lenguaje gestual aplicado a control de dispositivos de casa o del hogar. Para lograrlo toma como antecedente la comunicación humana a través de las propiedades del lenguaje, en donde el uso de una estructura llamada oración (sujeto, verbo y complemento) define la idea o el concepto [177, 86]. Hemos acudido a una estructura de datos jerárquica (grafo-dirigido-acíclico) con características propias (raíz, nodos, niveles, vértices, arcos, etc.) definidas e integradas para lograr el comando. En este sentido, el *Árbol de Control Gestual* (GCT) es definido en un concepto general llamado *Activity*, logrado mediante una tripleta la cual define un comando complejo (un gesto?varios dispositivos-varios comandos). Para capturar el gesto del usuario hemos usado a la pulsera gestual MYO. Se ha independizado al sistema de control del algoritmo de reconocimiento del fabricante al desarrollar un algoritmo de clasificación y reconocimiento gestual e implementado sobre un sistema empotrado. Esto nos permite ampliar la librería gestual lo suficiente para esta investigación. Finalmente, hemos probado la efectividad del GCT al implementarlo en un sistema de control (SC-ACG) para Dispositivos de Casa (Home Devices,HD) heterogéneos que utilizan diferentes protocolos de comunicación (WiFi, Bluetooth e iR). Todo ello usando y optimizando la misma librería gestual. Con el objeto de evaluar la

estructura propuesta, se ha realizado un estudio con veinticinco voluntarios los cuales han ejecutado el GCT en dos posiciones básicas. Dichas posiciones son: posición sentado y posición acostado. Posteriormente, poner en funcionamiento un sistema de control gestual que incorpora una Interfaz Natural de Usuario de Tipo gestual (INU-G) el cual implementa el concepto de GCT, orientado al control de dispositivos de casa. Finalmente, evaluar el grado de confortabilidad del gesto.

5.1. Métricas gestuales

Las métricas del gesto que planteamos en el uso del gesto como un lenguaje, tratan de orientarse a grado de confort que experimenta el usuario en el uso del gesto. Es indudable, que un gesto cómodo podrá tener mejor grado de aceptación en el usuario que aquel que le causa dolor o fatiga. Anteriormente, hemos analizado el gesto desde varios puntos de vista, ahora corresponde evaluar al gesto, tomando en consideración aquello que opina el usuario. Esto nos ayudará principalmente a conocer si la librería gestual propuesta es adecuada y además, aquellos gestos que cumplen características de confortabilidad.

La formación de un gesto y el estrés que causa en el usuario el tratar de lograrlo, afecta a la formación del gesto y a la consistencia de las posturas de la mano resultantes [151]. Es evidente, que una postura y sus variaciones pueden producir agotamiento debido a las exigencia físicas y cognitivas, las cuales provocan directamente la llamada comodidad o confortabilidad (comfort, en inglés) del gesto en el usuario. El confort esta relacionado con la fuerza necesaria para realizar el gesto [176], de allí que unos gestos sean más fáciles de realizar que otros. A pesar de un gesto ser confortable, con el transcurso del tiempo, puede pasara a ser incómodo (discomfort, en inglés) o inclusive doloroso y el músculo, o conjunto de músculos involucrados se ven sometidos a mayor tensión. El confort puede ser medido y generalmente se lo considera un escalar [176, 177]. Nosotros mediremos al gesto desde el punto de vista humano, mediante el confort, y desde el punto de vista de reconocimiento, mediante la exactitud.

5.1.1. Confort

El confort del gesto estará dado por una escala de 1 a 5 en donde: 1 es doloroso, 2 es incómodo, 3 es fácil, 4 es cómodo y 5 es muy cómodo. Una escala similar fue usada por Rempel et al. [151] para evaluar gestos del lenguaje de señas americano. La evaluación del gesto debe realizarla el usuario con la guía del investigador, con-

siderando la ejecución del gesto dentro de los parámetros experimentales previstos. La siguiente expresión nos dará una clara idea de la complejidad, en la escala planteada y en general una idea de la confortabilidad de la biblioteca gestual planteada e implementada.

La complejidad gestual (C_g) de un gesto G_i de la biblioteca implementada es una proporción de valores iguales a G_i en el conjunto de datos ($G_1, G_2, G_3, \dots, G_N$). Es decir, la complejidad gestual es la frecuencia absoluta dividida por el número de elementos N :

$$C_g = \frac{n_i}{N} \times 100 \quad (5.1)$$

donde C_g esta expresada en porcentaje en la escala implementada. n_i es el total de valores de G_i , siendo ($G_1, G_2, G_3, \dots, G_N$) el conjunto de datos.

La suma de la complejidad gestual parciales debe ser siempre 1. Supongamos en el la biblioteca gestual tenemos k gestos diferentes, entonces:

$$\sum_{i=1}^k C_g i = C_g 1 + C_g 2 + \dots + C_g k = 1 \quad (5.2)$$

Un gesto confortable es aquel que minimiza los factores de agotamiento, cansancio físico o dolor en el usuario. Por ello la métrica de C_g nos mostrará la complejidad de nuestra biblioteca gestual. Mientras, el grado de confort nos permitirá identificar aquellas posiciones gestuales confortables de otras dolorosas o demasiado forzadas.

5.1.2. Exactitud del comando gestual

La exactitud del comando trata de medir la eficacia de reconocimiento del gesto, basado en un algoritmo de reconocimiento y aplicado sobre un sistema de control gestual. Básicamente, esta dado por la exactitud de generación del gesto-comando por parte del usuario, y en porcentaje la podemos definir como:

$$ACC_c = \frac{totalgestos - gestosnoreconocidos}{totaldegestos} \times 100\% \quad (5.3)$$

donde ACC_c es la exactitud del comando. *Totalgestos* es el número total de gestos realizado con el afán de completar un comando. *gestosnoreconocidos* es el número de gestos fallidos o erróneos generados por el usuario. Esta definición la podremos adaptar para el caso de una *Activity*.

5.2. Estudio de confortabilidad

Para medir el grado de confort del usuario al realizar el gesto hemos realizado un estudio aplicando la biblioteca gestual que se describe en la Figura 5.2. Dicha biblioteca ha sido utilizada anteriormente en esta investigación en capítulos anteriores, y consta de 13 gestos. Cada gesto esta dedicado a lograr convertirse en un comando de control gestual, que podrá ser aplicados en dispositivos eléctricos y electrónicos.

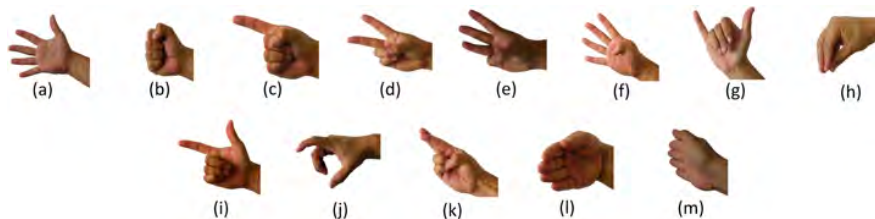


Figura 5.2: Gestos propuestos para estudio de Confortabilidad y control.

5.3. Fundamentos del Árbol de Control Gestual (GCT)

Lograr una diversa funcionalidad atendiendo a la misma biblioteca gestual es imposible si partimos de la idea de un comando simple. Si deseamos optimizar una biblioteca gestual y alcanzar una máxima funcionalidad en diferentes dispositivos, es necesario usar *comandos complejos*. Los *comandos complejos* son necesarios por dos razones.

1. Porque un comando complejo proporciona una forma compacta de expresar una acción en un dispositivo doméstico que personaliza su aplicación. Y,
2. Porque el comando complejo se puede romper en órdenes atómicas simples, los gestos. Esto logra reducir la complejidad gestual y extiende la funcionalidad del vocabulario gestual.

En este sentido, el Árbol de Control Gestual es una estructura que define al gesto en niveles de acción hasta lograr formar un comando complejo, al cual hemos llamado *Activity*. La *Activity* esta es una tripleta de comandos y que define, en sí misma, un comando complejo. La *Activity* resume el concepto de: un gesto - varios dispositivos - varias funcionalidades.

Para lograr esta estructura llamada GCT, debemos buscar la forma de expresar el gesto en componentes atómicos agrupables que logren el objetivo propuesto. Para lograrlo hemos usado la teoría de grafos como fundamento científico que logra plasmar la idea del GCT.

5.3.1. Análisis y definiciones

Definición 1 Un **grafo dirigido** G es un par (V, A) , en el que V es un conjunto ($V \neq \emptyset$) cuyos elementos se llaman **vértices**, y $A \subseteq \{(a, b) \in V \times V : a \neq b\}$ una familia de pares ordenados de **vértices** que llamaremos **arcos**, donde por definición un arco va del primer nodo (a) al segundo nodo (b) dentro del par. Un **bucle** es un arco cuyos extremos son el mismo vértice. Un bucle añade uno al grado de entrada y uno al grado de salida.

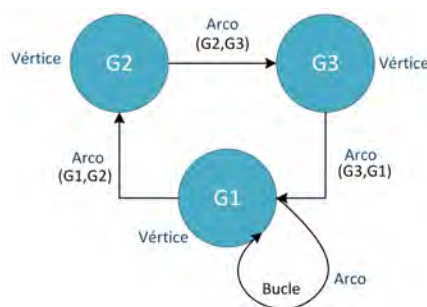


Figura 5.3: Componentes de un grafo.

Definición 2 Un **grafo dirigido** es **completo** si hay al menos un arco uniendo cada par de vértices distintos. Un grafo no dirigido diremos que es **bipartido** si, existe una partición X, Y del conjunto de vértices de forma que toda arista tiene un extremo en X y otro en Y . Un grafo dirigido es **bipartido** si lo es su grafo no dirigido asociado. Un **grafo bipartido** es **completo** si cada vértice de X está unido con cada vértice de Y .

Definición 3 Sean $G = (V, A)$ y $H = (V', A')$ dos grafos. H es un subgrafo de G si $V' \subseteq V$ y $A' \subseteq A$. Diremos que un subgrafo H de un grafo G es **generador** si sus conjuntos de arcos son iguales. Y dado $V' \subseteq V$ y $V' \neq \emptyset$, se llama subgrafo generador o inducido por V' al subgrafo maximal de grado G cuyo conjunto de vértices es V'

Definición 4 Designamos por $\Gamma(v)$ al conjunto de vértices adyacentes a v . Llamamos grado de salida $d_S(v)$ de un grafo dirigido G de vértice v al número de arcos salientes de v . Grado de entrada $d_E(v)$ de un vértice v al número de arcos entrantes en v . Grado de un vértice es la suma de estos dos grados. Análogamente, se puede definir $\Gamma(v)$ y $\Gamma^{-1}(v)$. Entonces, sea $G = (V, A)$ un grafo, luego

$$\sum_{v \in V} d_G(v) = 2 \text{card}(A) \quad (5.4)$$

Sea $G = (V, A)$ un grafo dirigido, entonces:

$$\sum_{v \in V} d_S(v) = \sum_{v \in V} d_E(v) = \text{card}(A) \quad (5.5)$$

Definición 5 Una cadena es una sucesión finita $W = v_0 E_1 v_1, \dots, E_k v_k$ cuyos términos son alternativamente arcos y aristas. La **longitud** de una cadena es el número de arcos que contiene. Una **cadena simple** es una cadena con todos sus arcos distintos. Un **camino** es una cadena con todos sus arcos distintos. Una **cadena cerrada** es una cadena de longitud no nula en donde el arco inicial y final coinciden. Un **circuito** es una cadena simple cerrada con todos sus arcos distintos. Dos vértices u y V están **conectados** si existe un camino de u a v y viceversa. Un grafo es **conexo** si todo par de vértices está conectado. Un grafo dirigido es **débilmente conexo** si su grafo no dirigido asociado es conexo.

Definición 6 Diremos que G es un **árbol** si G es conexo y acíclico. Diremos que T es un árbol generador de un grafo G si T es árbol y subgrafo generador de G . En un árbol dos vértices cualesquiera están unidos por un único camino. Un grafo G es conexo si y solo si tiene árbol generador. Si G es un árbol, entonces el número de arcos (aristas) es igual al número de vértices menos uno. Todo árbol T no trivial (más de un vértice) tiene al menos dos vértices de grado 1.

Definición 7 Sea T un árbol. Eligiendo un vértice v_R de T que llamamos **raíz**, al ser el árbol conexo, todo otro vértice estará conectado con v_R . Podemos entonces definir un grafo dirigido $T(v_R)$ donde todos los arcos sean extremos finales de un camino que se inicia en v_R . A este árbol lo llamaremos **árbol enraizado** en v_R . Sea T un árbol enraizado y u un vértice de T . Llamamos **nivel de vértice** u a la longitud del camino que va de la raíz a dicho vértice. La **altura** de un árbol es el valor del nivel máximo.

Definición 8 Sea T un árbol con raíz v_R . Supongamos que x, y, z son vértices de T y que $v_R v_1, \dots, v_{(n-1)} v_n$ es un camino en T , entonces:

$v_{(n-1)}$ es el padre de v_n . $v_0, \dots, v_{(n-1)}$ son los antepasados de v_n . v_n es el hijo de $v_{(n-1)}$.

Si x es antepasado de y , entonces y es un **descendiente** de x . Si x e y son hijos de z , entonces z e y son **hermanos**. Si x no tiene hijos diremos que es un **arco terminal**. Si x no es un arco terminal diremos que es **interno**. El subgrafo de T que consiste en x y todos sus descendientes, con x como raíz se llama **subárbol** de T que tiene a x como raíz.

La Figura 5.4 presenta la estructura del árbol y sus niveles.

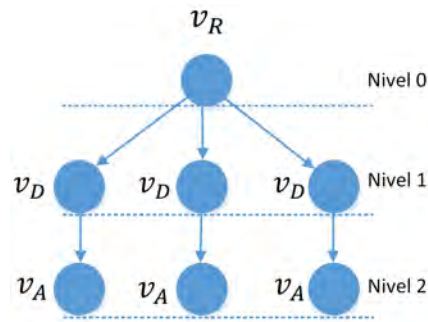


Figura 5.4: Estructura jerárquica basada en tres niveles.

Definición 9 Grafo **acíclico**. Un grafo dirigido no tiene circuitos si y solo si existe una numeración de los vértices para la que se cumple la condición si (i, j) es un arco del grafo entonces $i < j$. Con esta numeración las ecuaciones de Bellman pueden ser reemplazados por $u_1 = 0$, entonces

$$u_j = \min_{k < j} \{u_k + w_{kj}\} \quad j = 2, \dots, n \quad (5.6)$$

5.3.2. Estructura del GCT

El control gestual pensado como una estructura jerárquica permite lograr la atomicidad necesaria para estructurar comandos complejos en forma sencilla y clara. Planteado de esta manera, se ha pensado en una estructura tipo árbol, basada en tres niveles: Nivel cero o vértice raíz, nivel 1 o vértice de dispositivo y nivel 2 o vértice de acción. La suma de los componentes atómicos en cada vértice tendrá como resultado el comando. Cada vértice es un gesto, por lo que un $v_R \rightarrow G_R$, un $v_D \rightarrow G_D$ y un $v_A \rightarrow G_A$. La Figura 5.5 presenta el GCT definido y sus niveles.

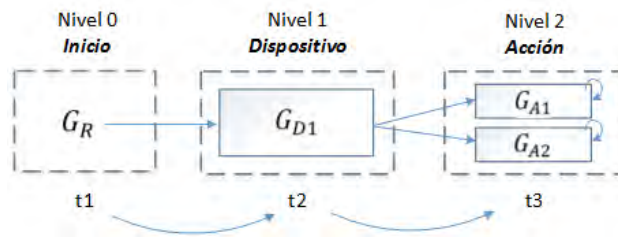


Figura 5.5: Estructura jerárquica basada en tres niveles.

- Nivel cero o vértice raíz (G_R). Esta pensado como aquel nivel que contendrá al gesto que iniciará la escucha para formar la estructura. Deberá estar sujeto a parámetros temporales, bajo los cuales garantiza la formación de un comando complejo. Solo puede existir un gesto raíz. Por definición, es un nodo al cual se adjuntan propiedades como: numero de dispositivos, tiempo de espera (segundos), parada (Booleano) y estado (Booleano).
- Nivel 1 o vértice de dispositivo (G_D). Es el nivel donde se agrupará a los n gestos que estarán relacionados con los D dispositivos $D = \{1, \dots, n\}$ dispositivos. Cada gesto logrará el acceso a un dispositivo y deberá contener las características necesarias para lograrlo. Su contenido puede debe guardar correspondencia con las f funcionalidades $f = \{1, \dots, n\}$ que disponga el dispositivo pues servirá para orientar las A acciones previstas en el mismo. Por definición es un nodo, al cual se adjuntan propiedades como: protocolo de comunicación (WiFi, Bluetooth, iR), tiempo espera (segundos), parada (Booleano) y estado (Booleano).
- Nivel 2 o vértice de acción (G_A). En este nivel se agrupan las funcionalidades f de los dispositivos, asignadas a un comando gestual específico. Deberán existir tantos comandos como funcionalidades a las cuales se desea acceder en el dispositivo. Todo nodo de este nivel deberá estar obligatoriamente relacionado con uno del nivel anterior ($nivel - 1$). Por definición es una hoja del árbol de control y tiene asociado propiedades como: persistencia (Booleano), tiempo espera (segundos) y parada (Booleano). Este nivel debe mantener un límite de tiempo $t, t > 0$ (segundos) indicado por la propiedad tiempo de espera mientras el usuario administre el mismo dispositivo. Si el límite k es alcanzado, el comando es cancelado (parada=TRUE), dando por terminada la generación de la *Activity* iniciada por un comando de nivel cero e invocando a generar una nueva *Activity* o cancelando definitivamente su construcción,

hasta la lectura de un nuevo gesto de inicio (G_R).

Propiedades La propiedad *parada* definida en un nodo de cualquier nivel en la estructura permite cancelar el comando en cualquier instante. Para lograr una cancelación de comando inmediata en caso de emergencia debe considerarse un *gesto emergente* adicional, cuyas propiedades sean consultadas en todos los niveles antes de continuar el recorrido. Este *gesto emergente* podrá ser invocado en cualquier momento mientras la *Activity* este activa (TRUE)

Tiempo de espera, esta dada en segundos, pero se debe convertir internamente a un control en milisegundos de espera para dar por cancelado el comando. Esto por la naturaleza de los dispositivos a controlar. Vencido el tiempo de espera, se libera el control en el nivel donde se haya cumplido primero. Por ejemplo, si se cumplió en un gesto de nivel 2, se cancela la acción y pero se mantienen el control del dispositivo; si por el contrario, se cumplió en un gesto de nivel 1, se cancela la *Activity* permitiendo la lectura de una nueva orden.

Persistencia, es una propiedad del G_A que permite su continuidad (una misma orden es enviada varias veces), hasta que haya finalizado el tiempo de espera o se haya cancelado la *Activity*. Depende de la funcionalidad, pues por ejemplo, si lo que se desea es subir el volumen del televisor esta orden tendrá “persistencia” mientras el usuario mantenga el G_A o hasta que se venza el tiempo si el G_A no es mantenido.

Otras propiedades como: *Estado*, permite conocer el estado del dispositivo si el protocolo lo permite. *Comunicación*, determina el protocolo y sus componentes mediante el cual se realizara el envío de datos (comandos) al dispositivo, entre otros que se pudiesen requerir.

Restricciones

- Un G_R no puede ser usado como *gesto emergente*.
- Un G_D o un G_A pueden ser usados solo una vez por cada nivel del GCT. Consecuentemente, se puede volver a usar un gesto pero en un contexto (nivel) diferente.
- Es posible usar un gesto de descanso para dar mayor robustez al sistema de control, pero deberá evaluarse el retardo (delay) que ello provocará en el sistema de reconocimiento y control. El delay máximo admitido por el sistema es de 300 ms debido a restricciones de tiempo real [7] y por la naturales de la captura gestual [64, 38].

5.3.3. Sintaxis de la Actividad

La *Activity* o tripleta de comandos se implementa mediante el GCT, que se muestra en la Figura 5.5. Las definiciones necesarias han sido realizadas en las Secciones 5.3.1 y 5.3.2. El GCT debe simplificar la construcción de una *Activity*, minimizando significativamente los gestos a reconocer en cada nivel del GCT. Esto como ya mencionamos anteriormente, tendría la gran ventaja de optimizar la librería gestual, garantizar un menor tiempo de reconocimiento e introducir un menor delay al sistema. A pesar de ello, la principal fortaleza es que puede ser aplicado al control de múltiples dispositivos al usar comandos gestuales complejos.

La definición de una *Activity*, crea un grafo dirigido simple, bipartido y completo, creando por cada *Activity* un subgrafo conexo al atravesar los tres niveles del GCT a los cuales hemos denominado: nivel de inicio (Nivel 0), nivel de dispositivo (Nivel 1) y nivel de acción (Nivel 2). Esto nos permite relacionar cada vértice, formando un grafo bipartido completo, como se muestra la Figura ???. Cada *Activity* logra definir un grafo conexo para lograr formar un comando complejo, construido desde el nivel de inicio hasta el nivel de acción.

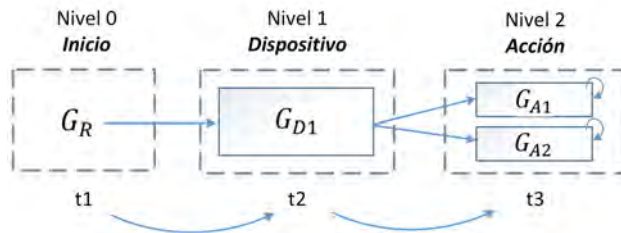


Figura 5.6: La *Activity* y componentes por nivel.

Una vez formada la *Activity* el sistema de control sobre el cual se haya implementado debe enviar el comando resultante al dispositivo (eléctrico o electrónico) específico, el cual lo ejecutará. La ejecución persistente de un comando también es posible dada la definición de bucle, logrando mantener un gesto de acción (G_A) para ese dispositivo además, de las restricciones antes indicadas.

Un gesto de nivel 1 (G_D) integra en sus propiedades las funcionalidades intrínsecas del dispositivo al cual se desea controlar. Un G_A puede ser asignado a una funcionalidad, de aquellas posibles, determinadas en un G_D . Cada G_A logra una funcionalidad según se haya definido, por ejemplo, en un sistema de aire acondicionado un G_A puede controlar el incremento de temperatura, mientras otro G_A podría encargarse de bajar el nivel térmico, y otro G_A , podría encargarse de fijar la temperatura deseada.

5.4. Materiales y métodos

En este capítulo hemos planteado dos aspectos básicos. La confortabilidad del gestos y la convalidación de la propuesta de Árbol de Control Gestual mediante un caso de estudio. Por ello, esta sección agrupa estos dos aspectos cuyos resultados podrían sugerir la validez de nuestra propuesta de protocolo de control gestual jerárquico.

5.4.1. Dispositivo de captura gestual

Como dispositivo de captura gestual (DCG) hemos usado a la pulsera gestual MYO la cual incluye ocho electrodos sEMG que se ubican radialmente en el antebrazo sujetos por una banda flexible. El dispositivo fue estudiado en mayor detalle en la Sección 3.3 del Capítulo 3, de esta tesis.

5.4.2. Biblioteca gestual

La biblioteca gestual que se presenta en la Figura 5.7, será sometida a un análisis de confortabilidad del usuario y servirá para implementar un caso de estudio para lograr la implementación del Árbol de Control Gestual. La biblioteca fue usada en el Capítulo 3 Sección 3.6.1 y plantea un total de 13 gestos estáticos, los cuales la conforman.

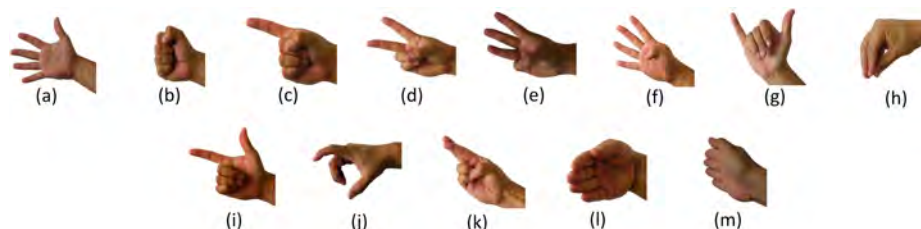


Figura 5.7: Biblioteca gestual propuesta para análisis de confortabilidad.

5.4.3. Participantes

Los participantes que a continuación se detallan participaron, tanto en las pruebas de confort del gesto, cuanto en las pruebas de aplicabilidad y usabilidad del concepto de árbol de control gestual. Los participantes fueron veinticinco voluntarios (6 mujeres y 19 hombres) en edades comprendidas entre 20 y 50 años ($Media = 34,24$, $DS = 9,54$). Todos los usuarios llenaron un formulario de consentimiento informado de la toma de datos a partir de las bio-señales producidas y su posterior uso

con fines de investigación. Todos los usuarios conocían el manejo del ordenador y tenían experiencia en el uso de dispositivos móviles, en especial Smartphone. Tres voluntarios eran zurdos. La circunferencia de sus antebrazos alcanzó una media de 27,5 ($DS = 2,01$), hombres y mujeres incluidos. Estos datos son necesarios y complementarios a nuestro estudio, pues se usará a MYO como Dispositivo de Captura Gestual (DCG) el cual tiene un rango de ajuste máximo de 34 cm y un mínimo de 19 cm.

Tabla 5.1: Características de los voluntarios participantes en esta investigación

Voluntario	Edad	Sexo	Mano	Peso	Estatura	LA	CA
No.	Años	M/F	D/S	Kgs	m	cm	cm
1	40	M	D	91,7	1,71	47	31
2	31	M	D	71,4	1,69	49	25
3	28	M	D	65,9	1,72	45	27
4	24	M	Z	69	1,76	48	31
5	39	M	D	86,7	1,63	46	28
6	23	M	Z	72,5	1,75	45	26
7	39	F	D	72,3	1,49	42	26
8	36	M	D	98,4	1,81	49	29
9	24	F	D	60,1	1,7	43	24
10	23	F	D	73,5	1,76	45	25
11	25	M	D	90,1	1,73	49	29
12	21	M	D	86	1,84	50	26,5
13	24	M	D	85,4	1,76	51	30
14	28	M	D	84	1,69	46,5	29
15	28	M	D	86,4	1,73	50	27
16	30	F	D	62,6	1,56	43	24
17	38	M	D	96,4	1,69	45	28
18	50	M	D	94,6	1,58	42	28
19	39	M	D	82,6	1,65	43	29
20	35	M	D	86	1,73	47	26
21	38	F	Z	89,3	1,69	49	29
22	43	M	D	95,3	1,66	45	28
23	48	M	D	92,3	1,67	42	28
24	38	M	D	86,4	1,72	43	29
25	38	F	D	68,3	1,62	42	25

LA: Longitud del antebrazo, CA: Circunferencia del antebrazo, Z: Zurdo, D: Derecha, M: Masculino, F: Femenino.

La técnica de calibración expuesta en el Capítulo 3 Sección 3.6.3 fue empleada.

5.4.4. Protocolo experimental

Diseño del GCT

Como Dispositivo de Captura gestual (DCG) se usó al mismo dispositivo descrito en la Sección 5.4.1, y como Dispositivo Controlador (DC) un sistema empotrado basado en Raspberry Pi 3. El empotrado recibía los datos del DCG por medio del protocolo Bluetooth (BLE) y procesaba la información usando un algoritmo similar al implementado en el Capítulo 3 de esta investigación. El proceso de clasificación estaba basado en SVM (Support Vector Machines).

Hemos tomado en consideración para esta implementación un subconjunto de nueve gestos de la biblioteca gestual inicialmente propuesta en la Sección 5.4.2. Con este catálogo implementaremos nuestro Sistema de Control basado en un Árbol de Control Gestual (SC-GCT). Los gestos se describen en la Figura 5.7 y según el estudio de *confortabilidad del gesto* realizado en la Sección ?? cumplen criterios de comodidad deseados. Todos los gestos son reproducibles y de fácil memorización como lo sugiere [128]. Los gestos considerados más cómodos (gestos (a) y (b) en Figura 5.7) han asignados a comandos más frecuentes y de constante repetición. Los demás gestos han sido asignados a comandos de Acción (G_A) y de dispositivo (G_D).

Formalmente, el control gestual mediante GCT involucra a tres comandos principales G_R, G_D, G_A , pero es necesario tener en consideración un comando especial (de parada o STOP) adicional y externo a la *Activity*. Un comando STOP es necesario para casos de emergencia o error, en los cuales es necesario una *parada* inmediata del comando y por lo tanto la cancelación inmediata de la *Activity* en proceso. La Figura 5.8 presenta la biblioteca gestual usada y el GCT implementado, en este caso de estudio.

Se planeó el control de tres dispositivos de casa (HDs), a saber: sistema de TV, audio (HIFI) y sistema de aire acondicionado (HVAC). La clasificación de los gestos de control y los detalles de funcionalidad asignada a nuestro prototipo se describen en la Tabla 5.2.

(G_R) Gesto (a), es aquel comando gestual que inicia la *Activity*.

(G_D) Gestos (d), (e), y (g) son aquellos comandos gestuales que identifican a los dispositivos en los cuales se ejecutarán los comandos de Acción.

(G_A) Gestos (h), (i), (f), y (m) son aquellos que activan/desactivan una funcionalidad específica en el dispositivo de acuerdo con la Tabla 5.2.

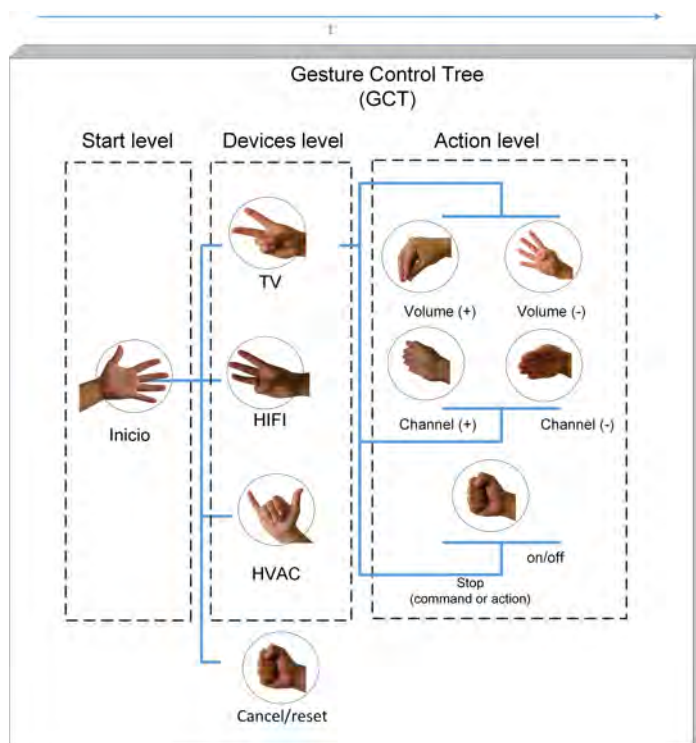


Figura 5.8: GCT implementado en el caso de estudio.

Una vez construida la *Activity*, el SC-ACG envía dicho comando al HD indicado en donde, se ejecutará el comando de Acción correspondiente. El uso reiterativo de un comando de acción es posible gracias a la *persistencia del comando* (P_{CMD}), siempre que el usuario continúe administrando el mismo dispositivo, como por ejemplo, en el caso de subir el nivel de volumen. La funcionalidad determina, en mayor o menor grado, las alternativas de trabajo que un dispositivo HD posee. La funcionalidad es accesible por el usuario o un sistema y su complejidad varía conforme de acuerdo al tipo de dispositivo y fabricante.

El modelo arquitectónico propuesto en el Capítulo 3 Sección 3.7.2 fue usado en la implementación del prototipo. Brevemente podemos indicar que el prototipo se compone de tres elementos principales:

- MYO como dispositivo de captura gestual (HGD).
- Un empotrado que implementa el algoritmo de clasificación y reconocimiento gestual propuesto en el Capítulo 3 al cual hemos llamado controlador HGC.
- Un conjunto de dispositivos de casa (HD) que usan múltiples protocolos de

Tabla 5.2: Comandos gestuales y su descripción

Gesto	Descripción	Tipo
(a)	Inicia la Activity	I
(b)	Detiene o cancela la ejecución de una Activity	D,A
(i)	Cambia la frecuencia, canal o modo en orden ascendente. El incremento es,secuencial dependiendo del dispositivo. Es una respuesta afirmativa en,caso de requerirla el Sistema.	A
(m)	Cambia la frecuencia, canal o modo en orden descendente. El decremento es secuencial y depende del dispositivo. También es una respuesta negativa a una pregunta del dispositivo.	A
(h)	Incrementa el nivel, intensidad de una funcionalidad tal como: volumen, temperatura, luz, apertura, etc	A
(f)	Decrementa el nivel o intensidad de una funcionalidad. Similar al anterior gesto (h).	A
(d), (e), (g)	Permiten identificar a un dispositivo	D

comunicación (WiFi, Bluetooth e iR). La Figura 5.9 muestra al prototipo SC-ACG controlando un sistema de TV.

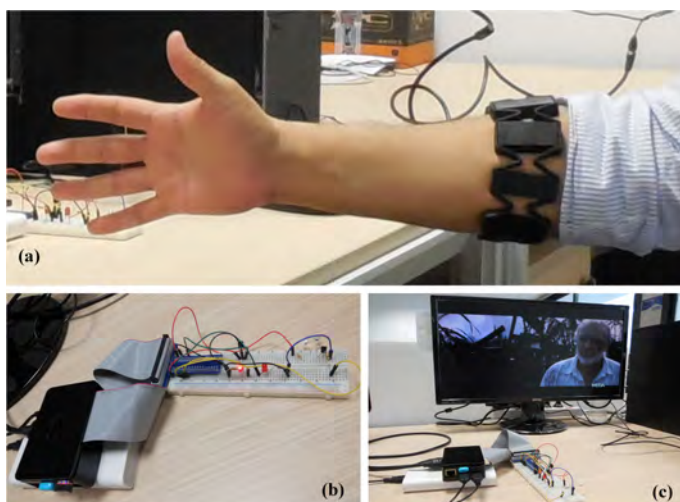


Figura 5.9: Imágenes del prototipo en operación.

La construcción de una *Activity* empieza cuando el gesto de inicio es detectado. A continuación, el controlador asigna el comando correspondiente. Si el comando gestual forma parte de una *Activity*, el sistema espera el siguiente comando gestual y continúa su construcción. Una vez construida la *Activity* (Inicio, Dispositivo, Ac-

ción), el sistema la ejecuta, seleccionando el dispositivo y enviando el comando de acción (funcionalidad) correspondiente, usando el protocolo de comunicación más adecuado (WiFi, Bluetooth, iR, etc), definido también en el sistema. El procedimiento de conformación de una *Activity*, trata de evitar los falsos positivos, pues el G_R asignado (gesto (a) en nuestro caso) es el único que puede iniciar la escucha y por lo tanto la *Activity*, ignorando de esta manera cualquier otro gesto detectado.

Las propiedades de un comando gestual, que conforma el GCT, indican como ha de formarse una *Activity* válida, pudiendo inclusive, no lograr formar la *Activity* por las restricciones que debe cumplir. En este sentido, una *Activity* puede ser cancelada o revocada por el usuario en cualquier momento, usando el comando gestual de parada o STOP y ,por el sistema, cuando supera el tiempo de espera definido en cada nivel. Las propiedades y restricciones, garantizan el flujo adecuado del comando, cumpliendo ciclos temporales internos y externos. La Figura 5.10 trata de mostrar este concepto para mayor claridad.

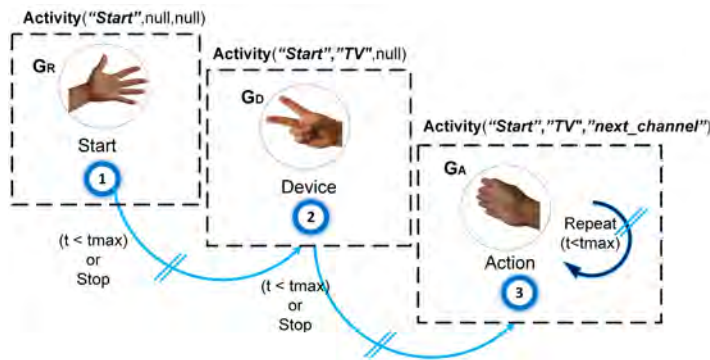


Figura 5.10: Flujo del comando complejo para formación de una *Activity* usando el GCT.

El usuario verifica la ejecución de una *Activity* cuando el dispositivo doméstico ejecuta la acción solicitada. Sin embargo, esto no es suficiente para que el usuario sepa que se inició y ejecutó un comando. En este prototipo, hemos implementado un sistema de notificación para el usuario basado en la activación de luces (en el empotrado) y modos de vibración (en la pulsera gestual).

El sistema de notificación por luces constaba de dos leds, uno para indicar el inicio de una *Activity* (rojo) y otro para admitir el comando completo (amarillo). Los dos led parpadeaban (encendían y apagaban) cuando se enviaba el comando para ser ejecutado una vez concluida una *Activity* (rojo y amarillo producen un parpadeo al mismo tiempo por tres veces).

Por otra parte, el sistema de vibración consistía en un conjunto de pulsos vi-

bratorios, que el usuario podía sentir en su antebrazo pues sucedían por activación del sensor vibratorio de la pulsera gestual. Este sistema logra independizar el sistema empotrado de la vista del usuario. La vibración podía ser larga (inicio de la *Activity*), media (admisión del dispositivo) y corta (admisión y envío del comando al HD). En dispositivos con protocolo iR se implementó un sistema de admisión de código infrarrojo, almacenamiento y asignación de funcionalidad. Este circuito es adicional al prototipo y permite ampliar la funcionalidad para nuevos dispositivos que usen este protocolo mediante el cual el SC-ACG selecciona el código iR de la funcionalidad y lo envía, como parte de la *Activity* construida por el usuario.

Test de confortabilidad

Todos los voluntarios debían ejecutar los gestos contemplados en el catálogo que se muestra en la Figura 5.7. El lugar de pruebas consistía en una gran mesa de trabajo con un sillón cómodo donde el voluntario se sentaba y realizaba cada uno de los gestos. La posición del brazo con respecto al antebrazo trataba de formar un ángulo recto. Cada gesto tenía un tiempo total de prueba de 100 s, dividido en intervalo de 10 s para formar el gesto y 10 s de descanso con cinco repeticiones para cada gesto. Entre cada repetición se estableció un estado de transición o descanso, al cual se debía retornar antes y después de formar el gesto. El estado de transición consistía en una posición relajada de la mano sin realizar gesto alguno y con igual duración temporal que la asignada para formar el gesto. El inicio del experimento estaba dado por el investigador y los intervalos de tiempo eran calculados por software mediante una aplicación realizada para el efecto. El software controlaba los intervalos de tiempo y emitía una señal auditiva tanto al inicio como al final de cada prueba. La correcta formación del gesto era controlada también por software, pues la misma aplicación integraba el software de reconocimiento desarrollado en el Capítulo 3. Para la toma de datos se usó un dispositivo móvil consistente en una Tableta ASUS con SO Android 5.1. Además, se mostraba en pantalla el gesto a ser realizado y/o en ejecución y el valor del temporizador en cuenta regresiva, para cada prueba.

En las pruebas de aplicabilidad y usabilidad del GCT, se plantearon dos posible escenarios de control de un dispositivo, tomando en consideración posiciones anatómicas diferentes. Terminada la prueba, el usuario debía llenar un cuestionario de preguntas en el cual se evaluaba a cada gesto en su nivel de confort en la escala creada para el efecto y expuesta en la Sección 5.1.1 de este capítulo. Este procedimiento se repetía con cada voluntario.

Test del GCT

Tal y como se indicó en la Sección 5.4.3 se planearon dos posibles escenarios aplicados al control de dispositivos de casa. El primero con el usuario cómodamente sentado y el segundo, con el usuario recostado en un cómodo sillón de pruebas. En ambos escenarios se usa el SC-GCT implementado en la Sección 5.4.4. Cada usuario tenía un tiempo de 10 s para lograr construir el mayor número de *Activity* para controlar un dispositivo de su elección como se muestra en la Figura 5.8. Cinco repeticiones fueron permitidas a cada usuario. Las pruebas estaban orientadas a medir el grado de exactitud en la formación de la *Activity* en las posiciones indicadas y además, la facilidad de uso (usabilidad) del sistema que implementa el GCT.

Las pruebas fueron realizadas usando la técnica de “doble ciego”, en la cual los sujetos ignoran si los gestos son reconocidos o no por el sistema. Pero, si podían observar el resultado de la ejecución de la *Activity* en el dispositivo o guiarse por las ayudas implementadas, tanto visuales (luces) como de vibración en el antebrazo usando la pulsera. Por ejemplo una *Activity*(MANO-ABIERTA, DOS, PALMA-FUERA), aplicada a un televisor, lograba cambiar de canal de TV al siguiente (ascendente).

El comando STOP asignado al gesto (b) (mano cerrada), podía ser invocado en cualquier momento y provocaba la anulación de la *Activity*, lo cual era contabilizado como una acción fallida o incompleta. Al caducar el tiempo máximo de espera ($k = 3s$), por las restricciones de tiempo impuestas en el diseño, la *Activity* era cancelada y contabilizada como un intento fallido. Los gestos o posturas inexistentes o mal realizadas no eran tomados en cuenta.

El software de pruebas, guardaba los datos por cada *Activity* exitosa y por aquellas canceladas o no completadas. Un total de ciento cinco (105) pruebas fueron realizadas. Cada gesto podía ser logrado en un tiempo aproximado entre 0,3 y 0,6 segundos y el número de posturas logrado, lo imponía el usuario por su habilidad de formar el gesto y generar la *Activity*. El set de datos obtenido fue procesado usando el software estadístico R.

5.5. Resultados

Test de confortabilidad

A continuación mostramos los resultados obtenidos del Test de Confortabilidad. En primer lugar se mostrarán el nivel de confort alcanzado por cada gesto y

posteriormente, los resultados totales de la biblioteca gestual y su grado de confortabilidad alcanzado.

Por gesto En la Figura 5.11 es posible apreciar el comportamiento individual del gesto, calculado en base a la definición realizada en la Sección 5.1.1, en relación al grado o nivel de confort dado por el voluntario. Por ejemplo, podemos observar que el Voluntario 1 calificó al gesto (a) y (b) con un grado 4 (cómodo), similar a la asignación realizada por el Voluntario 2. Sin embargo, al calificar a los demás gestos, sus resultados varían.

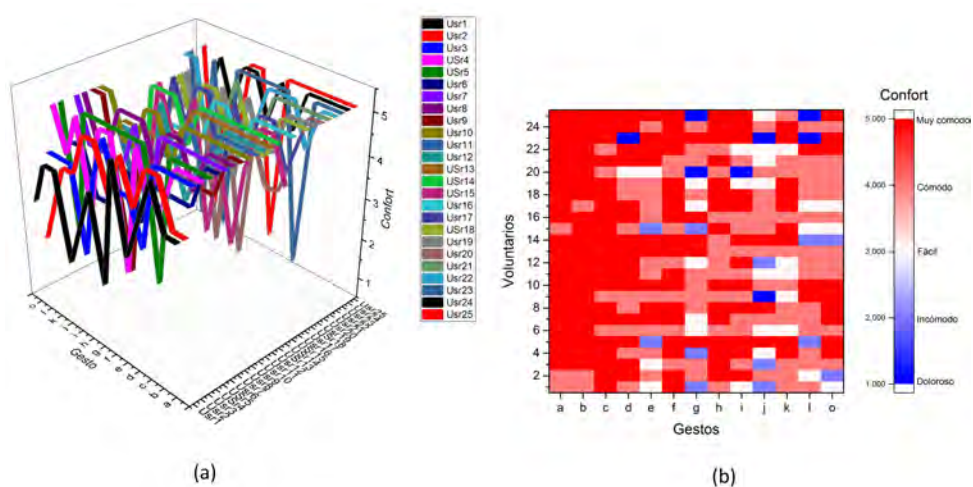


Figura 5.11: Nivel de confort individual del gesto.

La Figura 5.11.a nos presenta una idea general del confort del gesto criterio del usuario en cuanto a la confortabilidad del gesto. Cada gesto ha sido valorado conforme la escala indicada, pudiéndose apreciar que unos gestos son más confortables que otros. La Figura 5.11.b en un mapa de calor, nos indica en forma aún más clara la valoración en confort dada por los voluntarios. En ella se puede apreciar que los gestos (g) y (j) han sido calificados como dolorosos por varios voluntarios. Sin embargo, también es notorio que el resto del catálogo gestual ha sido señalado como muy comfortable (color rojo intenso) y es mayoría en el mapa de calor diseñado.

Para lograr una mejor apreciación del gesto en su conjunto así como para lograr determinar la confortabilidad de la biblioteca gestual, es necesario aplicar la ecuación 5.2. La Figura 5.12.a, nos presenta una imagen 3D de la valoración general de los voluntarios en la escala de confort implementada. La mayor concentración de esferas están en el nivel 5 (muy comfortable), seguida del nivel 4 (confortable). La

Figura 5.12.b, nos muestra el mismo escenario pero calculado en porcentaje.

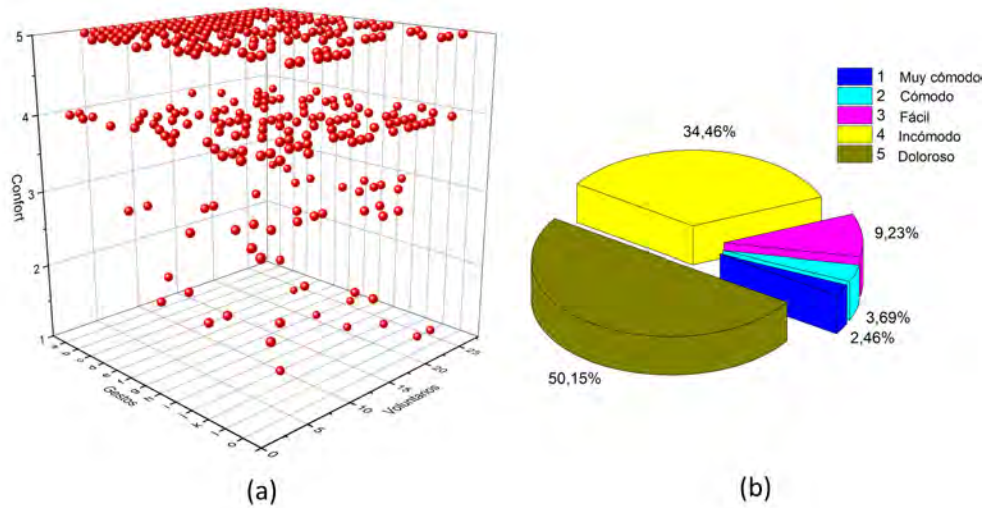


Figura 5.12: Nivel de confort de la biblioteca gestual.

Por biblioteca Conforme a la Figura 5.12.b, podemos decir que el 50,15% de los gestos de la biblioteca gestual son considerados muy confortables. El 34,46% es confortable. Los dos porcentajes sumados nos dice que el 84,61% de los gestos son gestos que cumplen con criterios de confortabilidad para los usuarios, y por lo tanto, válidos para ser usados en interacción gestual.

Un 9,23% de los gestos, aunque no llegan a ser confortables, si alcanzar a ser *fáciles* para el voluntario. Mientras, el 6,15% de todos los gestos del catálogo fueron considerados en escalas inferiores (incómodos y dolorosos).

Test del GCT

Durante los experimentos, los usuarios pudieron recordar los gestos para control del dispositivo al conformar la *Activity*, lo cual en muchos casos ayudó a mejorar su rendimiento. Esto sugiere además, que un lenguaje estándar de control podría beneficiar, en mayor grado, el manejo de un dispositivo de casa. El sistema SC-ACG respondió adecuadamente sin que los usuarios reportasen problema alguno en su funcionamiento. Además, los usuarios, no presentaron molestias por el uso del dispositivo de captura gestual o uso repetitivo del gesto. Los gestos usados fueron adecuadamente reconocidos para formar las tripletas de comando que constituyen la *Activity* en forma adecuada y oportuna. El uso del método de calibración por

defecto para cada participante, indicado en el Capítulo 3 Sección 3.6.3, ayudó a evitar lecturas erróneas por este motivo.

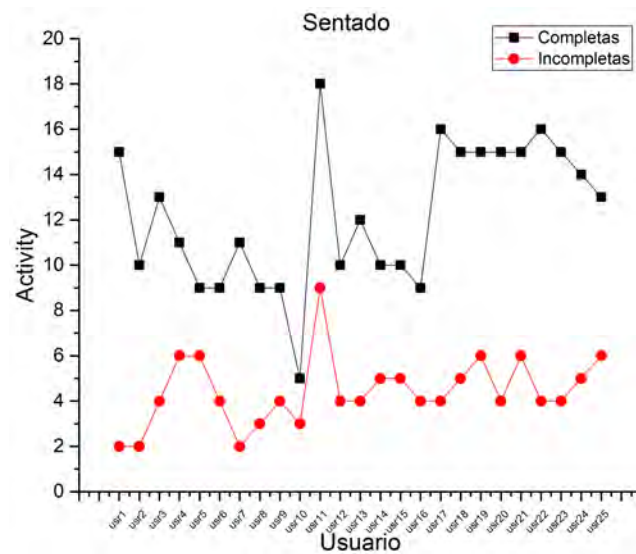


Figura 5.13: *Activity* por usuario en posición sentado.

En posición sentado los intentos que lograron una *Activity* completa alcanzaron una media 12.16 ($DS = 3,16p - value0,11038$). Mientras que los intentos fallidos alcanzaron una media de 4.44 ($SD = 1,56$).

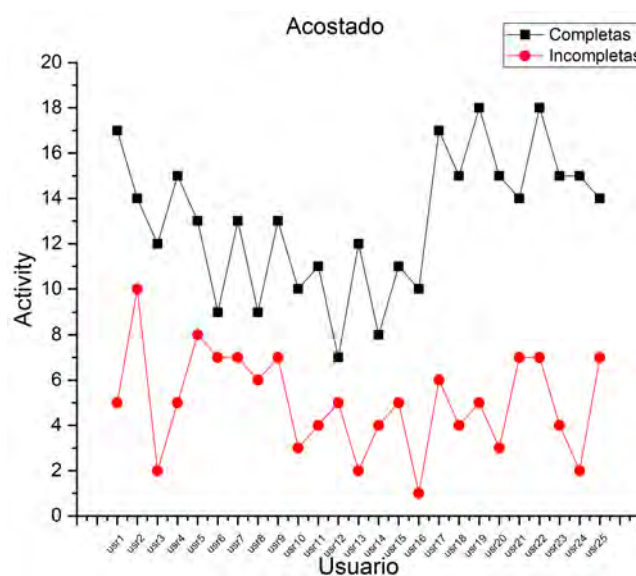


Figura 5.14: *Activity* por usuario en posición acostado.

En posición acostado, los intentos que lograron una *Activity* completa alcanzaron una media de 13 ($DS = 3,07p - value0,5348$) y 5.04 ($DS = 2,18$) para las *Activity* incompletas o canceladas. En ambos casos las *Activity* incompletas o canceladas no registran una distribución normal.

La Figura 5.15 muestra la *AR* alcanzada por cada usuario luego de aplicar la ecuación 5.3 para las dos posiciones. La (\bar{AR}), calculada como la *AR* media en posición sentado, obtuvo una media de 73% ($DS = 0,07$), con un valor mínimo de 60% y un máximo de 88%. En posición acostado la (\bar{AR}), alcanzó una media de 72% ($DS = 0,09$), con un valor mínimo de 56% y un máximo de 91%.

Es notoria la escasa incidencia de la posición escogida por el usuario al formar la *Activity*, al usar el SC-ACG. Pero, es evidente también que la exactitud presenta incrementos notorios en posición acostado. En ningún momento la media de la exactitud baja de 56% pero su mediana logra un 73%, lo cual es muy aceptable para esta nueva forma de control.

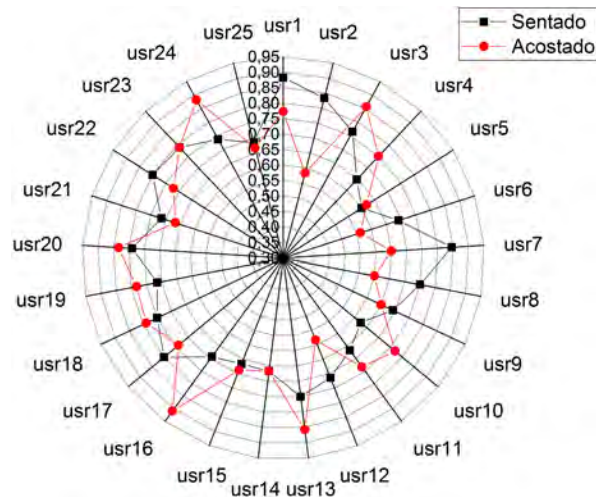


Figura 5.15: Exactitud *ACC* de los usuarios en las posiciones analizadas.

El control de un dispositivo HD usando la *Activity* puede ser logrado tanto de pie como acostado, sin existir mayor variación significativa por la posición adoptada por el voluntario. Durante nuestro experimento, la comunicación con los dispositivos HD fue lograda usando protocolo iR (infrarrojo) pero, en una aplicación real, dependería de la forma de comunicación con el dispositivo (WiFi, Bluetooth, e iR).

El circuito de comunicación iR implementado en este experimento, tenía capacidad de aprender nuevos códigos iR de un dispositivo HD, lo cual convertía a nuestro prototipo SC-GCT en un control remoto universal, pero gestionado en for-

ma gestual. Bajo estos términos, el sistema SC-GCT estaba en capacidad de enviar el comando en forma directa al HD usando el protocolo propietario del dispositivo indicado. Los códigos iR de varios fabricantes están disponibles en varios sitios web y pueden ser agregados para su uso al sistema en caso necesario. El registro de códigos iR fue realizado una sola vez y se mantenía almacenada en un archivo de códigos iR del sistema.

5.5.1. Validando el GCT

Con el fin de poder determinar la validez con concepto de *Árbol de Control Gestual (GCT)* desde el punto de vista del usuario, cada usuario participó en una encuesta para determinar la usabilidad del concepto. En dicha encuesta participaron todos los voluntarios que usaron que participaron en los test anteriores. Los datos de los usuarios se describen en la Sección 5.4.3. Todos ellos con previa experiencia en el Test de confortabilidad y en el Test del GCT descritos en la Sección 5.4.

Las primeras preguntas estaban orientadas a la facilidad de uso (preguntas 1 a 4) y las siguientes (preguntas 5 a 7) a satisfacción subjetiva del uso del ACG implementado. Estas preguntas se muestran en la Tabla 5.3 junto a las respuestas resumidas de los usuarios en la Figura 5.16.

Tabla 5.3: Cuestionario usado para validar el concepto GCT

No.	Preguntas
P1	¿Le ha parecido fácil el aprender a usar el ACG para lograr una funcionalidad determinada en un dispositivo de casa?
P2	¿Ha sido sencillo construir una orden o comando por niveles al aplicar el ACG?
P3	¿Fue fácil el uso del gesto al usarlo en la interacción con SC-ACG y aplicarlo al control de dispositivos?
P4	¿Le resultó fácil de comprender el hecho de usar el mismo gesto en diferentes dispositivos con funcionalidades similares?
P5	¿Considera Ud. que un sistema basado en ACG le puede dar mayor posibilidad de control al ser usado en forma de control universal sobre los dispositivos de casa?
P6	¿Le resultó útil la ayuda mediante luces y vibración para lograr formar y enviar un comando a un dispositivo de casa?
P7	¿Implementaría un sistema SC-ACG en su domicilio, como una ayuda en la administración y control de dispositivos de casa, basado en gestos?

De acuerdo a las respuestas los usuarios podemos decir que: (P1) Al 64% de los usuarios le pareció fácil aprender el uso del GCT hasta alcanzar la funcionalidad

y destreza deseada; (P2) El 80 % dice que considera sencillo lograr un comando recorriendo los niveles del GCT; (P3) El 72 % indica que el uso del gesto en la interacción para controlar un dispositivo de casa es fácil; (P4) El 68 % de los usuarios considera que el gesto usado en varios dispositivos es comprensible al tener funcionalidades similares (g.e., subir volumen con aumento temperatura); (P5) El 88 % manifiesta que el uso del GCT como control universal de dispositivos sería una buena opción para control de múltiples dispositivos de casa; (P6) El 96 % respondió que la ayuda implementada fue útil y oportuna en la navegación y aplicación del GCT y garantiza su aceptación y uso en aquel sistema que lo implemente; (P7) El 100 % coincide en que la implementación de un sistema SC-GCT para administración y control de dispositivos de casa sería una buena opción.

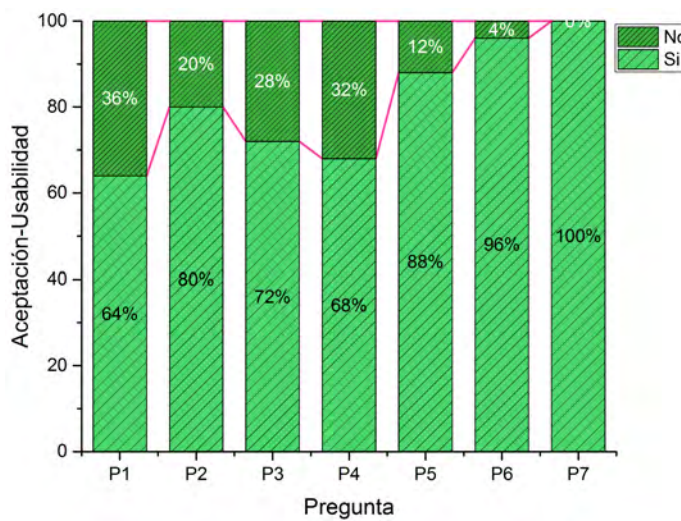


Figura 5.16: Respuestas de los usuarios en usabilidad.

5.6. Conclusiones

En este Capítulo hemos expuesto el diseño e implementación de un Sistema de Control Gestual usando el GCT (DC-GCT). El GCT está diseñado para lograr la formación de comandos complejos llamados (*Activity*). Un prototipo fue diseñado para demostrar la validez de los conceptos y funcionalidad en la integración del gesto al control de un dispositivo. El sistema SC-GCT fue implementado usando una interfaz gestual que identifica el gesto a través de las señales EMG, generadas al formar el gesto de control. Dos posiciones (sentado y acostado) fueron usadas para evaluar la exactitud de formación de la *Activity* usando el GCT. Los

resultados indican que la *Activity* alcanzó una exactitud promedio ($\bar{AR} = 0,73$) en posición sentado y 0,72 en posición acostado. Sin embargo, la exactitud por usuario alcanzó hasta un 0,88 en posición sentado y hasta 0,91 en posición acostado.

En el diseño y construcción del prototipo se usó el modelo arquitectónico propuesto en Capítulo 3 Sección 3.7.2, demostrándose su validez. Además, la construcción de un comando complejo formado por una tripleta de comandos simples (INICIO, DISPOSITIVO, ACCIÓN) es posible gracias a la *Activity* y su aplicación posible en el control de dispositivos eléctricos y electrónicos.

Una de las grandes ventajas observadas en el uso del GCT es, permitir la optimización del gesto al reutilizarlo en otro u otros dispositivo con funcionalidad similar o diferente (por ejemplo, cambiar de canal en un sistema de TV con cambiar de frecuencia en sistema de Audio). Esta gran ventaja, impulsa la idea de que un lenguaje gestual de control unificado es posible y a la cual los fabricantes deberían tener en cuenta. Además, esto permitiría al usuario usar el gesto en forma estándar, como sucede actualmente con otros gestos, en infinidad de dispositivos móviles. En nuestro experimento hemos usado al SC-ACG como un sistema de control remoto universal basado en gestos de la mano que es capaz de gestionar, al mismo tiempo, varios dispositivos del hogar con una misma librería gestual. Esto inclusive, en aquellos dispositivos que usan diferentes protocolos de comunicación (iR, Bluetooth, WiFi).

El sistema SC-ACG desarrollado fue muy intuitivo, y demostró además ser funcional y adaptable. La evaluación del GCT realizada por los voluntarios participantes en estas pruebas, nos permite expresar que un sistema de control basado en el concepto de Árbol de Control Gestual (SC-ACG) es una buena opción para gestión de dispositivos heterogéneos, basado en una misma librería gestual. Todo lo anteriormente expuesto abre nuevas perspectivas en el uso e implementación de sistemas de control que implementen interfaces naturales de usuario basadas en gestos. Interfaces de este tipo, pueden hacer que la HCI sea más intuitiva y personalizada. También podemos decir que, el uso de dispositivos wearables que implementan comunicación inalámbrica (por ejemplo, MYO), además de sensores (sEMG, IMU, EEC, etc) y facilidades de adaptación para desarrollo e investigación pueden lograr una gran transformación en la forma de interacción entre el humano y la máquina.

Capítulo 6

Conclusiones y trabajos futuros

En este capítulo presentamos un resumen general de los logros alcanzados, resaltando los resultados más relevantes que se han obtenido y las conclusiones principales a las que hemos llegado con esta tesis. Por otro lado, presentamos las publicaciones que hemos realizado a lo largo de esta investigación y los trabajos futuros que consideramos de relevancia para dar continuidad a la misma.

Conclusiones

Como se ha venido comentando a lo largo de esta memoria, la investigación realizada ha sido centrada en el análisis, diseño y evaluación de sistemas basados en el uso de Interfaces Naturales de Usuario de tipo Gestual (INU-G) orientados al control y supervisión, teniendo en consideración diferentes ámbitos reales de aplicación en los cuales hemos probado su comportamiento y eficacia. Bajo este contexto, los objetivos planteados inicialmente en esta tesis fueron los siguientes:

- Establecer los componentes gestuales en la interfaz, su clasificación y métodos de captura y reconocimiento gestual.
- Analizar los métodos de captura y reconocimiento del gesto de la mano usando dispositivos basados en imagen.
- Investigar la captura y reconocimiento del gesto basado en sensores.
- Estudiar la captura y reconocimiento del gesto basado en superficies táctiles (TUI) y establecer parámetros de comparación y evaluación con interfaces GBUI.

- Establecer los principios de diseño de un lenguaje complejo de comandos usando el gesto. Como parte de esta etapa de la investigación se ha logrado determinar la confortabilidad del gesto, principal componente de las interfaces GBUI.

Como consecuencia del trabajo realizado, se describen a continuación los resultados y conclusiones que obtuvimos a partir del cumplimiento de cada uno de estos objetivos:

- En primer lugar pudimos determinar los componentes básicos del gesto y su interfaz al realizar un análisis detallado y profundo de sus características, fundamentos y aplicaciones potenciales en la interacción Humano-Máquina. Por otra parte, dado que el reconocimiento gestual es fundamental para este tipo de interfaces, hemos estudiado las formas más populares de captura gestual y que al mismo tiempo han merecido importancia por su amplia aplicación y excelentes resultados. Hemos observado que el uso de Técnicas de Machine learning para el procesamiento de información es compatible totalmente con el reconocimiento gestual y sus resultados, por ello hacemos un enfoque pormenorizado de las técnicas, conceptos y sus principios de aplicación.

Específicamente, la aplicación del interfaces gestuales integradas en aquellas formas de interacción natural de usuario a la resolución de problemas de control y supervisión de dispositivos. Con este análisis logramos cumplir el objetivo 1.

- Para lograr el objetivo 2 y poder validar el uso de técnicas de dispositivos basados en medios ópticos propusimos un análisis de la interfaz gestual basándonos en el reconocimiento óptico. Dicha forma de reconocimiento pensada para trabajo con gestos de la mano tanto estáticos (simples) como dinámicos (complejos). En ambos casos, planteamos una biblioteca gestual propia para su evaluación y uso. Se pudo determinar las variables que afectan directamente al reconocimiento en dispositivos ópticos y en base a ello, proponemos métricas orientadas a optimizar el uso del gesto y a lograr determinar aquellos gestos de la mano considerados óptimos para operaciones de control definiendo conceptos de efectividad y tasa de reconocimiento gestual.

Este estudio además, posibilitó desarrollar dos estudios científicos promocionando nuestros resultados. El primero en el cual se hace una aplicación directa de las métricas planteadas en exactitud y precisión para lograr el concepto

de efectividad aplicada al gesto de control. Y la segunda, una aplicación de dichos conceptos sobre una plataforma robótica de entrenamiento muy conocida en medios educativos cual es la plataforma LEGO Mindstorms EV3. Los estudios han sido publicados. Se pudo determinar además, que el gesto puede ser orientados en dos aspectos específicas de aplicación: el GtoC, o gesto orientado a comando y GtoT o gesto orientado al seguimiento o reproducibilidad (tracking). En este objetivo, planteamos GtoC enfocado sobre el control reactivo en interfaces de control de dispositivos robóticos.

- Para cumplir el objetivo 3, se desarrolló e implementó un sistema de clasificación y reconocimiento gestual basado en señales EMG y aplicado sobre un sistema embebido, en este caso aplicando técnicas de Machine Learning concretamente un algoritmos de clasificación gestual basado en Support Vector Machines (SVM). Para lograrlo, dos experimentos de diferente naturaleza fueron realizados, a saber off-line y on-line. El primero, off-line logró establecer el algoritmo de clasificación que mejor se ajustaba a la identificación de patrones usando señales EMG de los dos implementados a saber k-Nearest Neighbor (k-NN) y SVM. Fue necesario la aplicación de extracción de características basadas en el tiempo para lograr una alta exactitud. El segundo, on-line determinó que un sistema de reconocimiento gestual puede lograr gran versatilidad al lograr un adecuado reconocimiento gestual que permita el uso de protocolos inalámbricos de comunicación e incorporen facilidades de portabilidad y comunicación al usar sistemas embebidos para su ejecución. Para probar su validez, se utilizó una biblioteca gestual amplia y compleja, que produjo óptimos resultados al ser aplicada en un sistema de interacción y control gestual. En este estudio participaron 21 voluntarios para lograr un amplio dataset que le dio mayor robustez al clasificador y a los sistemas de reconocimiento y control implementados.

Este estudio, al igual que el anterior, logró sentar las bases para elaborar dos trabajos de investigación que se hallan publicados. El primero, en el cual se plantea la interacción gestual aplicada al control de dispositivos de eléctricos y electrónicos. Y el segundo, que expone los resultados de análisis y comparación entre los algoritmos k-NN y SVM.

- El objetivo 4, fue alcanzado realizando un extenso análisis del impacto en el usuario al interactuar directamente con su Smartphone utilizando una interfaz basada en gestos (GBUI), en contraste con la tradicional TUI (Touchs-

creen User Interface). En este análisis participaron 25 personas y fueron consideradas dos posiciones anatómicas de uso, del pie o “normalz acostado o “especial”. Las dos posiciones fueron evaluadas en exactitud. Se desarrollaron varias aplicaciones software, tanto para captura de datos como para evaluación de resultados, principalmente compatibles con Android OS. Los resultados fueron validados a través de una aplicación Android llamada Sistema Interactivo de Control de llamadas de Emergencia (SICLLE), la cual se encargaba de gestionar las llamadas desde y hacia el Smartphone del usuario. Esta aplicación y forma de interacción logró excelentes resultados, los mismo que fueron extendidos a dos documentos científicos publicados.

Lo anterior, nos permite decir que la interfaz basada en gestos logró muy buenos resultados, independientemente de la posición anatómica del usuario, que si bien no compiten con la gran exactitud de una interfaz TUI, si nos indica que es una muy buena alternativa de interacción basada en el usuario y con grandes perspectivas de aplicación.

- Finalmente, el objetivo 5 fue alcanzado a través de dos estudios: el primero fundamentado en el análisis de la confortabilidad del gesto para el usuario; y, el segundo, basado en un nuevo concepto al cual hemos llamado Árbol de Control Gestual (Gestural Control Tree, GCT). El análisis de confort logró establecer los gestos más confortables y fáciles de realizar, a criterio del usuario, deducidos a partir de una extensa librería gestual estudiada. El GCT logró demostrar que la implementación de comando complejos, a partir de la atomicidad del gesto es posible. Es decir, lograr un control más amplio, preciso y adecuado de varios dispositivos, a partir de una biblioteca misma gestual. Este nuevo concepto, optimiza el uso del gesto de control y logra una mejor cobertura de control. Además, nos llevó a la conclusión de que un lenguaje gestual unificado lograría un impacto real en la interacción y control de dispositivos de casa o de múltiple naturaleza. La propuesta de GCT y su aplicación ha sido llevada a un manuscrito que actualmente esta en proceso de revisión.

Publicaciones derivadas de la investigación

Durante el desarrollo de esta tesis se obtuvieron las siguientes publicaciones:

- Publicaciones en Revistas Internacionales de Alto Impacto JCR:

-
1. G. Pomboza-Junez, Juan A. Holgado, Nuria Medina-Medina, *Towards the gestural interface: Comparative Analysis between Touch User Interfaces vs Gesture-based User Interfaces on Mobile Devices*, Universal Access in the Information Society, Exploring New Natural User Experiences, 2017. En prensa.
 2. G. Pomboza-Junez, Juan A. Holgado, *Analysis of surface EMG signals on a wearable armband device based on Support Vector Machines and k-Nearest Neighbor algorithms for Hand Gestures Interfaces*, International Journal of Sensors, 2017. En revisión.
- Publicaciones en Congresos Internacionales:
 1. G. Pomboza-Junez, Juan A. Holgado, *Control of Home Devices based on Hand Gestures*, 5a. IEEE International Conference on Consumer Electronics - ICCE2015, ISBN:978-1-4799-8748-1, Berlin, Alemania, 6 al 9 de septiembre de 2015.
 2. G. Pomboza-Junez, Juan A. Holgado, *Hand Gesture Recognition based on sEMG signals using Support Vector Machines*, 6a. IEEE International Conference on Consumer Electronics - ICCE2016, ISBN:978-1-50920-2096-6, Berlin, Alemania, 5 al 7 de septiembre de 2016.
 3. G. Pomboza-Junez, Juan A. Holgado, *Gestural Interaction with Mobile Devices: An analysis based on the user's posture*, XVII Proceedings of the XVII International Conference on Human Computer Interaction-Interacción 2016, ISBN:978-1-4503-4119-6, Salamanca, Spain, 13 al 16 de septiembre de 2016.
 - Publicaciones en Congresos Nacionales:
 1. G. Pomboza-Junez, Juan A. Holgado, *Control de dispositivos domóticos mediante interfaces hombre-máquina basadas en gestos de una mano*, VI Edición de las Jornadas de Computación Empotrada JCE 2015, ISBN: 978-84-16017-54-6, Córdoba, España, 23 al 25 de septiembre de 2015.
 2. G. Pomboza-Junez, Juan A. Holgado, *Sistema de Arquitectura Abierta: el Gesto Aplicado al Control Robótico*, VII Edición de las Jornadas de Computación Empotrada SARTECO 2016, ISBN:978-84-9012-626-4, Salamanca, España, 14 al 16 de septiembre de 2016.
 3. G. Pomboza-Junez, Juan A. Holgado, *Interacción Gestual en Dispositivos Móviles: Análisis basado en la posición del usuario*, XVII Con-

greso Internacional de Interacción Persona-Ordenador-Interacción 2016, ISBN:978-84-9012-629-5, Salamanca, España, 13 al 16 de septiembre de 2016.

- Publicaciones como co-autor en investigaciones conexas:

1. José A. Jiménez-Montanes, G. Pomboza-Junez, Juan A. Holgado, *El gesto aplicado al control de dispositivos en tiempo real usando un guante de interacción de bajo costo*, II Jornadas de Computación Empotrada y Reconfigurable SARTECO 2017, ISBN:978-1-4503-4119-6, Málaga, España, 19 al 22 de septiembre de 2017.

- Otras publicaciones en preparación:

1. G. Pomboza-Junez, Juan A. Holgado, *El control de dispositivos heterogéneos mediante una librería gestual unificada: Aplicación del Árbol de Control Gestual*, International Journal of Sensors, 2017.

Trabajos futuros

En esta última sección nos centramos en describir las nuevas ideas que han surgido a partir de esta investigación. En este sentido, podemos vislumbrar trabajos potenciales de marcada relevancia que en un futuro cercano se podrían llevar a cabo. Estos trabajos se pueden enmarcar dentro de tres líneas principales de investigación, las cuales se describen a continuación.

1. Interacción Humano-Máquina.

- Aplicar el gesto en dispositivos robóticos industriales y verificar su exactitud usando conceptos de GCT. Este trabajo lograría validar a una interfaz de control industrial en ambientes operativos reales y pondría la base para aplicaciones orientadas a la industria usando el gesto de las manos.
- Ampliar la biblioteca gestual para lograr un conjunto más amplio de interacción.
- Explorar en el uso de electrodos EMG en el rostro, para lograr un control gestual facial.
- Explorar el uso combinado de sensores EMG y dispositivos ópticos para establecer interfaces basadas en gesto más robustas.

2. Soporte al usuario en salud.

- Implementar soluciones de ayuda al usuario con discapacidad visual. Para este proyecto se pueden usar los resultados obtenidos en este trabajo de investigación.
- Usar los sensores EMG como dispositivos de control de salud en una persona. Un proyecto en este sentido lograría identificar el cansancio muscular o ayuda en el diagnóstico de problemas musculares.
- Proponer el uso de sensores EMG para control de conductores, estableciendo parámetros de respuesta muscular para determinar el cansancio físico. Este proyecto es muy interesante pues mediante él se podría lograr establecer el cansancio físico por conducción y poner alarmas que indique el nivel de cansancio para que sea posible o no la conducción.

3. Diseño de soluciones hardware y software para control de dispositivos

- Realizar un análisis, diseño y construcción de un guante de datos que incorpore sensores EMG en combinación de Unidades Inerciales. Un trabajo como el planteado, lograría imponer nuevos conceptos de aplicación de una interfaz gestual aplicada a cualquier dispositivo que permita conexión
- Mejorar y ampliar el uso de SICLEE para que pueda ser descargado de PlayStore de Google como una aplicación de ayuda al usuario en momentos de emergencia.
- Implementar un sistema de reconocimiento gestual usando sensores EMG en sistemas empujados de bajos recursos, como Arduino.

Apéndice A

Señales EMG

En este apéndice se presentan capturas de señales EMG que por su amplitud y detalles, no han sido ubicados en el cuerpo principal de la tesis. Su estudio es complementario para dar mayor fundamento a los hallazgos encontrados.

Señal Raw EMG

La Figura [A.1](#) muestra el dataset total recopilado de los 21 voluntarios que participaron en el experimento del Capítulo [3](#).

Señales EMG con características de rectificación e integración aplicadas

La Figura [A.2](#) muestra el dataset en los diferentes estados de análisis al aplicar las características. En ella se puede observar, el Raw, la señal aplicada una rectificación de onda comple, señal EMG integrada y la media de ensamble en la señal.

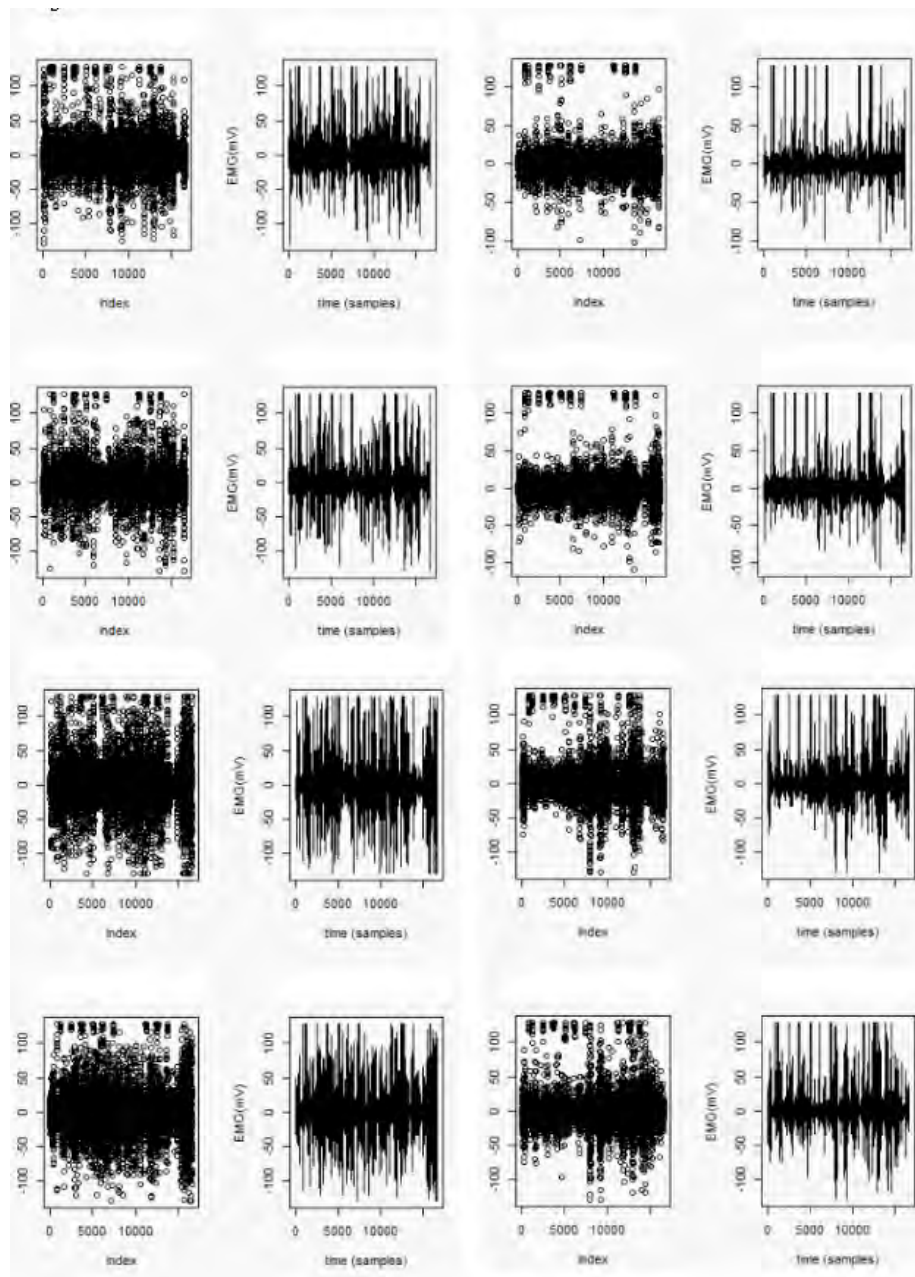


Figura A.1: La totalidad del dataset de todos los gestos vistos por distribución y lineal representados.

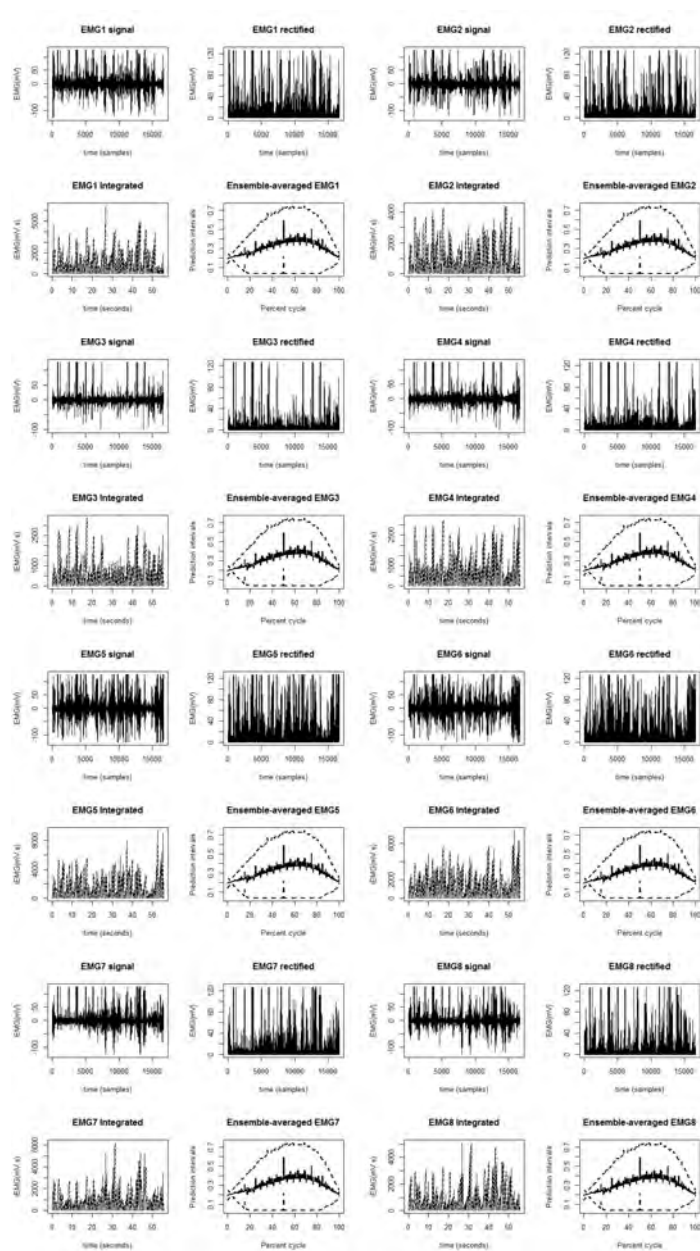


Figura A.2: Señales EMG con características de rectificación e integración.

Apéndice B

Análisis y conceptos complementarios

En este apéndice se describen varios aspectos que son parte de la investigación expuesta en esta tesis, pero que por su generalidad o mayor detalle han sido ubicados como agregados que deben ser revisados si es necesario.

Análisis del wearable MYO

Como parte de esta tesis, hemos analizado el brazalete gestual MYO desde varios puntos de vista:

- El primero (Figura B.1.a), el comportamiento de la exactitud al aplicar el clasificador SVM y sus kernels, sobre el Raw en forma directa, es decir sin incluir la extracción de características.
- El segundo (Figura B.1.b), la variación de la exactitud al agrupar los sensores sEMG del brazalete por criterios tales como: por la geometría del brazalete, y por la ubicación de los sensores sEMG en los músculos.

La Figura B.1 nos ilustra mejor el análisis propuesto.

Variación de la exactitud solo con Raw

En este sentido podemos decir que un clasificador basado sólo en los datos brutos generados por el brazalete presenta problemas de clasificación cuando el número de gestos a identificar crece como se muestra en la Figura B.2. En ella

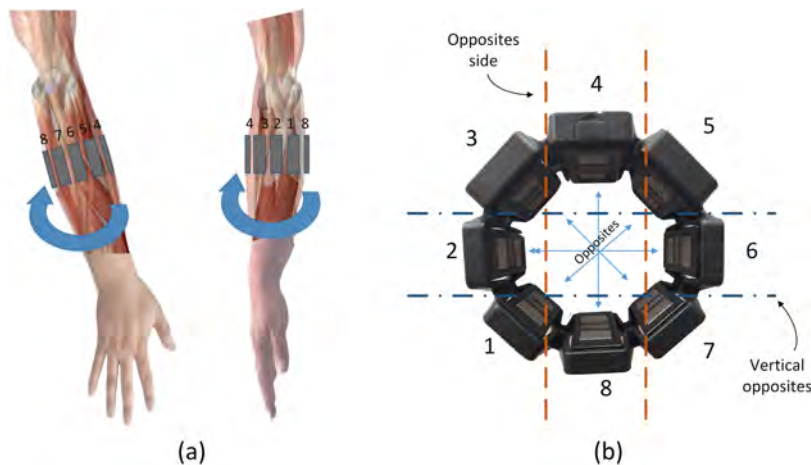


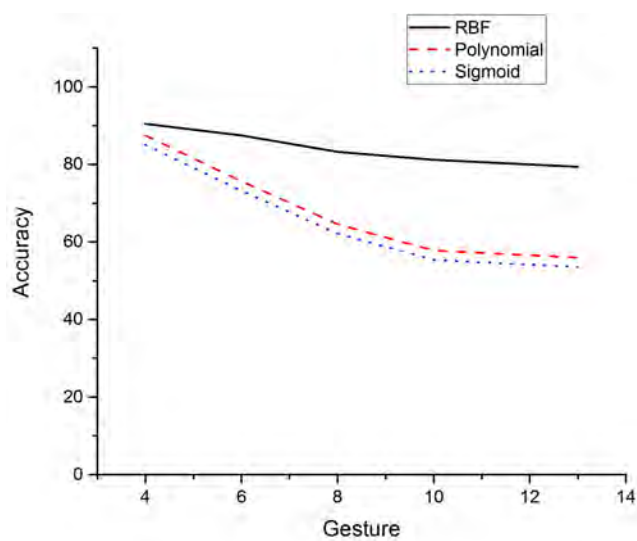
Figura B.1: Puntos de vista para análisis de MYO.

se puede observar, el nivel de exactitud del clasificador SVM usando únicamente señales de Raw EMG disminuye a medida que aumenta el número de gestos y, tiende a ser menor aún, especialmente con los núcleos Sigmoide y Polinomial. Sólo el núcleo RBF logra una exactitud, decreciente, pero superior a los anteriores. Esto se debe es probablemente a que el clasificador no puede discriminar los patrones gestuales adecuadamente, ya que las señales Raw EMG son muy similares similares, de un gesto a otro, y también puede deberse al hecho, que los sensores sEMG leen la actividad eléctrica del mismo conjunto muscular situado en el antebrazo (Figura B.1.a).

Los resultados completos del experimento se muestran en la Tabla B.1. En ella se muestra el uso de los tres tipos de kernel, detallados en la Sección 3.5.2 del Capítulo 3.

Estos resultados nos indican, que la clasificación usando únicamente el Raw proveniente de la pulsera gestual no es suficiente para lograr una adecuada clasificación del gesto. En el mejor de los casos, usando kernel RBF se alcanzó 57,84% de exactitud, resultado que aunque bueno, aún está muy por debajo de la exactitud aceptable para un sistema de reconocimiento gestual.

En contraste, este comportamiento puede ser mejorado mediante la inclusión de características (FW-REMG e IEMG) en el conjunto de datos, lo que permite al clasificador distinguir los patrones gestuales ocultos mucho mejor, como se expone en el Capítulo 3.



(a)

Figura B.2: Variación de la exactitud usando el Raw de datos sin extracción de características.

Tabla B.1: Valores alcanzados de exactitud usando kernel RBF, Polinomial y Sigmoide.

Tipo de	Parámetros			Valores obtenidos				
	Kernel	C	Cross	Gamma	ACC	Kappa	Rand	Crand
Radial	1	5	0.0625	54.24	49.80	89.29	28.41	53.67
	10	5	0.0725	53.48	53.77	90.37	33.28	57.34
	20	5	0.04	15.24	58.33	54.84	90.48	14.216
	75	5	0.0225	57.84	55.82	90.63	31.10	59.23
	100	5	0.0225	58.04	55.91	90.66	35.27	59.31
Polinomial	5	5	0.0625	14.52	10.23	67.50	2.94	17.16
	10	5	1	16.05	28.70	85.94	14.21	34.20
	100	5	0.02	17.11	98.42	64	2.44	16.80
	100	5	0.008	17.25	45.67	33.81	4.6	11.94
Sigmoide	10	5	0.08	6.52	-0.03	82.92	0.004	7.21
	10	5	0.06	5.36	-0.009	83.53	0.003	6.8

Variación de la exactitud en base a la geometría

En relación al análisis de la exactitud en base a la geometría de MYO, el objetivo es tratar de establecer una posible relación entre los sensores, su ubicación y su exactitud de clasificación. Para lograr esto, la exactitud de clasificación fue

medida usando grupos de sensores y utilizando, en cada uno de ellos, el mismo conjunto de datos, pero de acuerdo con los grupos creados.

- Por geometría del brazalete (ver Figura B.1.b), fue posible obtener ocho grupos de dos, tres y cuatro sensores sEMG. Por opuestos: grupos (4, 8), (2, 6); grupos laterales opuestos (3, 2, 1), (5, 6, 7); opuestos verticales (3, 4, 5), (1, 7, 8); complementarios opuestos (2, 4, 6, 8) y (1, 3, 5, 7).
- Por ubicación de los sensores sEMG en los músculos (ver Figura B.1.a), considerando los principales músculos que rodean al antebrazo involucrado en la realización de un gesto, fue posible formar ocho grupos de tres, cinco y siete sensores EMG respectivamente. Es necesario definir los músculos que intervienen en cada sensor: sensor 8 (flexor carpi ulnaris), sensor 7 (flexor profundo de los dedos), sensor 6 (extensor carpi ulnaris y abductor pollicis longus), sensor 5 (extensor digiti minimi y extensor digitorum), Sensor 4 (extensor carpi radialis brevis y extensor carpi longus), sensor 3 (brachioradialis y pronato teres), sensor 2 (flexor carpi radialis y digitorum superficialis) y sensor 1 (flexor digitorum superficialis). Los grupos fueron: (2, 8, 5), (3, 5, 8), (1, 4, 7), (5, 1, 8), (1, 3, 5), (3, 5, 7), (2, 1, 8, 5, 4) y (4, 3, 2, 1, 8, 7, 5).

La Figura B.3 muestra los niveles de exactitud alcanzados por cada grupo. En general, el nivel de exactitud permanece por debajo de 0,63 cuando se consideran grupos de pocos sensores EMG ($sensores \leq 3$), que están situados en un lado de la matriz o en el lado opuesto, principalmente porque no rodean el antebrazo. Entonces, la información proporcionada por un número bajo de sensores EMG, dos o tres sensores, no es suficiente para contener la actividad muscular de todos los músculos presentes en el antebrazo, y entonces, el clasificador no es capaz de discriminar las diferencias en la activación de cada músculo.

Sin embargo, cuando se considera un grupo de cuatro o más sensores alrededor del antebrazo, se mejora el nivel de exactitud ($sensores \geq 4$). En estos casos, el valor de los niveles de precisión está dentro de un rango de 0,7151 a 0,9458, logrando la mejor precisión para SVM con núcleo polinomial (0,9458). De hecho, la exactitud de la clasificación para los complementarios opuestos (1, 3, 5, 7) y (2, 4, 6, 8) son muy similares (de 0,7151 a 0,7344). Esto indica que una disposición radial de los sensores EMG alrededor del antebrazo es más importante para cubrir todos los grupos de músculos en ese lugar, y no ubicar el sensor en un lugar ajeno a la actividad de un músculo. También podemos ver que algunos grupos como (4-3-2-1-8-7-5), prácticamente recogen la mayor información de las señales EMG

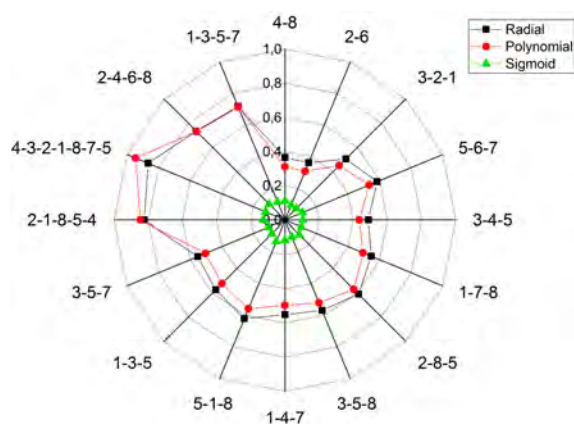


Figura B.3: Variación de la exactitud formando grupos de sensores sEMG de MYO.

proporcionando una exactitud muy alta (núcleo RBF 0,8665 y núcleo polinomial 0,9458), pero inferior a la exactitud obtenida en el Capítulo 3, considerando todos los sensores.

Esto demuestra que, un wearable con un gran número de sensores alrededor del antebrazo puede proporcionar buenos resultados para determinar la actividad muscular total, sin la necesidad de colocar sensores EMG en un lugar específico. En este caso, los ocho sensores EMG del brazalete son completamente necesarios. En consecuencia, el uso de los ocho canales (uno por sensor) en un clasificador puede discriminar mejor la diferencia entre gestos, y puede utilizarse para construir una interfaz gestual, basada en la mano del usuario, sólo fundamentada en la señal EMG. En este análisis, el núcleo sigmooidal no ha sido incluido porque tiene una precisión muy baja.

Aplicación de SVM para problemas linealmente separables

El problema del club de tenis

Como ejemplo para ilustrar la construcción de un sistema de aprendizaje se ha tomado el propuesto en [134] para una persona que suele ir los domingos a jugar a su club de tenis (que esta a dos horas de su casa). La gerencia de dicho club permite o no el juego, en función de una condición combinada de humedad y temperatura, al momento de empezar cada partido. Dicha condición no es revelada a los socios, por lo que la persona del supuesto se entera de la posibilidad de jugar o no después

de dos horas de camino. Para evitar perder 4 horas los fines de semana, la persona de nuestro ejemplo, ha tomado nota durante las últimas 13 semanas con objeto de averiguar el modelo que aplican en el club de tenis para autorizar o no el juego.

En la Tabla B.2 se muestra la información recopilada a lo largo de esas 13 semanas (fuente de datos). Las dos primeras filas son atributos que definen el dato x (vector de características) de cada día que, en este caso, tiene dos componentes. La tercera fila representa el valor asignado a cada dato, $y=+1$ si se pudo jugar e $y=-1$ en caso contrario.

Tabla B.2: Datos de entrenamiento recogidos en 13 semanas

Atributos	Temperatura	29	27	28	21	20	18	18	22	21	24	24	22	27
	Humedad	85	90	81	86	80	97	65	95	70	80	70	85	75
	Clase	-1	-1	1	1	1	-1	1	-1	1	1	1	1	1

Con estos datos de entrenamiento se puede determinar a simple vista que la decisión para predecir la posibilidad o no de jugar puede basarse en un modelo lineal, como se muestra en la Fig. B.4. La función de decisión (o frontera de separación entre clases) para determinar si se autoriza o no el juego, es una línea recta (por ejemplo la $r1$ en la Fig. B.4. Si el punto (temperatura, humedad) de un día cualquiera cae por encima de ella, entonces no se podrá desarrollar la actividad deportiva. Por el contrario, si cae por debajo, entonces se podrá jugar. Pero si analizamos el ejemplo, podremos notar que pueden existir infinitas rectas que separan a las dos clases de puntos. Para determinarlo, la solución debe estar dada por la máquina de aprendizaje.

En este caso estamos ante un problema biclase, pues son dos posibles soluciones al problema: jugar o no jugar. Utilizando los datos de entrenamiento (x,y) de la Tabla B.2, la máquina propuesta deberá construir el modelo óptimo (la recta óptima) de separación de las dos clases.

En general, cuando el valor asignado a un punto x solamente puede tomar un conjunto discreto y finito de valores (como en el ejemplo), se habla de sistema de clasificación. En este caso, cada dato pertenece a una única clase y el número de valores posibles de y determina el número de clases del problema. En cambio si los valores de y pueden ser número reales, entonces el problema de aprendizaje es un problema de regresión. En esta situación, en lugar de hablar de pertenencia de los datos a clases, se habla de pertenencia a familias de funciones.

La Fig. B.4 muestra un conjunto de puntos de entrenamiento donde cada dato

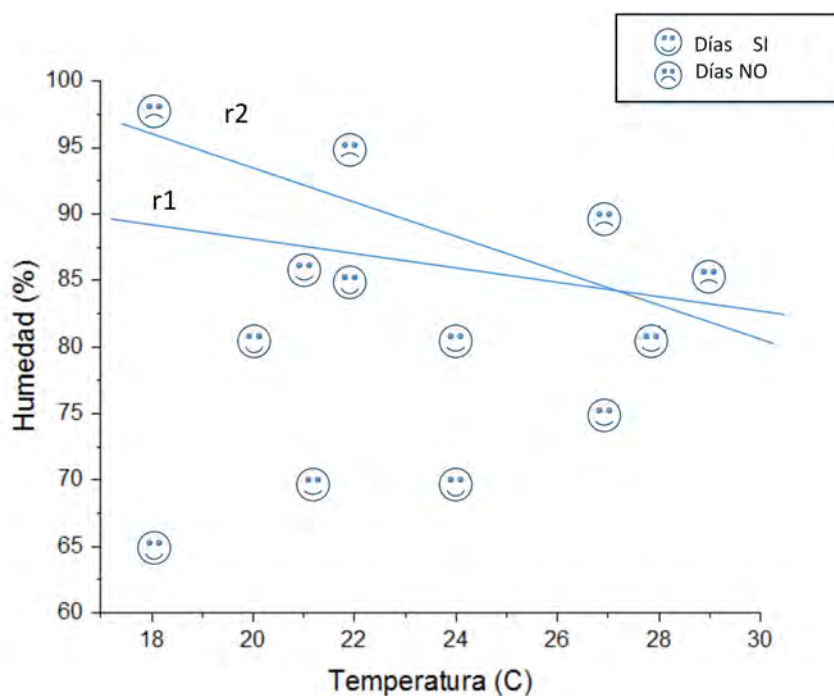


Figura B.4: La función de decisión es una línea recta.

x es un valor de la abscisa y el valor y asignado a cada abscisa es una ordenada. La máquina de aprendizaje recibirá el conjunto de pares (x,y) con objeto de estimar si pertenecen a la familia $y=\Phi_1(x,A_1,f_1)$, o a la $y=\Phi_2(x,A_2,f_2)$ o finalmente a la $y=\Phi_3(x,m,b)$. Los casos primero y segundo corresponden a una función senoidal con amplitudes y frecuencias A_1,f_1 y A_2, f_2 respectivamente. La última familia define el conjunto de rectas con pendiente angular m y ordenada en el origen b .

Conceptos básicos de Machine Learning

Dentro de Aprendizaje Automático, existen varios conceptos clave ¹ que trataremos de resumir y explicar adecuadamente a continuación.

Clasificación , trata de predecir una categoría. Un ejemplo de ello, es la forma como muchos antivirus “categorizan” direcciones IP (Internet Protocol) como “atacantes” como “legítimas”. El objetivo aquí, es generar una lista de posibles máquinas, configuradas para atacar a un sistema o robar información y, con ello, ayudar al usuario en su protección.

¹<http://cleverdata.io/conceptos-basicos-machine-learning/>

Regresión , trata de predecir un número. Un ejemplo sería el precio de un artículo en el mercado o el número de reservaciones en una aerolínea.

Aprendizaje (Learning) , es el proceso en el cual se identifican los patrones, logrando en base a ellos, realizar predicciones con nuevos datos.

Entrenamiento (Training) , es la fase en la cual se incorporan nuevos datos al sistema en espera de lograr predicciones basados en el Learning anterior.

Dataset , o también llamado conjunto de datos, es la parte primordial del sistema de predicción. Los datos, en su conjunto, son usados para entrenamiento del sistema el cual detecta los patrones. El conjunto de datos está compuesto por instancias, y las instancias, de factores, características o propiedades. Por ejemplo, en una hoja de electrónica, cada instancia sería una columna pero, pueden ser usadas indistintamente, por el investigador o de acuerdo a su contexto.

Objetivo , es el atributo o clase que queremos predecir. Es el objetivo de la predicción, realizada de acuerdo con el conjunto de datos (dataset) usado y conforme al modelo creado.

Modelo (Model) , es el resultado de la detección de patrones. Se lo puede considerar un filtro cuya entrada son los nuevos datos, y cuya salida, es la clasificación según los patrones detectados en el entrenamiento.

Árbol de decisión (Decision tree) , es la representación gráfica del modelo de predicción. Las ramas constituyen los patrones reconocidos (aprendizaje) y las hojas son las predicciones para cada patrón (objetivo o clase resultante).

Confianza (confidence) , es la probabilidad de acierto que calcula el sistema para cada predicción. En muchos casos, la confianza tiene poca diferencia con la exactitud. La confianza hace referencia al modelo en su conjunto. La exactitud, puede ser evaluada por cada objetivo o clase y difiere (en muchos casos) de la exactitud promedio del sistema de predicción.

Glosario

Fidelidad: Exactitud de una señal al ser medida en relación a la original. Indica que la captura de una señal es muy fiel o muy similar a la original.

Ruido: Es toda señal no deseada que se mezcla con una señal útil que se requiere recibir o transmitir. Es el resultado de diversos tipos de perturbaciones que tienden a enmascarar la información, cuando se presenta en la banda de frecuencias del espectro de la señal, es decir, dentro de su ancho de banda.

Distorsión: Es la diferencia entre la señal que entra a un equipo o sistema y la señal que sale del mismo. Por tanto, puede definirse como la deformación que sufre una señal tras su paso por un sistema.

Crosstalk: Se dice que entre dos circuitos existe diafonía, cuando parte de las señales presentes en uno de ellos, considerado perturbador, aparece en el otro, considerado perturbado. Se mide como la atenuación existente entre el circuito perturbador y el perturbado, por lo que también se denomina atenuación de diafonía.

Contracción isométrica: Es aquella que somete a los músculos a un movimiento contra una resistencia elevada a lo largo de un recorrido corto.

Contracción isotónica: Es aquella que somete a los músculos a un movimiento contra una resistencia baja a lo largo de un recorrido largo.

Invasivo: Procedimiento en el cual el cuerpo es “invadido” o penetrado por una aguja, una sonda, un dispositivo o un endoscopio. Lo contrario es considerado *no invasivo*.

Estocástico: Se dice de aquel proceso cuyo comportamiento es no determinista, en la medida que el subsiguiente estado del sistema está determinado tanto por las acciones predecibles del procesos como por elementos aleatorios. También se aplica a cualquier desarrollo temporal analizable en términos de probabilidad.

Wearable: Se denominan de esta manera a aquellos aparatos y dispositivos electrónicos que se incorporan (wearable significa “vestible.” “llevable”) en alguna parte del cuerpo y que interactúan en forma continua con el usuario y con otros dispositivos. Se les asigna una función concreta, por ejemplo relojes con GPS

incorporado.

Frame: Parte e una secuencia de fotogramas capturados a una determinada tasa o frecuencia de muestreo. También se le considera una imagen instantánea en un momento dado.

Sensor: Dispositivo que capta magnitudes físicas (variaciones de luz, temperatura, sonido, etc.) u otras alteraciones de su entorno.

Bibliografía

- [1] M. R. Abid, F. Shi, and E. M. Petriu. Dynamic hand gesture recognition from Bag-of-Features and local part model. In *2012 IEEE International Workshop on Haptic Audio Visual Environments and Games (HAVE 2012) Proceedings*, pages 78–82. IEEE, oct 2012.
- [2] N. AdnanIbraheem and R. Zaman Khan. Survey on Various Gesture Recognition Technologies and Techniques. *International Journal of Computer Applications*, 50(7):38–44, jul 2012.
- [3] A. H. Al-Timemy, G. Bugmann, J. Escudero, and N. Outram. Classification of Finger Movements for the Dexterous Hand Prosthesis Control With Surface Electromyography. *IEEE Journal of Biomedical and Health Informatics*, 17(3):608–618, may 2013.
- [4] S. Alavi, D. Arsenault, and A. Whitehead. Quaternion-Based Gesture Recognition Using Wireless Wearable Motion Capture Sensors. *Sensors (Basel, Switzerland)*, 16(5):605, apr 2016.
- [5] J. Alon, V. Athitsos, Q. Yuan, and S. Sclaroff. A unified framework for gesture recognition and spatiotemporal gesture segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(9):1685–1699, sep 2009.
- [6] P. K. Artemiadis and K. J. Kyriakopoulos. An EMG-Based Robot Control Scheme Robust to Time-Varying EMG Signal Features. *IEEE Transactions on Information Technology in Biomedicine*, 14(3):582–588, may 2010.
- [7] M. Asghari Oskoei and H. Hu. Myoelectric control systems. A survey. *Biomedical Signal Processing and Control*, 2(4):275–294, oct 2007.
- [8] A. Asokan, A. J. Pothan, and R. K. Vijayaraj. ARMatron ? A wearable gesture recognition glove: For control of robotic devices in disaster management

- and human Rehabilitation. In *2016 International Conference on Robotics and Automation for Humanitarian Applications (RAHA)*, pages 1–5. IEEE, dec 2016.
- [9] D. Bachmann, F. Weichert, and G. Rinkenauer. Evaluation of the Leap Motion Controller as a New Contact-Free Pointing Device. *Sensors*, 15(1):214–233, dec 2014.
- [10] F. Balagtas-Fernandez, J. Forrai, and H. Hussmann. *Evaluation of User Interface Design and Input Methods for Applications on Mobile Touch Screen Devices*, pages 243–246. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.
- [11] J. V. Basmajian and C. J. De Luca. *Muscles Alive: Their function revealed by electromyography*. N.C., 1985.
- [12] S. Benatti, F. Casamassima, B. Milosevic, E. Farella, P. Schonle, S. Fateh, T. Burger, Q. Huang, and L. Benini. A Versatile Embedded Platform for EMG Acquisition and Gesture Recognition. *IEEE Transactions on Biomedical Circuits and Systems*, 9(5):620–630, oct 2015.
- [13] S. Benatti, B. Milosevic, F. Casamassima, P. Schonle, P. Bunjaku, S. Fateh, Q. Huang, and L. Benini. EMG-based hand gesture recognition with flexible analog front end. In *2014 IEEE Biomedical Circuits and Systems Conference (BioCAS) Proceedings*, pages 57–60. IEEE, oct 2014.
- [14] M. R. Bhalla and A. V. Bhalla. Comparative Study of Various Touchscreen Technologies. *International Journal of Computer Applications*, 6(8):12–18, sep 2010.
- [15] S. Bilal, R. Akmeiliawati, M. J. E. Salami, and A. a. Shafie. Vision-based hand posture detection and recognition for Sign Language — A study. In *2011 4th International Conference on Mechatronics (ICOM)*, pages 1–6. IEEE, may 2011.
- [16] C. M. C. C. M. Bishop. *Pattern Recognition and Machine Learning*, volume 4 of *Information science and statistics*. Springer, 2006.
- [17] N. Bizzotto, A. Costanzo, L. Bizzotto, D. Regis, A. Sandri, and B. Magnan. Leap Motion Gesture Control With OsiriX in the Operating Room to Control Imaging. *Surgical Innovation*, 21(6):655–656, dec 2014.

-
- [18] M. Blazquez Cano, P. Perry, R. Ashman, and K. Waite. The influence of image interactivity upon user engagement when using mobile touch screens. *Computers in Human Behavior*, pages 1–7, mar 2017.
- [19] D. Brown, N. Renney, A. Stark, C. Nash, and T. Mitchell. Leimu: Gloveless Music Interaction Using a Wrist Mounted Leap Motion. *Proceedings of the International Conference on New Interfaces for Musical Expression*, 16:300–304, 2016.
- [20] M. A. Brown, W. Stuerzlinger, and E. J. M. Filho. The performance of uninstrumented in-air pointing. In *Proceedings of Graphics Interface 2014*, GI '14, pages 59–66, Toronto, Ont., Canada, Canada, 2014. Canadian Information Processing Society.
- [21] H. Buchner, E. Petersen, M. Eger, and P. Rostalski. Convolutional blind source separation on surface EMG signals for respiratory diagnostics and medical ventilation control. In *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 3626–3629. IEEE, aug 2016.
- [22] A. L. Buczak and E. Guven. A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection. *IEEE Communications Surveys & Tutorials*, 18(2):1153–1176, 2016.
- [23] A. Buke, F. Gaoli, W. Yongcai, S. Lei, and Y. Zhiqi. Healthcare algorithms by wearable inertial sensors: a survey. *China Communications*, 12(4):1–12, apr 2015.
- [24] V. Cevher, A. Sankaranarayanan, M. F Duarte, D. Reddy, R. G Baraniuk, and R. Chellappa. Computer Vision. *Computer Vision*, 5303(1):1–14, 2008.
- [25] C. Chen, R. Jafari, and N. Kehtarnavaz. Improving Human Action Recognition Using Fusion of Depth Camera and Inertial Sensors. *IEEE Transactions on Human-Machine Systems*, 45(1):51–61, feb 2015.
- [26] W. H. Chen, C. T. Hsieh, and T. T. Liu. A Real Time Hand Gesture Recognition System Based on DFT and SVM. *Applied Mechanics and Materials*, 284-287(June):3004–3009, jan 2013.
- [27] X. Chen and Z. J. Wang. Pattern recognition of number gestures based on a wireless surface EMG system. *Biomedical Signal Processing and Control*, 8(2):184–192, mar 2013.

-
- [28] L. Chittaro. Visualizing Information on Mobile Devices. *Computer*, 39(3):40–45, mar 2006.
- [29] C.-H. Chuan, E. Regina, and C. Guardino. American Sign Language Recognition Using Leap Motion Sensor. In *2014 13th International Conference on Machine Learning and Applications*, pages 541–544. IEEE, dec 2014.
- [30] E. A. Clancy and N. Hogan. Probability density of the surface electromyogram and its relation to amplitude detectors. *IEEE Transactions on Biomedical Engineering*, 46(6):730–739, 1999.
- [31] M. C. Corballis. From mouth to hand: Gesture, speech, and the evolution of right-handedness. *Behavioral and Brain Sciences*, 26(02):199–208, apr 2003.
- [32] J. Corde Lane, C. Carignan, B. Sullivan, D. Akin, T. Hunt, and R. Cohen. Effects of time delay on telerobotic control of neutral buoyancy vehicles. In *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, volume 3, pages 2874–2879. IEEE, 2002.
- [33] F. Corno and F. Razzak. SAT based enforcement of domotic effects in smart environments. *Journal of Ambient Intelligence and Humanized Computing*, 5(4):565–579, 2013.
- [34] T. Cover and P. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27, jan 1967.
- [35] L. Cromwell, E. A. Usselman, L. B. Weibell, and J. Fred. *Instrumentación y medidas biomédicas*. Marcombo,, 1980.
- [36] Y. Cui and J. Weng. Hand sign recognition from intensity image sequences with complex backgrounds. In *Proceedings of the Second International Conference on Automatic Face and Gesture Recognition*, pages 259–264. IEEE Comput. Soc. Press, 1996.
- [37] P. A. Daneri. *PLC Automatización y Control Industrial*. Editorial Hispuno Americana HASA, Paraguay, 1 edition, 2008.
- [38] C. J. de Luca. Physiology and Mathematics of Myoelectric Signals. *IEEE Transactions on Biomedical Engineering*, BME-26(6):313–325, 1979.
- [39] C. J. de Luca. Physiology and Mathematics of Myoelectric Signals. *IEEE Transactions on Biomedical Engineering*, BME-26(6):313–325, 1979.

-
- [40] D. de Waard, F. Westerhuis, D. Joling, S. Weiland, R. Stadtbäumer, and L. Kaltofen. Visual map and instruction-based bicycle navigation: a comparison of effects on behaviour. *Ergonomics*, 0139(April):1–14, jan 2017.
- [41] S. Devine, K. Rafferty, and S. Ferguson. Real time robotic arm control using hand gestures with multiple end effectors. In *2016 UKACC 11th International Conference on Control (CONTROL)*, pages 1–5. IEEE, aug 2016.
- [42] I.-j. Ding, C.-w. Chang, and C.-j. He. A kinect-based gesture command control method for human action imitations of humanoid robots. In *2014 International Conference on Fuzzy Theory and Its Applications (iFUZZY2014)*, pages 208–211. IEEE, nov 2014.
- [43] D.-L. Dinh, J. T. Kim, and T.-S. Kim. Hand Gesture Recognition and Interface via a Depth Imaging Sensor for Smart Home Appliances. *Energy Procedia*, 62(62):576–582, 2014.
- [44] L. Dipietro, A. Sabatini, and P. Dario. A Survey of Glove-Based Systems and Their Applications. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 38(4):461–482, jul 2008.
- [45] H. Eglowstein. reach out and touch your data: Three input devices, ranging from your hand it to computers. *j-Byte*, 15:283–286, 288–290, 1990.
- [46] K. Englehart and B. Hudgins. A robust, real-time control scheme for multi-function myoelectric control. *IEEE Transactions on Biomedical Engineering*, 50(7):848–854, jul 2003.
- [47] K. Englehart, B. Hudgins, P. A. Parker, and M. Stevenson. Classification of the myoelectric signal using time-frequency based representations. *Medical Engineering and Physics*, 21(6-7):431–438, 1999.
- [48] F. Erden and A. Çetin. Hand gesture based remote control system using infrared sensors and a camera. *IEEE Transactions on Consumer Electronics*, 60(4):675–680, nov 2014.
- [49] A. Erol, G. Bebis, M. Nicolescu, R. D. Boyle, and X. Twombly. Vision-based hand pose estimation: A review. *Computer Vision and Image Understanding*, 108(1-2):52–73, oct 2007.

- [50] D. Farina and R. Merletti. Comparison of algorithms for estimation of EMG variables during voluntary isometric contractions. *Journal of Electromyography and Kinesiology*, 10(5):337–349, oct 2000.
- [51] R. A. S. Fernandez, J. L. Sanchez-Lopez, C. Sampedro, H. Bavle, M. Molina, and P. Campoy. Natural user interfaces for human-drone multi-modal interaction. In *2016 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1013–1022. IEEE, jun 2016.
- [52] P. M. Fitts. the Amplitude of Movement. *Journal of Experimental Psychology*, 47:381–391, 1954.
- [53] A. Flores-Abad, O. Ma, K. Pham, and S. Ulrich. A review of space robotics technologies for on-orbit servicing. *Progress in Aerospace Sciences*, 68:1–26, jul 2014.
- [54] R. Francese, I. Passero, and G. Tortora. Wiimote and Kinect: Gestural User Interfaces add a Natural third dimension to HCI. *Proceedings of the International Working Conference on Advanced Visual Interfaces*, pages 116–123, 2012.
- [55] Y.-C. Fung. *Biomechanics*. Springer New York, New York, NY, second edition, 1993.
- [56] G. P. V. Galen, M. L. T. M. Muller, R. G. J. Meulenbroek, and A. W. A. Van Gemmert. Forearm EMG Response Activity During Motor Performance in Individuals Prone to Increased Stress Reactivity. *American journal of industrial medicine*, 419(November 2001):406–419, 2002.
- [57] E. Gokgoz and A. Subasi. Effect of multiscale PCA de-noising on EMG signal classification for diagnosis of neuromuscular disorders. *Journal of Medical Systems*, 38(4):31–38, apr 2014.
- [58] W. B. Green and J. D. Heckman. *The Clinical Measurement of Joint Motion*. American Academy of Orthopaedic Surgeons, 1994.
- [59] N. Grira, M. Crucianu, and N. Boujemaa. Unsupervised and Semi-supervised Clustering: A Brief Survey. *A Review of Machine Learning Techniques for Processing Multimedia Content, Report of the MUSCLE European Network of Excellence (6th Framework Programme)*, pages 1–12, 2004.

-
- [60] F. N. Guerrero, E. M. Spinelli, and M. A. Haberman. Analysis and Simple Circuit Design of Double Differential EMG Active Electrode. *IEEE Transactions on Biomedical Circuits and Systems*, 10(3):787–795, 2016.
- [61] J. Guna, G. Jakus, M. Pogačnik, S. Tomažič, and J. Sodnik. An Analysis of the Precision and Reliability of the Leap Motion Sensor and Its Suitability for Static and Dynamic Tracking. *Sensors*, 14(2):3702–3720, feb 2014.
- [62] Guodong Guo, S. Li, and Kapluk Chan. Face recognition by support vector machines. In *Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580)*, pages 196–201. IEEE Comput. Soc, 2000.
- [63] K. Hara, T. Umezawa, and N. Osawa. Effect of Button Size and Location When Pointing with Index Finger on Smartwatch. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 9170, pages 165–174. Springer-Verlang, 2015.
- [64] L. J. Hargrove, K. Englehart, and B. Hudgins. A comparison of surface and intramuscular myoelectric signal classification. *IEEE Transactions on Biomedical Engineering*, 54(5):847–853, 2007.
- [65] H. Hasan and S. Abdul-Kareem. Static hand gesture recognition using neural networks. *Artificial Intelligence Review*, 41(2):147–181, feb 2014.
- [66] H. Heo, E. C. Lee, K. R. Park, C. J. Kim, and M. Whang. A realistic game system using multi-modal user interfaces. *IEEE Transactions on Consumer Electronics*, 56(3):1364–1372, aug 2010.
- [67] H. J. Hermens, B. Freriks, C. Disselhorst-Klug, and G. Rau. Development of recommendations for SEMG sensors and sensor placement procedures. *Journal of Electromyography and Kinesiology*, 10(5):361–374, 2000.
- [68] S. Hooper. No Title, 2013.
- [69] C.-C. Hsieh and D.-H. Liou. Novel Haar features for real-time hand gesture recognition using SVM. *Journal of Real-Time Image Processing*, 10(2):357–370, jun 2015.
- [70] J. Huang, S. Gutta, and H. Wechsler. Detection of human faces using decision trees. In *Proceedings of the Second International Conference on Automatic*

- Face and Gesture Recognition*, pages 248–252. IEEE Comput. Soc. Press, 1996.
- [71] F. Hug. Can muscle coordination be precisely studied by surface electromyography? *Journal of Electromyography and Kinesiology*, 21(1):1–12, 2011.
- [72] R. Ibañez, Á. Soria, A. Teyseyre, and M. Campo. Easy gesture recognition for Kinect. *Advances in Engineering Software*, 76:171–180, oct 2014.
- [73] J. Indulska and P. Sutton. Location management in pervasive systems. In *ACSW Frontiers '03 Proceedings of the Australasian information security workshop conference on ACSW frontiers 2003*, volume 21, pages 143–151, 2003.
- [74] J. Ismirle, I. OBara, S. J. Swierenga, and J. E. Jackson. Touchscreen Voting Interface Design for Persons with Disabilities: Insights from Usability Evaluation of Mobile Voting Prototype. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 60(1):780–784, sep 2016.
- [75] R. R. Itkarkar, A. Nandi, and B. Mane. Contour-Based Real-Time Hand Gesture Recognition for Indian Sign Language. In L. C. Jain, H. S. Behera, J. K. Mandal, and D. P. Mohapatra, editors, *Smart Innovation, Systems and Technologies*, volume 33 of *Smart Innovation, Systems and Technologies*, pages 683–691. Springer India, New Delhi, 2017.
- [76] A. Jaimes and N. Sebe. Multimodal human computer interaction: A survey. *Computer Vision and Image Understanding*, 108(1-2):116–134, oct 2007.
- [77] J.-W. Jeong, M. K. Kim, H. Cheng, W.-H. Yeo, X. Huang, Y. Liu, Y. Zhang, Y. Huang, and J. A. Rogers. Epidermal Electronics: Capacitive Epidermal Electronics for Electrically Safe, Long-Term Electrophysiological Measurements. *Advanced Healthcare Materials*, 3(5):621–621, 2014.
- [78] Z. X. Jin, T. Plocher, and L. Kiff. Touch Screen User Interfaces for Older Adults: Button Size and Spacing. In C. Stephanidis, editor, *Universal Access in Human Computer Interaction. Coping with Diversity*, volume 4554 of *Lecture Notes in Computer Science*, pages 933–941. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.
- [79] T. Joachims. Training linear SVMs in linear time. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '06*, page 217, New York, New York, USA, 2006. ACM Press.

- [80] C. B. T. Juliana M. Fernández, Rubén C. Acevedo. Influencia de la fatiga muscular en la señal electromiográfica de músculos estimulados eléctricamente. *Revista EIA Escuela de ingeniería de antioquia, Medellín*, 7:111–119, 2007.
- [81] Y. Jung. What a smartphone is to me: understanding user values in using smartphones. *Information Systems Journal*, 24(4):299–321, jul 2014.
- [82] M. B. Kaaniche. *Gesture Recognition From Video Sequences*. PhD thesis, Université Nice Sophia Antipolis, 2009.
- [83] N. M. Kakoty, S. M. Hazarika, and J. Q. Gan. EMG Feature Set Selection Through Linear Relationship for Grasp Recognition. *Journal of Medical and Biological Engineering*, 36(6):883–890, dec 2016.
- [84] C.-Y. Kao and C.-S. Fahn. A human-machine interaction technique: Hand gesture recognition based on hidden Markov models with trajectory of hand motion. In *2011 International Conference on Advanced in Control Engineering and Information Science, CEIS 2011*, volume 15, pages 3739–3743, 2011.
- [85] M. Kaur, S. Mathur, D. Bhatia, and S. Verma. siGnum: Graphical user interface for EMG signal analysis. *Journal of Medical Engineering & Technology*, 39(1):19–25, jan 2015.
- [86] M. Kaushik and R. Jain. Gesture Based Interaction NUI: An Overview. *International Journal of Engineering Trends and Technology*, 9(12):633–636, mar 2014.
- [87] N. C. Kiliboz and U. Guukbay. A hand gesture recognition technique for human computer interaction. *Journal of Visual Communication and Image Representation*, 28:97–104, apr 2015.
- [88] H.-K. Kim, S. Lee, and K.-S. Yun. Capacitive tactile sensor array for touch screen application. *Sensors and Actuators A: Physical*, 165(1):2–7, jan 2011.
- [89] J. Kim, S. Mastnik, and E. André. EMG-based hand gesture recognition for realtime biosignal interfacing. In *Proceedings of the 13th international conference on Intelligent user interfaces - IUI '08*, volume 39, page 30, New York, New York, USA, 2008. ACM Press.
- [90] R. Kleissen, J. Buurke, J. Harlaar, and G. Zilvold. Electromyography in the biomechanical analysis of human movement and its clinical application. *Gait & posture*, 8(2):143–158, 1998.

- [91] S. B. Kotsiantis. Supervised machine learning: A review of classification techniques. *Informatica*, 31:249–268, 2007.
- [92] C. Kuhnel, T. Westermann, F. Hemmert, S. Kratz, A. Muller, and S. Moller. I'm home: Defining and evaluating a gesture set for smart-home control. *International Journal of Human-Computer Studies*, 69(11):693–704, oct 2011.
- [93] A. H. Kulkarni and S. A. Urabinahatti. Performance Comparison of Three Different Classifiers for Hci Using Hand Gestures. *International Journal of Modern Engineering Research*, 2(4):2857–2861, 2012.
- [94] M. La Polla, F. Martinelli, and D. Sgandurra. A Survey on Security for Mobile Devices. *IEEE Communications Surveys & Tutorials*, 15(1):446–471, 2013.
- [95] P. M. Langdon, J. Lazar, A. Heylighen, and H. Dong. *Reconciling usability, Accessibility and Inclusive Design*. Springer International Publishing, Cham, 2014.
- [96] O. D. Lara and M. a. Labrador. A Survey on Human Activity Recognition using Wearable Sensors. *IEEE Communications Surveys & Tutorials*, 15(3):1192–1209, jan 2013.
- [97] J. J. LaViola. 3D Gestural Interaction: The State of the Field. *ISRN Artificial Intelligence*, 2013(2):1–18, 2013.
- [98] J. J. LaViola. Context aware 3D gesture recognition for games and virtual reality. In *ACM SIGGRAPH 2015 Courses on - SIGGRAPH '15*, pages 1–61, New York, New York, USA, 2015. ACM Press.
- [99] M. Lee, Y. Hong, S. Lee, J. Won, J. Yang, S. Park, K.-T. Chang, and Y. Hong. The effects of smartphone use on upper extremity muscle activity and pain threshold. *Journal of Physical Therapy Science*, 27(6):1743–1745, 2015.
- [100] M. Leon, J. M. Gutierrez, L. Leija, and R. Munoz. EMG pattern recognition using Support Vector Machines classifier for myoelectric control purposes. In *2011 Pan American Health Care Exchanges*, pages 175–178. IEEE, mar 2011.
- [101] G. Li, Y. Li, Z. Zhang, Y. Geng, and R. Zhou. Selection of sampling rate for EMG pattern recognition based prosthesis control. In *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC'10*, pages 5058–5061, 2010.

- [102] J. Li, L. Zheng, Y. Chen, Y. Zhang, and P. Lu. A real time hand gesture recognition system based on the prior facial knowledge and SVM. *Journal of Convergence Information Technology*, 8(11):185, 2013.
- [103] Q. X. Li, P. P. K. Chan, D. Zhou, Y. Fang, H. Liu, and D. S. Yeung. Improving robustness against electrode shift of sEMG based hand gesture recognition using online semi-supervised learning. *2016 International Conference on Machine Learning and Cybernetics (ICMLC)*, pages 344–349, 2016.
- [104] C. K. Lim, Z. Luo, I.-M. Chen, and S. H. Yeo. Wearable wireless sensing system for capturing human arm motion. *Sensors and Actuators A: Physical*, 166(1):125–132, mar 2011.
- [105] Y.-C. Lin. The Relationship Between Touchscreen Sizes of Smartphones and Hand Dimensions. In *Proceedings of the 7th International Conference on Universal Access in Human-Computer Interaction: Applications and Services for Quality of Life - Volume Part III*, UAHCI'13, pages 643–650, Berlin, Heidelberg, 2013. Springer-Verlag.
- [106] G. S. Linoff and M. J. Berry. *Data mining techniques: for marketing, sales, and customer relationship management*. John Wiley & Sons, 2011.
- [107] M. M. Liu, W. Herzog, and H. H. C. M. Savelberg. Dynamic muscle force predictions from EMG: An artificial neural network approach. *Journal of Electromyography and Kinesiology*, 9(6):391–400, 1999.
- [108] Y. Liu and P. Zhang. Vision-Based Human-Computer System Using Hand Gestures. In *2009 International Conference on Computational Intelligence and Security*, volume 2, pages 529–532. IEEE, 2009.
- [109] S. Ljubic, V. Glavinic, and M. Kukec. Finger-Based Pointing Performance on Mobile Touchscreen Devices: Fitts' Law Fits. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 9175 of *Lecture Notes in Computer Science*, pages 318–329. Springer International Publishing, Cham, 2015.
- [110] G.-C. Luh, Y.-H. Ma, C.-J. Yen, and H.-A. Lin. Muscle-gesture robot hand control based on sEMG signals with wavelet transform features and neural network classifier. In *2016 International Conference on Machine Learning and Cybernetics (ICMLC)*, volume 2, pages 627–632. IEEE, jul 2016.

-
- [111] G. Marin, F. Dominio, and P. Zanuttigh. Hand gesture recognition with leap motion and kinect devices. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 1565–1569. IEEE, oct 2014.
- [112] G. Marin, F. Dominio, and P. Zanuttigh. Hand gesture recognition with jointly calibrated Leap Motion and depth sensor. *Multimedia Tools and Applications*, 75(22):14991–15015, feb 2015.
- [113] S. Markovitch and D. Rosenstein. Feature Generation Using General Constructor Functions. *Machine Learning*, 49(1):59–98, 2002.
- [114] A. Mehler, A. Lücking, and G. Abrami. WikiNect: image schemata as a basis of gestural writing for kinetic museum wikis. *Universal Access in the Information Society*, 14(3):333–349, aug 2015.
- [115] R. Menon, G. Di Caterina, H. Lakany, L. Petropoulakis, B. Conway, and J. Soraghan. Study on interaction between temporal and spatial information in classification of EMG signals in myoelectric prostheses. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 4320(c):1–1, 2017.
- [116] R. Merletti, A. Botter, A. Troiano, E. Merlo, and M. A. Minetto. Technology and instrumentation for detection and conditioning of the surface electromyographic signal: State of the art. *Clinical Biomechanics*, 24(2):122–134, feb 2009.
- [117] R. Merletti and D. F. Stegeman. European Recommendations for Surface ElectroMyoGraphic.
- [118] B. W. Min. Improvement of mobile U health services system. *Communications in Computer and Information Science*, 262 CCIS(PART 1):44–51, 2011.
- [119] L. Miranda, T. Vieira, D. Martinez, T. Lewiner, A. W. Vieira, and M. F. M. Campos. Real-Time Gesture Recognition from Depth Data through Key Poses Learning and Decision Forests. In *2012 25th SIBGRAPI Conference on Graphics, Patterns and Images*, pages 268–275. IEEE, aug 2012.
- [120] S. Mitra and T. Acharya. Gesture Recognition: A Survey. *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, 37(3):311–324, may 2007.

-
- [121] M. V. Moo, U. Cetina, and C. B. Loeza. Clasificación de documentos usando Máquinas de Vectores de Apoyo. *Abstraction and Application (UADY)*, 6(2012):40–51, 1998.
- [122] G. Murthy and R. Jadon. A review of vision based hand gestures recognition. *International Journal of Information Techonology and Knowledge Management*, 2(2):405–410, 2009.
- [123] S. M. Nadgeri, S. D. Sawarkar, and A. D. Gawande. Hand Gesture Recognition Using CAMSHIFT Algorithm. In *2010 3rd International Conference on Emerging Trends in Engineering and Technology*, pages 37–41. IEEE, nov 2010.
- [124] R. Navarro. Electromiografía. Technical Report Electromiografía, Universidad de Alcalá, 2013.
- [125] A. Newell. *Inclusive design or assistive technology*, pages 172–181. Springer London, London, 2003.
- [126] Q. Nguyen and P. Premaratne. Consumer electronics control system based on hand gesture moment invariants. *IET Computer Vision*, 1(1):35–41, mar 2007.
- [127] J. Nielsen. Noncommand user interfaces. *Communications of the ACM*, 36(4):83–99, apr 1993.
- [128] M. Nielsen, M. Störring, T. B. Moeslund, and E. Granum. A Procedure for Developing Intuitive and Ergonomic Gesture Interfaces for HCI. In A. Camurri and G. Volpe, editors, *Gesture-Based Communication in Human-Computer Interaction*, pages 409–420. Springer-Verlag, 2004.
- [129] C. L. Nikias and M. R. Raghuveer. Bispectrum estimation: a digital signal processing framework. *Proc. IEEE*, 75:869–891, 1987.
- [130] E. Ohn-Bar and M. M. Trivedi. Hand Gesture Recognition in Real Time for Automotive Interfaces: A Multimodal Vision-Based Approach and Evaluations. *IEEE Transactions on Intelligent Transportation Systems*, 15(6):2368–2377, dec 2014.
- [131] I. Oropesa, T. de Jong, P. Sánchez-González, J. Dankelman, and E. Gómez. Feasibility of tracking laparoscopic instruments in a box trainer using a Leap Motion Controller. *Measurement*, 80:115–124, feb 2016.

-
- [132] T. Page. Touchscreen and Perceived Usability : A Comparison of attitudes between Older and Younger mobile device. *I-Manager's Journal on Mobile Applications & technologies*, 3(1):1–16, 2016.
- [133] T. Page. Touchscreen and Perceived Usability : A Comparison of attitudes between Older and Younger mobile device. *I-Manager's Journal on Mobile Applications & technologies*, 3(1):1–16, 2016.
- [134] G. Pajares Martinsanz and J. M. De la Cruz Garcia. *Aprendizaje Automático: Un enfoque práctico*. RA-MA Publicaciones, 2010.
- [135] D. G. Park and H. C. Kim. Muscleman : Wireless input device for a fighting action game based on the EMG signal and acceleration of the human forearm. In *International Symposium on Neural Networks*, pages 1–5, 2011.
- [136] V. I. Pavlovic, R. Sharma, and T. S. Huang. Visual interpretation of hand gestures for human-computer interaction: a review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):677–695, 1997.
- [137] L. Peppoloni, F. Brizzi, C. A. Avizzano, and E. Ruffaldi. Immersive ROS-integrated framework for robot teleoperation. In *2015 IEEE Symposium on 3D User Interfaces (3DUI)*, pages 177–178. IEEE, mar 2015.
- [138] A. Phinyomark, P. Phukpattaranont, and C. Limsakul. Feature reduction and selection for EMG signal classification. *Expert Systems with Applications*, 39(8):7420–7431, jun 2012.
- [139] J. Pickering. Touch-sensitive screens: the technologies and their application. *International Journal of Man-Machine Studies*, 25(3):249–269, sep 1986.
- [140] E. Z. Pirani, F. Bulakiwala, M. Kagalwala, M. Kalolwala, and S. Raina. Android Based Assistive Toolkit For Alzheimer. *Procedia Computer Science*, 79:143–151, 2016.
- [141] P. Plawiak, T. Sosnicki, M. Niedzwiecki, Z. Tabor, and K. Rzecki. Hand Body Language Gesture Recognition Based on Signals From Specialized Glove and Machine Learning Algorithms. *IEEE Transactions on Industrial Informatics*, 12(3):1104–1113, jun 2016.
- [142] G. Pomboza-Junez and J. A. Holgado-Terriza. Control of home devices based on hand gestures. In *2015 IEEE 5th International Conference on Consumer Electronics - Berlin (ICCE-Berlin)*, pages 510–514. IEEE, IEEE, sep 2015.

- [143] G. Pomboza-Junez and J. A. Holgado-Terriza. Hand gesture recognition based on sEMG signals using Support Vector Machines. In *2016 IEEE 6th International Conference on Consumer Electronics - Berlin (ICCE-Berlin)*, pages 174–178. IEEE, sep 2016.
- [144] S. R. B. Prabhu, C. V. Dhasharathi, R. Prabhakaran, M. R. Kumar, S. W. Feroze, and S. Sophia. Environmental monitoring and greenhouse control by distributed sensor network. *International Journal of Advanced Networking and Applications*, 5(5):2060–2065, Mar 2014.
- [145] P. Premaratne, S. Ajaz, and M. Premaratne. Hand gesture tracking and recognition system using Lucas-Kanade algorithms for control of consumer electronics. *Neurocomputing*, 116:242–249, sep 2013.
- [146] K. Qian, J. Niu, and H. Yang. Developing a Gesture Based Remote Human-Robot Interaction System Using Kinect. *International Journal of Smart Home*, 7(4):203–208, 2013.
- [147] F. K. Quek. Eyes in the interface. *Image and Vision Computing*, 13(6):511–525, aug 1995.
- [148] M. B. I. Raez, M. S. Hussain, F. Mohd-Yasin, M. Reaz, M. S. Hussain, and F. Mohd-Yasin. Techniques of EMG signal analysis: detection, processing, classification and applications. *Biological procedures online*, 8(1):11–35, 2006.
- [149] A. Rainoldi, G. Melchiorri, and I. Caruso. A method for positioning electrodes during surface EMG recordings in lower limb muscles. *Journal of Neuroscience Methods*, 134(1):37–43, 2004.
- [150] S. S. Rautaray and A. Agrawal. Vision based hand gesture recognition for human computer interaction: a survey. *Artificial Intelligence Review*, 43(1):1–54, jan 2015.
- [151] D. Rempel, M. J. Camilleri, and D. L. Lee. The design of hand gestures for human-computer interaction: Lessons from sign language interpreters. *International Journal of Human Computer Studies*, 72(10-11):728–735, oct 2014.
- [152] A. Rezende, G. Schneider, and P. Prakash. Method and apparatus for customizing a display screen of a user interface, Apr. 5 2016. US Patent 9,304,668.

-
- [153] D. C. Rizzo. *Fundamentals of anatomy and physiology*. Cengage Learning, 2015.
- [154] C. Rocha, J. Rodriguez, D. Pineda, and V. Sánchez. Traslador prototype of hand signals a text using Kinect. *Avances, Investigación en Ingeniería*, 10(2):64–72, 2013.
- [155] H. A. Romo, J. C. Realpe, P. E. Jojoa, and U. Cauca. Surface EMG Signals Analysis and Its Applications in Hand Prosthesis Control. *Avances en Sistemas e Informática*, 4(1):127–136, 2007.
- [156] Rung-Huei Liang and Ming Ouhyoung. A real-time continuous gesture recognition system for sign language. In *Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition*, pages 558–567. IEEE Comput. Soc, 1998.
- [157] D. Saha and A. Mukherjee. Pervasive computing: a paradigm for the 21st century. *Computer*, 36(3):25–31, mar 2003.
- [158] C. Saikiran. Different paradigm for Touch-Screen technology : A Survey. *Applied Ergonomics*, 2013.
- [159] J. Sanchez. Mercado de wearables creció 25 % en 2016: IDC, aug 2017.
- [160] E. Sanchez-Nielsen, M. Hernandez-Tejera, and L. Anton-Canalís. Hand Gesture Recognition for Human- Machine Interaction. *Journal of WSCG*, 12(1-3):395–402, 2004.
- [161] C. L. Sandoval Rodriguez. Caracterización de la dinámica de los movimientos básicos de la mano a partir de la actividad electromiográfica del antebrazo. Master's thesis, Universidad Industrial de Santander, 2013.
- [162] A. Sanna, F. Lamberti, G. Paravati, and F. Manuri. A Kinect-based natural interface for quadrotor control. *Entertainment Computing*, 4(3):179–186, aug 2013.
- [163] A. Sarkar, K. A. Patel, R. K. G. Ram, and G. K. Capoor. Gesture control of drone using a motion controller. In *2016 International Conference on Industrial Informatics and Computer Systems (CIICS)*, pages 1–5. IEEE, mar 2016.

-
- [164] C. Schüldt, I. Laptev, and B. Caputo. Recognizing human actions: A local SVM approach. *Proceedings - International Conference on Pattern Recognition*, 3(September):32–36, 2004.
- [165] A. Sears. Improving touchscreen keyboards: design issues and a comparison with other devices. *Interacting with Computers*, 3(3):253–269, dec 1991.
- [166] A. Sekmen and P. Challa. Assessment of adaptive human robot interactions. *Knowledge Based Systems*, 42:49–59, apr 2013.
- [167] H. Setiawan, I. Setyawan, and S. Nugroho. Hand gesture recognition using Optimized Neural Network Shape Fitting on ARM11. In *2013 International Conference on Information Technology and Electrical Engineering (ICITEE)*, volume 1, pages 131–136, Salatiga, Indonesia, oct 2013. IEEE.
- [168] T. Shany, S. J. Redmond, M. R. Narayanan, and N. H. Lovell. Sensors-Based Wearable Systems for Monitoring of Human Movement and Falls. *IEEE Sensors Journal*, 12(3):658–670, mar 2012.
- [169] S. Shruthi, S. Kc, and K. K. S. Classification on Hand Gesture Recognition and Translation from Real Time Video using SVM-KNN. *International Journal of Applied Engineering Research ISSN*, 11(8):5414–5418, 2016.
- [170] M. Simao, P. Neto, and O. Gibaru. Natural control of an industrial robot using hand gesture recognition with neural networks. In *IECON 2016 - 42nd Annual Conference of the IEEE Industrial Electronics Society*, pages 5322–5327. IEEE, oct 2016.
- [171] S. Sofia. Environmental Monitoring and Greenhouse Control by Distributed Sensor Network. *International Journal of Advanced Networking and Applications*, 2065:2060–2065, 2014.
- [172] Y. Soo, M. Sugi, H. Yokoi, T. Arai, R. Du, and J. Ota. Simultaneous measurement of force and muscle fatigue using frequency-band wavelet analysis. *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, 2008:5045–5048, 2008.
- [173] V. Srhoj-Egekher, M. Cifrek, and V. Medved. The application of Hilbert-Huang transform in the analysis of muscle fatigue during cyclic dynamic

- contractions. *Medical and Biological Engineering and Computing*, 49(6):659–669, 2011.
- [174] A. Steinfeld, T. Fong, D. Kaber, M. Lewis, J. Scholtz, A. Schultz, and M. Goodrich. Common metrics for human-robot interaction. In *Proceeding of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction - HRI '06*, number 1 in ACM'2006, page 33, New York, New York, USA, 2006. ACM Press.
- [175] E. Stergiopoulou and N. Papamarkos. Hand gesture recognition using a neural network shape fitting technique. *Engineering Applications of Artificial Intelligence*, 22(8):1141–1158, dec 2009.
- [176] H. Stern, J. Wachs, and Y. Edan. Hand gesture vocabulary design: a multicriteria optimization. In *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No.04CH37583)*, volume 1, pages 19–23. IEEE, 2004.
- [177] H. Stern, J. Wachs, and Y. Edan. A Method for Selection of Optimal Hand Gesture Vocabularies. In *Gesture-Based Human-Computer Interaction . . .*, pages 57–68. Springer-Verlang, Berlin, 2009.
- [178] C.-J. Su, C.-Y. Chiang, and J.-Y. Huang. Kinect-enabled home-based rehabilitation system using Dynamic Time Warping and fuzzy logic. *Applied Soft Computing*, 22(November 2010):652–666, sep 2014.
- [179] J. Suarez and R. R. Murphy. Hand gesture recognition with depth images: A review. In *2012 IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication*, pages 411–417. IEEE, sep 2012.
- [180] F. Tenore, A. Ramos, A. Fahmy, S. Acharya, R. Etienne-Cummings, and N. Thakor. Decoding of Individuated Finger Movements Using Surface Electromyography. *IEEE Transactions on Biomedical Engineering*, 56(5):1427–1434, may 2009.
- [181] M. B. Trudeau, D. S. Asakawa, D. L. Jindrich, and J. T. Dennerlein. Two-handed grip on a mobile phone affords greater thumb motor performance, decreased variability, and a more extended thumb posture than a one-handed grip. *Applied Ergonomics*, 52:24–28, jan 2016.

- [182] J. Y. Tung, T. Lulic, D. a. Gonzalez, J. Tran, C. R. Dickerson, and E. a. Roy. Evaluation of a portable markerless finger position capture device: accuracy of the Leap Motion controller in healthy adults. *Physiological measurement*, 36(5):1025–35, may 2015.
- [183] M. Turk. Multimodal Human-Computer Interaction. In *Real-Time Vision for Human-Computer Interaction*, number August in Lecture Notes in Real-Time Vision, pages 269–283. Springer-Verlag, New York, 2005.
- [184] J. L. Varela-Benitez, J. O. Rivera-Delgado, J. M. Espina-Hernández, and J. M. De la Rosa-Vázquez. Electrodo capacitivo de alta sensibilidad para la detección de biopotenciales eléctricos. *Revista Mexicana de Ingeniería Biomédica*, 36(2):131–142, 2015.
- [185] M. Vuskovic and Sijiang Du. Classification of prehensile EMG patterns with simplified fuzzy ARTMAP networks. In *Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02 (Cat. No.02CH37290)*, volume 3, pages 2539–2544. IEEE, 2002.
- [186] D. Wang, Z. Xiang, and D. R. Fesenmaier. Smartphone Use in Everyday Life and Travel. *Journal of Travel Research*, 55(1):52–63, jan 2016.
- [187] J. Wang, H. Ren, W. Chen, and P. Zhang. A portable artificial robotic hand controlled by EMG signal using ANN classifier. In *2015 IEEE International Conference on Information and Automation*, pages 2709–2714. IEEE, aug 2015.
- [188] F. Weichert, D. Bachmann, B. Rudak, and D. Fisseler. Analysis of the Accuracy and Robustness of the Leap Motion Controller. *Sensors*, 13(5):6380–6393, may 2013.
- [189] J. P. Weir, A. L. McDonough, and V. J. Hill. The effects of joint angle on electromyographic indices of fatigue. *European Journal of Applied Physiology and Occupational Physiology*, 73(3-4):387–392, may 1996.
- [190] D. Wettschereck, D. W. Aha, and T. Mohri. A Review and Empirical Evaluation of Feature Weighting Methods for a Class of Lazy Learning Algorithms. *Artificial Intelligence Review*, 11(1/5):273–314, 1997.
- [191] K. Wheeler, M. Chang, and K. Knuth. Gesture-based control and EMG decomposition. *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, 36(4):503–514, jul 2006.

- [192] K. Wheeler and C. Jorgensen. Gestures as input: neuroelectric joysticks and keyboards. *IEEE Pervasive Computing*, 2(2):56–61, apr 2003.
- [193] J. R. Windmiller and J. Wang. Wearable Electrochemical Sensors and Biosensors: A Review. *Electroanalysis*, 25(1):29–46, jan 2013.
- [194] X. Wu, V. Kumar, J. Ross Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, P. S. Yu, Z.-H. Zhou, M. Steinbach, D. J. Hand, and D. Steinberg. *Top 10 algorithms in data mining*. Chapman&Hall/CRC, 2008.
- [195] X. Wu, V. Kumar, J. Ross Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, P. S. Yu, Z.-H. Zhou, M. Steinbach, D. J. Hand, and D. Steinberg. Top 10 algorithms in data mining. *Knowledge and Information Systems*, 14(1):1–37, jan 2008.
- [196] K. Yamagishi, L. Jing, and Z. Cheng. A system for controlling personal computers by hand gestures using a wireless sensor device. In *2014 IEEE International Symposium on Independent Computing (ISIC)*, pages 1–7. IEEE, dec 2014.
- [197] S. Yang, P. Premaratne, and P. Vial. Hand gesture recognition: An overview. In *2013 5th IEEE International Conference on Broadband Network & Multimedia Technology*, pages 63–69. IEEE, nov 2013.
- [198] S.-W. Yang, C.-S. Lin, S.-K. Lin, and C.-H. Lee. Design of virtual keyboard using blink control method for the severely disabled. *Computer methods and programs in biomedicine*, 111(2):410–8, aug 2013.
- [199] A. Yilmaz, O. Javed, and M. Shah. Object tracking. *ACM Computing Surveys*, 38(4):13–es, dec 2006.
- [200] E. Yörük and E. Konuko. Shape-Based Hand Recognition. *IEEE transactions on Image Processing*, 15(7):1803–1815, 2006.
- [201] M. Zecca, S. Micera, M. C. Carrozza, and P. Dario. Control of multifunctional prosthetic hands by processing the electromyographic signal. *Critical reviews in biomedical engineering*, 30(4-6):459–85, 2002.
- [202] X. Zhai, B. Jelfs, R. H. M. Chan, and C. Tin. Self-Recalibrating Surface EMG Pattern Recognition for Neuroprosthesis Control based on Convolutional Neural Network. *Under Review*, 11(July):1–11, 2017.

-
- [203] Y. Zhao and Y. Cen. *Data Mining Applications with R*. Elsevier, Miami, 2014.
- [204] C. Zhu and W. Sheng. Wearable sensor-based hand gesture and daily activity recognition for robot-assisted living. *IEEE Transactions on Systems, Man, and Cybernetics Part A:Systems and Humans*, 41(3):569–573, 2011.
- [205] T. G. Zimmerman, J. Lanier, C. Blanchard, S. Bryson, and Y. Harvill. A hand gesture interface device. *ACM SIGCHI Bulletin*, 17(SI):189–192, may 1986.