

Universidad de Granada
Departamento de Arquitectura y Tecnología de
Computadores



**Estudio Comparativo de la técnica Análisis de
Componentes Independientes “ICA” Aplicado al
Procesamiento Digital de Imágenes con Ruido**

TESIS DOCTORAL

Salua Esther Nassabay Pardo
Granada, 2006

Editor: Editorial de la Universidad de Granada
Autor: Salua Esther Nassabay Pardo
D.L.: Gr. 133 - 2007
ISBN: 978-84-338-4229-9

D. Carlos García Puntonet, Profesor Titular de Universidad, del Departamento de Arquitectura y Tecnología de Computadores de la Universidad de Granada, y **D. Rubén Martín Clemente**, Profesor Titular de Universidad, del Departamento de Teoría de la Señal y Comunicaciones de la Universidad de Sevilla,

CERTIFICAN:

Que la memoria titulada “**Estudio Comparativo de la técnica Análisis de Componentes Independientes “ICA” Aplicado al Procesamiento Digital de Imágenes con Ruido**”, ha sido realizada por **Dña. Salua Esther Nassabay Pardo** bajo la dirección en el Departamento de Arquitectura y Tecnología de Computadores de la Universidad de Granada para optar al grado de Doctor por la Universidad de Granada.

Granada, a 12 de Enero de 2007

Fdo.: Dr. Carlos García Puntonet Fdo.: Dr. Rubén Martín Clemente
Director de la Tesis Co-director de la Tesis

Universidad de Granada
Departamento de Arquitectura y Tecnología de
Computadores



Estudio Comparativo de la técnica Análisis de
Componentes Independientes “ICA” Aplicado al
Procesamiento Digital de Imágenes con Ruido

Memoria presentada por
Salua Esther Nassabay Pardo
Para optar al grado de
DOCTOR POR LA UNIVERSIDAD DE GRANADA

Fdo.: Salua Esther Nassabay Pardo

A mis padres y hermana, a Ingo

Agradecimientos

Quisiera agradecer a toda la gente que me ha ayudado durante el tiempo de escribir esta tesis, especialmente a:

- A Ingo, por permanecer a mi lado en todo momento, y ser mi guía a la hora de escribir esta disertación. “Sin tu ayuda este trabajo no habría sido posible”. Ich Danke dir für alle die Sache die ich mit Dir gelernt habe.
- A mi mamá Alejandra y a mi hermana Samia, por insistirme en la culminación de este proyecto; porque a pesar de la distancia han estado más cerca que nunca.
- A mi papá Abdul, “Me haces mucha falta”.
- A los Profesores Carlos G. Puntonet y Rubén Martín-Clemente, por darme la oportunidad de realizar este trabajo y ser guías constantes en todo momento.
- A la gente en el Departamento de Arquitectura y Tecnología de Compu-

tadores de la Universidad de Granada y especialmente: Encarni, Juanma, Paco and Luis-Javier.

- A Prof. Dr. Elmar W. Lang por abrirme las puertas en mi estadía en Alemania y ser una gran ayuda a la hora de orientarme cuando más lo necesitaba.
- A mi familia en Colombia, a todos los que ya no están y los que seguimos aquí. Por todas esas alegrías y tristezas que hemos vivido juntos.
- A mi familia en Alemania y el Líbano, porque han sido siempre un gran apoyo.
- A mis amigos y amigas.

“Son cosas chiquitas. No acaban con la pobreza, no nos sacan del subdesarrollo, no socializan los medios de producción y de cambio, no expropian las cuevas de Alí Babá. Pero quizá desencadenen la alegría de hacer, y la traduzcan en actos. Y al fin y al cabo, actuar sobre la realidad y cambiarla aunque sea un poquito, es la única manera de probar que la realidad es transformable.” (Eduardo Galeano)

Índice general

Resumen	xxvii
Introducción	xxxi
1. Introducción a ICA y BSS	1
1.1. Visión general	2
1.2. Análisis de Componentes Independientes (ICA)	5
1.2.1. Ambigüedades de ICA	10
1.3. Principios matemáticos para la valoración de ICA	11
1.3.1. Función de distribución acumulada	11
1.3.2. Función de densidad de probabilidad	12
1.3.3. Funciones de distribución y densidad de vectores alea- torios	14
1.3.4. Distribuciones y densidades conjuntas y marginales	15
1.3.5. Teorema del límite central	16
1.3.6. Minimización de la información mutua	17
1.3.7. Valoración de la máxima probabilidad	18

1.3.8.	Momentos y Cumulantes	18
1.3.9.	Métodos basados en medidas estadísticas de alto orden	20
1.4.	Medidas de no gaussianidad	21
1.5.	ICA y el pre-procesamiento previo de los datos	24
1.5.1.	Substracción de la media	25
1.5.2.	Blanqueamiento de datos	25
1.6.	ICA vs PCA	27
1.6.1.	PCA (Análisis de Componentes Principales)	27
1.6.2.	ICA (Análisis de Componentes Independientes)	30
1.6.3.	Comparación ICA y PCA	31
1.7.	Modelos de mezcla lineales	33
1.7.1.	Modelos lineal de mezcla instantáneo	33
1.7.2.	Modelos con distinto número de mezclas y de fuentes .	34
1.7.3.	Mezclas no estacionarias	36
1.7.4.	Conocimiento de las fuentes	36
1.8.	Modelos de mezcla no lineales	37
1.8.1.	El modelo de mezcla post no lineal	38
1.9.	Algoritmos BSS	39
1.9.1.	Métodos de eliminación de ruido	39
1.9.2.	Métodos basados en redes neuronales	41
1.9.3.	Métodos basados en la utilización estadística	42
1.9.4.	Métodos basados en el principio de la Máxima Entropía	44
1.10.	Medidas de prestaciones	44

<i>ÍNDICE GENERAL</i>	III
1.11. Aplicaciones	47
1.12. Conclusiones	48
2. Conexión entre SVH, ICA y el Ruido	49
2.1. Bases del procesamiento de imágenes digitales	50
2.1.1. El Sistema Visual Humano (SVH)	50
2.2. ICA e Imágenes	64
2.2.1. Relación entre ICA y las imágenes	66
2.3. ICA y el ruido	69
2.3.1. Ruido de las fuentes	70
2.3.2. Pocas fuentes de ruido	71
2.3.3. Técnicas de diagonalización para la estimación de la matriz de mezcla	72
2.3.4. Separación por medio de código “sparse”	73
2.4. Conclusiones	76
3. Algoritmos ICA	79
3.1. Breve reseña algoritmos ICA	79
3.2. Algoritmo fastICA	83
3.3. Algoritmo JADE	86
3.4. Algoritmo AMUSE	88
3.5. Algoritmo fastGEO	90
3.6. Denoising Source Separation DSS	91
3.7. Otros algoritmos ICA	94

3.8. Conclusiones	98
4. El ruido en ICA	101
4.1. Introducción general al análisis	102
4.2. Preparación de la información	104
4.2.1. Discusión de resultados	114
4.3. Evaluación de resultados	122
4.3.1. Evaluación de resultados presentados con fastICA . . .	124
4.3.2. Evaluación de resultados presentados con JADE	137
4.3.3. Evaluación de resultados presentados con AMUSE . . .	146
4.3.4. Evaluación de resultados presentados con fastGEO . .	162
4.3.5. Evaluación de resultados presentados con dss_fastica . .	174
4.4. Comportamiento de AMUSE	185
4.5. Conclusiones	187
5. Aplicación al De-noising	197
5.1. Método desarrollado	198
5.1.1. Análisis de resultados	213
5.1.2. Conclusiones	218
Conclusiones	219
Índice	223

Índice de figuras

1.1. Imágenes originales sin ninguna perturbación.	3
1.2. Mezclas obtenidas de las figuras originales.	4
1.3. Izquierda: variables aleatorias. Derecha: señales mezcladas. . .	5
1.4. Gráfica de señales gaussianas.	9
1.5. Arriba se muestran las fuentes antes de aplicar la matriz de mezcla. Abajo se muestra las mezclas obtenidas y los ejes de PCA (sólido) y de ICA (punteado).	32
2.1. Gráfica espectro electromagnético adaptado de: http://edison. upc.es/curs/llum/luz_vision/luz.html	52
2.2. Gráfica del ojo humano. Vista lateral, adaptado de: http: //webvision.med.utah.edu/spanish/anatomia.html	53
2.3. Organización de las distintas capas en la retina, adaptado de: http://webvision.med.utah.edu/spanish/vgeneral.html .	56
2.4. Respuestas de los tipos de conos (rojo, verde y azul) en la visión a color	57

2.5.	Recorrido visual de la imagen desde el ojo al cerebro.	59
2.6.	Partes de una neurona.	62
2.7.	Arriba: Imagen original cameraman 256 x 256 píxeles. Abajo: características típicas de la imagen, obtenidas aplicando ICA a bloques de 8 x 8 píxeles.	67
3.1.	Trasformación resultante de rectángulo a paralelepípedo a pro- ducirse la mezcla.	95
4.1.	Diagrama general. Esquema que introduce a la forma en que se hace el desarrollo de la evaluación en este capítulo.	103
4.2.	Imágenes originales de estructuras.	106
4.3.	Imágenes de estructuras después de la inserción de ruido gaus- siano (arriba), multiplicativo (centro) y de sal y pimienta (aba- jo).	108
4.4.	Estructura de la matriz de almacenamiento.	109
4.5.	Algoritmo fastICA. Matriz de filtros estables resultante de la muestra original. Tamaño de la matriz: 64 x 45.	110
4.6.	Algoritmo fastICA. Matriz de filtros estables resultante de la muestra original con ruido gaussiano. Tamaño de la matriz: 64 x 29.	111
4.7.	Algoritmo fastICA. Matriz de filtros estables resultante de la muestra original con ruido multiplicativo. Tamaño de la ma- triz: 64 x 28.	111

4.8. Algoritmo fastICA. Matriz de filtros estables resultante de la muestra original con ruido de sal y pimienta. Tamaño de la matriz: 64 x 69. 112

4.9. Algoritmo JADE. Matriz de mezcla **A** resultante después de la evaluación de las muestras sin ninguna perturbación (esquina superior izquierda), muestras con insercción de ruido gaussiano (esquina superior derecha), muestras con ruido multiplicativo (esquina inferior izquierda) y muestras con ruido de sal y pimienta (esquina inferior derecha. 113

4.10. Algoritmo AMUSE: Resultado de la matriz de mezcla **A** con la evaluación de las muestras sin ninguna perturbación. Imagen superior izquierda. Tamaño de la matriz: 64 x 64. Resultado de la matriz de mezcla **A** con la evaluación de las muestras original con ruido gaussiano. Imagen superior derecha. Tamaño de la matriz: 64 x 64. Resultado de la matriz de mezcla **A** con la evaluación de las muestras original con ruido multiplicativo. Imagen inferior izquierda. Tamaño de la matriz: 64 x 64. Resultado de la matriz de mezcla **A** con la evaluación de las muestras original con ruido de sal y pimienta. Imagen inferior derecha. Tamaño de la matriz: 64 x 64. 115

- 4.11. Algoritmo fastGEO. Resultado de la matriz de mezcla \mathbf{A} con la evaluación de las muestras sin ninguna perturbación. Imagen superior izquierda. Tamaño de la matriz: 64 x 64. Resultado de la matriz de mezcla \mathbf{A} con la evaluación de las muestras original con ruido gaussiano. Imagen superior derecha. Tamaño de la matriz: 64 x 64. Resultado de la matriz de mezcla \mathbf{A} con la evaluación de las muestras original con ruido multiplicativo. Imagen inferior izquierda. Tamaño de la matriz: 64 x 64. Resultado de la matriz de mezcla \mathbf{A} con la evaluación de las muestras original con ruido de sal y pimienta. Imagen inferior derecha. Tamaño de la matriz: 64 x 64. 116
- 4.12. dss_fastICA. Matriz de filtros estables resultante de la muestra original. Tamaño de la matriz: 64 x 76. 117
- 4.13. dss_fastICA. Matriz de filtros estables resultante de la muestra original con ruido gaussiano. Tamaño de la matriz: 64 x 19. . . 117
- 4.14. dss_fastICA. Matriz de filtros estables resultante de la muestra original con ruido multiplicativo. Tamaño de la matriz: 64 x 15. 118
- 4.15. dss_fastICA. Matriz de filtros estables resultante de la muestra original con ruido de sal y pimienta. Tamaño de la matriz: 64 x 75. 118

4.16. Descripción de las neuronas de la corteza visual primaria respondiendo selectivamente a bordes orientados en ángulos diversos. Imagen adaptada de la página web: <http://www.ucm.es/info/pslogica/mente/topo.htm> 121

4.17. Características típicas de una función gaussina, obtenidas de las imágenes de muestras 3 y 7 después de la inserción del ruido gaussiano. 123

4.18. Función de densidad obtenidas de las imágenes de muestras 3 y 7 después de la inserción del ruido multiplicativo. 124

4.19. Función de densidad obtenidas de las imágenes de muestras 3 y 7 después de la inserción del ruido de sal y pimienta. 125

4.20. Algoritmo fastICA. Evaluación de cada columna de \mathbf{A} para datos sin ruido. 126

4.21. Algoritmo fastICA. Evaluación de cada columna de \mathbf{A} para datos con ruido gaussiano. 127

4.22. Algoritmo fastICA. Evaluación de cada columna de \mathbf{A} para datos con ruido multiplicativo. 127

4.23. Algoritmo fastICA. Evaluación de cada columna de \mathbf{A} para datos con ruido de sal y pimienta. 128

4.24. Algoritmo fastICA. Filtros de componentes entre la señal original y la señal con ruido gaussiano de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación. 129

- 4.25. Algoritmo fastICA. Filtros de componentes entre la señal original y la señal con ruido multiplicativo de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación. 130
- 4.26. Algoritmo fastICA. Filtros de componentes entre la señal original y la señal con ruido de sal y pimienta de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación. 130
- 4.27. Algoritmo fastICA. Resultado de comparar la matriz de mezclas originales sin perturbaciones (parte superior) con la matriz de mezclas con ruido gaussiano (parte inferior). 131
- 4.28. Algoritmo fastICA. Resultado de comparar la matriz de mezclas originales sin perturbaciones (parte superior) con la matriz de mezclas con ruido multiplicativo (parte inferior). 133
- 4.29. Algoritmo fastICA. Resultado de comparar la matriz de mezclas originales sin perturbaciones (parte superior) con la matriz de mezclas con ruido de sal y pimienta (parte media e inferior). 134
- 4.30. Algoritmo fastICA. Resultado de poner 0 en la matriz de filtros \mathbf{A} con ruido de sal y pimienta los puntos con mayor valor. 135
- 4.31. Algoritmo fastICA. Resultado de poner 0 en la matriz de filtros \mathbf{A} con ruido de sal y pimienta presentada en la figura 4.8, los puntos con mayor valor. 136
- 4.32. Algoritmo JADE. Evaluación de cada columna de \mathbf{A} para datos originales. 137

4.33. Algoritmo JADE. Evaluación de cada columna de \mathbf{A} para datos con ruido gaussiano. 138

4.34. Algoritmo JADE. Evaluación de cada columna de \mathbf{A} para datos con ruido multiplicativo. 139

4.35. Algoritmo JADE. Evaluación de cada columna de \mathbf{A} para datos con ruido de sal y pimienta. 140

4.36. Algoritmo JADE. Ocho primeros filtros de componentes entre la señal original y la señal con ruido multiplicativo de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación. 141

4.37. Algoritmo JADE. Filtros de componentes entre la señal original y la señal con ruido de sal y pimienta de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación. . . 141

4.38. Algoritmo JADE. Filtros de componentes entre la señal original (parte superior y centro parte izquierda) y la señal con ruido multiplicativo (centro derecha y parte inferior) de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación. 143

4.39. Algoritmo JADE. Parte A. Filtros de componentes entre la señal original (parte A) y la señal con ruido de sal y pimienta (parte B) de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación. 144

- 4.40. Algoritmo JADE. Parte B. Filtros de componentes entre la señal original (parte A) y la señal con ruido de sal y pimienta (parte B) de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación. 145
- 4.41. Algoritmo JADE. Comparación de los diferentes filtros resultantes llevando a 0 los puntos con valor más alto con el fin de buscar la información contenida. En la parte superior se presenta el resultado con todos los primeros filtros encontrados una vez llevado a cabo JADE. En la parte inferior se presentan todos los filtros resultantes después de llevada a cabo la correlación. 147
- 4.42. Algoritmo AMUSE. Evaluación de cada columna de \mathbf{A} para datos originales. 149
- 4.43. Algoritmo AMUSE. Evaluación de cada columna de \mathbf{A} para datos con ruido gaussiano. Puede observarse como muchos de los filtros pierden las características típicas de una distribución gaussiana, dificultando más, la rápida evaluación. 150
- 4.44. Algoritmo AMUSE. Evaluación de cada columna de \mathbf{A} para datos con ruido multiplicativo. Se observa la prevalencia de la mayoría de los filtros con tendencia a comportarse como una distribución gaussiana, perdiendo las propiedades multiplicativas. 151

4.45. Algoritmo AMUSE. Evaluación de cada columna de \mathbf{A} para datos con ruido de sal y pimienta. Este resultado es interesante ya que los filtros para este caso, no se comportan como en casos anteriores, sino que parecieran describir otro tipo de característica de ruido. 152

4.46. Algoritmo AMUSE. Comparación entre los filtros de componentes entre la señal original y la señal con ruido gaussiano de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación. 153

4.47. Algoritmo AMUSE. Comparación entre los filtros de componentes entre la señal original y la señal con ruido multiplicativo de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación. 154

4.48. Algoritmo AMUSE. Comparación entre los filtros de componentes entre la señal original y la señal con ruido de sal y pimienta de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación. 154

4.49. Algoritmo AMUSE. Parte A. Resultado de ruido obtenido después de eliminar las componentes estables entre los datos originales (parte A) y la señal con ruido gaussiano (parte B). . . 156

4.50. Algoritmo AMUSE. Parte B. Resultado de ruido obtenido después de eliminar las componentes estables entre los datos originales (parte A) y la señal con ruido gaussiano (parte B). . . 157

- 4.51. Algoritmo AMUSE. Parte A. Resultado de ruido obtenido después de eliminar las componentes estables entre los datos originales (parte A) y la señal con ruido multiplicativo (parte B). 158
- 4.52. Algoritmo AMUSE. Parte B. Resultado de ruido obtenido después de eliminar las componentes estables entre los datos originales (parte A) y la señal con ruido multiplicativo (parte B). 159
- 4.53. Algoritmo AMUSE. Parte A. Resultado de ruido obtenido después de eliminar las componentes estables entre los datos originales (parte A) y la señal con ruido de sal y pimienta (parte B). 160
- 4.54. Algoritmo AMUSE. Parte B. Resultado de ruido obtenido después de eliminar las componentes estables entre los datos originales (parte A) y la señal con ruido de sal y pimienta (parte B). 161
- 4.55. Algoritmo fastGEO. Evaluación de cada columna de \mathbf{A} para datos originales. Puede apreciarse una distribución clara en los filtros de información que contienen bordes o esquinas de interés. 162

4.56. Algoritmo fastGEO. Evaluación de cada columna de \mathbf{A} para datos con ruido gaussiano. Se observan filtros en su mayoría contaminados, donde la evaluación de su distribución se comporta de forma gaussiana. 163

4.57. Algoritmo fastGEO. Evaluación de cada columna de \mathbf{A} para datos con ruido multiplicativo. La mayoría de los filtros contaminados no presentan la distribución típica del ruido multiplicativo hasta ahora trabajado. En muchos casos, estos resultados podrían ser confundidos con la forma de distribución del ruido gaussiano. 164

4.58. Algoritmo fastGEO. Evaluación de cada columna de \mathbf{A} para datos con ruido de sal y pimienta. Se retoman los resultados presentados por los algoritmos de fastICA y JADE; pero puede observarse también la presencia más numerosa de algunos filtros representando bordes. 165

4.59. Algoritmo fastGEO. Primeros ocho filtros de componentes entre la señal original y la señal de ruido gaussiano de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación. 166

4.60. Algoritmo fastGEO. Primeros ocho filtros de componentes entre la señal original y la señal de ruido multiplicativo de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación. 166

- 4.61. Algoritmo fastGEO. Primeros ocho filtros de componentes entre la señal original y la señal de ruido de sal y pimienta de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación. 167
- 4.62. Algoritmo fastGEO. Resultado después de comparar las muestras originales (parte superior y centro), con las muestras de ruido gaussiano (parte inferior). 168
- 4.63. Algoritmo fastGEO. Parte A. Resultado después de comparar las muestras originales (parte A), con las muestras de ruido multiplicativo (parte B). 169
- 4.64. Algoritmo fastGEO. Parte B. Resultado después de comparar las muestras originales (parte A), con las muestras de ruido multiplicativo (parte B). 170
- 4.65. Algoritmo fastGEO. Parte A. Resultado después de comparar las muestras originales (parte A), con las muestras de ruido de sal y pimienta (parte B). 171
- 4.66. Algoritmo fastGEO. Parte B. Resultado después de comparar las muestras originales (parte A), con las muestras de ruido de sal y pimienta (parte B). 172

4.67. Algoritmo fastGEO. Comparación de las diferentes componentes resultantes llevando a 0 los puntos mas altos con el fin de buscar que hay debajo. A la derecha todas las componentes sin realizar la correlación. A la izquierda todas las componentes una vez llevada a cabo la correlación. 173

4.68. dss_fastICA. Evaluación de cada columna de \mathbf{A} para datos originales. 175

4.69. dss_fastICA. Evaluación de cada columna de \mathbf{A} para datos con ruido gaussiano. 176

4.70. dss_fastICA. Evaluación de cada columna de \mathbf{A} para datos con ruido multiplicativo. 176

4.71. dss_fastICA. Evaluación de cada columna de \mathbf{A} para datos con ruido de sal y pimienta. 177

4.72. Algoritmo dss_fastICA. Primeros ocho filtros de componentes entre la señal original y la señal de ruido gaussiano de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación. 178

4.73. Algoritmo dss_fastICA. Primeros ocho filtros de componentes entre la señal original y la señal de ruido multiplicativo de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación. 179

- 4.74. Algoritmo `dss_fastICA`. Filtros de componentes entre la señal original y la señal de ruido de sal y pimienta de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación. . . . 179
- 4.75. Algoritmo `dss_fastICA`. Comparación entre los filtros originales (parte superior y media) y los filtros de componentes con ruido gaussiano (parte inferior) de las respectivas matrices de mezcla \mathbf{A} 181
- 4.76. Algoritmo `dss_fastICA`. Comparación entre los filtros originales (parte superior y media) y los filtros de componentes con ruido multiplicativo (parte inferior) de las respectivas matrices de mezcla \mathbf{A} 182
- 4.77. Algoritmo `dss_fastICA`. Comparación entre los filtros originales (parte superior y media) y los filtros de componentes con ruido de sal y pimienta (parte inferior) de las respectivas matrices de mezcla \mathbf{A} 183
- 4.78. Algoritmo `dss_fastICA`. Comparación de las diferentes componentes resultantes llevando a 0 los puntos mas altos con el fin de buscar que hay debajo. En la parte superior todas los filtros aún sin realizar la correlación. En la parte inferior todo los filtros una vez llevada a cabo la correlación. 184

4.79. Algoritmo AMUSE. Resultados de aplicar el algoritmo AMUSE realizando previamente PCA a las señales originales. Parte superior izquierda, reducción de dimensiones a 49; parte superior derecha, reducción de dimensiones a 36; parte central izquierda, reducción de dimensiones a 25; parte central derecha, reducción de dimensiones a 16 y parte inferior, reducción de dimensiones a 9. Con este número reducido de dimensiones puede apreciarse filtros con información libre de mezclas. . . . 188

4.80. Algoritmo AMUSE. Resultados de aplicar el algoritmo AMUSE realizando previamente PCA a las señales con ruido gaussiano. Parte superior izquierda, reducción de dimensiones a 49; parte superior derecha, reducción de dimensiones a 36; parte central izquierda, reducción de dimensiones a 25; parte central derecha, reducción de dimensiones a 16 y parte inferior, reducción de dimensiones a 9. Puede observarse al final la obtención de una determinada reducción de filtros libres de cualquier influencia de ruido gaussiano. . . . 189

- 4.81. Algoritmo AMUSE. Resultados de aplicar el algoritmo AMUSE realizando previamente PCA a las señales con ruido multiplicativo. Parte superior izquierda, reducción de dimensiones a 49; parte superior derecha, reducción de dimensiones a 36; parte central izquierda, reducción de dimensiones a 25; parte central derecha, reducción de dimensiones a 16 y parte inferior, reducción de dimensiones a 9. Puede observarse una reducción de filtros que no poseen influencia de ruido multiplicativo. . . . 190
- 4.82. Algoritmo AMUSE. Resultados de aplicar el algoritmo AMUSE realizando previamente PCA a las señales con ruido de sal y pimienta. Parte superior izquierda, reducción de dimensiones a 49; parte superior derecha, reducción de dimensiones a 36; parte central izquierda, reducción de dimensiones a 25; parte central derecha, reducción de dimensiones a 16 y parte inferior, reducción de dimensiones a 9. Puede observarse al final que la información contenida en los filtros resultantes, son datos libres de ruido de sal y pimienta. 191
- 5.1. Algoritmo AMUSE. Parte A. Resultados obtenidos después de aplicar la técnica ICA al grupo de Imágenes de muestra contaminadas con ruido gaussiano. Imágenes mostradas de la 1 a la 8. 201

5.2. Algoritmo AMUSE. Parte B. Resultados obtenidos después de aplicar la técnica ICA al grupo de Imágenes de muestra contaminadas con ruido gaussiano. Imágenes mostradas de la 9 a la 15. 202

5.3. Algoritmo AMUSE. Parte A. Resultados obtenidos después de aplicar la técnica ICA al grupo de Imágenes de muestra contaminadas con ruido multiplicativo. Imágenes mostradas de la 1 a la 8. 203

5.4. Algoritmo AMUSE. Parte B. Resultados obtenidos después de aplicar la técnica ICA al grupo de Imágenes de muestra contaminadas con ruido multiplicativo. Imágenes mostradas de la 9 a la 15. 204

5.5. Algoritmo AMUSE. Parte A. Resultados obtenidos después de aplicar la técnica ICA al grupo de Imágenes de muestra contaminadas con ruido de sal y pimienta. Imágenes mostradas de la 1 a la 8. 205

5.6. Algoritmo AMUSE. Parte B. Resultados obtenidos después de aplicar la técnica ICA al grupo de Imágenes de muestra contaminadas con ruido de sal y pimienta. Imágenes mostradas de la 9 a la 15. 206

- 5.7. Algoritmo AMUSE. Parte A. Resultados obtenidos después de aplicar la técnica PCA al grupo de Imágenes de muestra contaminadas con ruido gaussiano. Imágenes mostradas de la 1 a la 8. 207
- 5.8. Algoritmo AMUSE. Parte B. Resultados obtenidos después de aplicar la técnica PCA al grupo de Imágenes de muestra contaminadas con ruido gaussiano. Imágenes mostradas de la 9 a la 15. 208
- 5.9. Algoritmo AMUSE. Parte A. Resultados obtenidos después de aplicar la técnica PCA al grupo de Imágenes de muestra contaminadas con ruido multiplicativo. Imágenes mostradas de la 1 a la 8. 209
- 5.10. Algoritmo AMUSE. Parte B. Resultados obtenidos después de aplicar la técnica PCA al grupo de Imágenes de muestra contaminadas con ruido multiplicativo. Imágenes mostradas de la 9 a la 15. 210
- 5.11. Algoritmo AMUSE. Parte A. Resultados obtenidos después de aplicar la técnica PCA al grupo de Imágenes de muestra contaminadas con ruido de sal y pimienta. Imágenes mostradas de la 1 a la 8. 211

5.12. Algoritmo AMUSE. Parte B. Resultados obtenidos después de aplicar la técnica PCA al grupo de Imágenes de muestra contaminadas con ruido de sal y pimienta. Imágenes mostradas de la 9 a la 15. 212

Índice de cuadros

2.1. Espectro electromagnético	51
3.1. Tipos de mezclas de información y algoritmos orientados a ellas	80
5.1. Tabla comparativa de las técnicas ICA y PCA en busca de las diferencias en el cálculo del error cuadrático medio. Datos extraídos de las muestras originales y de las contaminadas con ruido gaussiano. Puede apreciarse diferencias mínimas en los resultados.	215
5.2. Tabla comparativa de las técnicas ICA y PCA en busca de las diferencias en el cálculo del error cuadrático medio. Datos extraídos de las muestras originales y de las contaminadas con ruido multiplicativo. Puede apreciarse diferencias mínimas en los resultados.	216

- 5.3. Tabla comparativa de las técnicas ICA y PCA en busca de las diferencias en el cálculo del error cuadrático medio. Datos extraídos de las muestras originales y de las contaminadas con ruido de sal y pimienta. Puede apreciarse diferencias mínimas en los resultados. 217

Resumen

En la actualidad, el procesamiento de señales, en especial el procesamiento de imágenes adquiere cada día más importancia. Son muchos los campos en los que se requiere un mejoramiento visual de la información con el fin de obtener buenos resultados. De estos campos puede destacarse ramas como la de la medicina con imágenes de rayos X, tomografías, fMRI, etc, biología, geografía, arqueología, astronomía, defensa y fuerzas militares, entre otras.

La investigación en general a través de los años, a permitido determinar como conclusión clara que el conocer un problema a fondo permite búsquedas de soluciones más adecuadas y efectivas para el mismo. Es aquí donde radica la importancia del desarrollo de esta tesis como un estudio comparativo del diferentes algoritmos más usados, basados en la técnica del análisis de componentes independientes y el ruido en imágenes.

A nivel general el ruido se presenta como una mezcla de señales aleatorias con diferentes frecuencias las cuales pueden ser mayores o menores a las señal de interés. El problema de la separación ciega de señales consiste en la recuperación de las señales originales a partir de las mezclas detectadas por

sensores, conociendo tan sólo estas últimas. Estas mezclas de señales tiene lugar en el medio en que se propagan y en los sensores y como característica de éste método es que a priori no se cuenta con ninguna información de las señales originales ni de la forma en que fueron mezcladas.

El presente trabajo doctoral tiene como objetivo la utilización de la herramienta del análisis de componentes independientes como estudio comparativo de un problema común que se presenta en muchas aplicaciones y que al final se convierte en el primer paso del preprocesamiento de cualquier señal independiente o no de que sea una imagen.

La presente memoria se encuentra estructurada como sigue:

Capítulo 1: Introducción al análisis de componentes independientes y la separación ciega de señales. En este capítulo se presenta la definición y formulación matemática de la técnica de Análisis de Componentes Independientes aplicado al problema de la Separación Ciega de Señales. En primer lugar se presentan los fundamentos estadísticos y su importancia en la separación de señales. A continuación se introduce en el tema de teoría de la información y los fundamentos básicos necesarios para la técnica ICA. Seguidamente se introducen las técnicas del Análisis de Componentes Principales (PCA) y el Análisis de Componentes Independientes (ICA) con sus correspondientes definiciones matemáticas, restricciones y algoritmos. Finalmente se realiza una comparación entre PCA e ICA.

Capítulo 2: Conexión entre el sistema visual humano (SVH), el análisis de componentes independientes (ICA) y el ruido. Es este capítulo se presenta definiciones y la conexión entre la técnica de ICA el procesamiento de imágenes y el ruido. En primera instancia se hace una reseña historia retomando los últimos años de investigación en el procesamiento de imágenes con una introducción al sistema visual humano y la representación del color. Después se introduce a la relación de ICA con el estudio del ruido y las imágenes, para terminar introduciendo las clases de ruido que el este trabajo se pretenden estudiar con sus características.

Capítulo 3: Algoritmos ICA. En este capítulo se desarrollan los algoritmos de la técnica ICA más utilizados por los diferentes investigadores, explicando sus ventajas y desventajas con el fin de preparar el camino para su utilización en el siguiente capítulo.

Capítulo 4: El ruido en ICA. En este capítulo se evalúa el comportamiento de cinco diferentes algoritmos ICA (fastICA, JADE, AMUSE, fastGEO y dss_fastICA), en respuesta a tres clases diferentes de ruido (gaussiano, multiplicativo e impulsivo (sal y pimienta), aplicados al área del procesamiento digital de imágenes. De manera particular se estudia como se comportan los diferentes filtros de las matrices de mezcla \mathbf{A} obtenidas de las observaciones y la influencia de cada tipo de ruido sobre la señal de muestra.

Capítulo 5: Aplicación al de-noising. Por si sola, el querer recuperar la señal de datos originales sin perturbaciones parásitas por medio de la matriz de mezcla es tarea difícil. En este capítulo se ve como al recurrir a la matriz de fuentes \mathbf{S} de las señales observadas, junto con unos pocos filtros de cada una de las matrices de mezcla \mathbf{A} , pueden obtenerse resultados bastante satisfactorios a la hora de reducción de cualquiera de las clases tres clases de ruido aquí trabajadas.

En la sección de “Conclusiones y trabajos futuro” se resumen las principales aportaciones que se han realizado en la materia con el desarrollo del presente trabajo doctoral.

Introducción

Empezaremos por aclarar que es una imagen. A nivel teórico, una imagen es una función de dos dimensiones con coordenadas espaciales, matemáticamente esto se representa como:

$$f(x, y)$$

donde las variables x, y representan coordenadas espaciales.

Las imágenes digitales son imágenes $f(x, y)$ que ha sido discretizadas en el espacio y en magnitud. Este tipo de imágenes puede considerarse como una matriz cuyos índices de fila y columna identifican un punto en la imagen y el correspondiente valor del elemento de la matriz equivale al valor de magnitud o nivel de gris en ese punto. Por ejemplo, para imágenes en escala de grises el nivel de gris o intensidad en un punto de la imagen, se encuentra representado por la amplitud de f . Las imágenes digitales se componen de un número finito de elementos, en donde cada uno de ellos cuenta con una localización y valor específico; a esto se le conoce comúnmente como pixel.

El nacimiento y evolución del procesamiento y el análisis de imágenes digitales así como el estudio de las propiedades topológicas de las mismas tienen lugar a principios de los años 60, cuando se dispone de tecnología para poder captar y manipular grandes cantidades de información espacial en forma de matrices de valores, lo cual permite el desarrollo de nuevas técnicas orientadas al análisis de datos multidimensionales. A nivel histórico la primera vez que se usó el procesamiento de imágenes digital fue con la transmisión de fotografías por medio de un cable submarino desde New York a Londres, en los años 20. Este sistema de transmisión comenzaba codificando las imágenes a transmitir por medio de un cable y una vez alcanzado el objetivo eran reconstruidas por un receptor de equipos de impresión especializado. A este sistema de transmisión se le denominó Bartlane [60] [80].

La evolución del diseño de arquitectura de computadores alcanzado en los 80 permitió un desarrollo más rápido de diferentes técnicas de procesamiento, presentando las mayores contribuciones orientadas a la elaboración de algoritmos para la detección de características como bordes, líneas, texturas, etc, así como en el desarrollo de técnicas globales de segmentación de una imagen en regiones. Igualmente, se comienza con el desarrollo de nuevas teorías y algoritmos que permitan la interpretación de imágenes en segunda y tercera dimensión, así como el estudio del movimiento, de las formas a partir de las sombras, búsqueda de objetos, etc. [60] [80].

Desde el año 1964 hasta la actualidad, el campo del procesamiento de imágenes ha crecido vigorosamente, permitiendo utilizar las técnicas de PDI

(procesamiento digital de imágenes), para resolver problemas que requieren comúnmente, métodos capaces de mejorar la información visual para interpretación humana y análisis. Igualmente investigaciones que toman como referencia el funcionamiento del sistema visual humano (SVH) han aportado grandes avances en este tema. En este sentido se destacan las investigaciones llevadas a cabo por David H. Hubel y Torsten N. Wiesel [37], [38], ganadores del Premio Nobel de Medicina en 1981, por sus investigaciones sobre como la corteza visual analiza la información que capta la retina. Gracias a esta investigación se sabe hoy en día que las mayoría de las neuronas corticales en presencia de estímulos visuales consistentes en contornos orientados responden con mayor intensidad.

Propiamente en el campo de la técnica ICA, en 1997 los investigadores Bell y Sejnowski [15], basándose en las investigaciones de Hubel y Wiesel; establecieron una conexión entre la dicha técnica aplicada a imágenes naturales y la forma como se comportan ciertas neuronas de la corteza visual primaria.

Actualmente, todos estos estudios logrados con el tiempo han permitido realizar una catalogación de las diferentes técnicas existentes en procesamiento de imágenes, en donde todas aquellas técnicas asociadas a la captura, codificación y representación de las imágenes, las cuales no introducen sobre las mismas ningún tipo de interpretación, se denomina conceptos de técnicas para el procesamiento de imágenes digitales; mientras que las técnicas para extraer información presente en la imagen con el fin de hacer una interpre-

tación de las escena representada se le conoce como técnicas para el análisis de imágenes digitales.

Como se menciona con anterioridad existen infinidad de ramas en donde se requiere del procesamiento de imágenes para mejorar la información visual con el fin de una correcta interpretación y análisis humano. Algunas de las ramas a destacar son:

Medicina y medicina nuclear: los rayos X y las proyecciones CT médicas se digitalizan para examinar áreas internas del cuerpo. En general, las técnicas PDI logran mejora el contraste de las imágenes o codifican los niveles de intensidad luminosa en colores para una mayor facilidad de interpretación visual de imágenes biomédicas como las radiografías y las ecografías. [63], [55], [34], [52], [45], [59], [12], [76].

Biología: se realiza con el análisis de muestras biológicas a nivel visual, permitiendo la clasificación, análisis, identificación automática y categorización de las diferentes muestras. [60], [80].

Geografía: el procesamiento de imágenes geográficas permite el análisis, previsión y eliminación de algunos problemas ambientales. En ésta área en particular se pueden obtener imágenes de diversas fuentes tales como satélites o fotografías aéreas en formato digital. [60], [80].

Arqueología: esta rama se encuentra enmarcada dentro de la Antropología y tiene como objetivo la búsqueda de información acerca de las características de las diferentes comunidades humanas. El procesamiento de

imágenes proporciona un mejor análisis y permite mayores posibilidades para alcanzar mejores resultados en la investigación. [60], [80].

Astronomía: en esta área en particular el procesamiento de imágenes permite desarrollar infinidad de posibilidades con el fin de obtener datos de las imágenes captadas por satélites. [60], [80].

Defensa y fuerzas militares: permiten monitorizar la precipitación atmosférica para fines operacionales y de investigación. A nivel histórico en la segunda guerra mundial aparecieron los primeros radares que manejaban el principio Doppler, estos tenían como fin detectar mejor a los aviones y otros objetos móviles en presencia de ecos de fondo “perturbadores” [60], [80].

Procesado de datos: Una recolección y procesado automático de documentos e imágenes se presenta útil para bancos y compañías de seguros. Estos documentos están digitalmente comprimidos y guardados. Se detectan e identifican automáticamente la información impresa sobre cheques y otros documentos contables. [60], [80].

Automatización en la industria: El procesamiento de imágenes se usa para la inspección y supervisión automática en líneas de producción de grandes empresas. Este sistema reduce mucho el error humano al tiempo que proporciona estabilidad y precisión en la producción. [60], [80].

Así mismo existen muchas más áreas en donde las imágenes juegan un papel básico: la automatización en la industria, la teledetección, el procesado de documentos, la toma de imágenes de video en tiempo real, la meteorología, marcas de agua, ciencias de los materiales (texturas), etc, y es mucho el trabajo que se puede elaborar con cada una de las imágenes disponibles a procesar, como por ejemplo el cambio de contraste y de dimensiones, filtros para suavizar o resaltar los bordes, etc.

Capítulo 1

INTRODUCCIÓN AL ANÁLISIS DE COMPONENTES INDEPENDIENTES Y A LA SEPARACIÓN CIEGA DE SEÑALES

Durante el desarrollo de este capítulo se hará una introducción al los conceptos de Separación Ciega de Señales, por sus siglas en ingles BSS (Blind Source Separation) y el Análisis de Componentes Independientes, conocido

como ICA (Independent Component Analysis). Igualmente se dará una visión panorámica de los procedimientos aplicaciones más conocidos en BSS-ICA, los distintos modelos e hipótesis que se suelen considerar para la resolución del mismo y las diferentes aplicaciones que pueden tener.

1.1. Visión general

Una de las aplicaciones de la técnica de ICA (Análisis de Componentes Independientes) es BSS (Separación Ciega de Fuentes). BSS consiste en la recuperación de señales originales partiendo de mezclas observadas. En principio fué planteado como una solución matemática al problema biológico del sistema nervioso central, en el cual se presenta la transmisión de información mezclas por distintas fibras nerviosas [36]. En el ámbito de las comunicaciones Barnes [11], proponía soluciones similares usando como ejemplo el conocido efecto “Cocktail Party”, el cual se produce cuando varios locutores hablan al mismo tiempo y cuyo objetivo final consiste en separar cada una de esas señales de voz para poder escucharlas sin las influencias de las otras, como si sólo hablara un locutor, mientras los demás permanecen callados; en otras palabras la idea general de ICA resulta de querer conocer ciertas señales que han sido mezcladas en el medio con anterioridad y de las cuales no se conocen las fuentes, ni las características exactas del medio donde se han mezclado.

A manera ilustrativa para el procesamiento de imágenes podemos ver lo

anteriormente descrito de manera más clara por medio de un ejemplo. A continuación se presentan dos imágenes originales (ver figura 1.1) las cuales se convierten en nuestras fuentes **S**. Como puede apreciarse en un principio estas dos imágenes no poseen ninguna clase de distorsión o ruido que puedan interferir de alguna forma con alguna investigación que se desee llevar a cabo con ellas; pero este es el caso ideal, ya que generalmente en la transmisión y recepción de información los datos obtenidos de alguna manera están siempre contaminados por alguna clase de ruido.



Figura 1.1: Imágenes originales sin ninguna perturbación.

Llevada a cabo la recepción de la información por medio de sensores y tal como hemos descrito con anterioridad generalmente obtenemos información mezclada de la cual en la realidad, no conocemos las fuentes ni la forma como se han mezclado la información, ver figura 1.2

Lo que pretende la técnica ICA es a partir de esta información mezclada



Figura 1.2: Mezclas obtenidas de las figuras originales.

reconstruir la información original original tal cual y como se aprecia en la figura 1.1. Resulta interesante resaltar que el modelo ICA esta enfocado siempre en la búsqueda continua de las proyección de datos interesantes dentro de todo el conjunto de señales de mezcla que obtenemos.

Por medio del uso de variables aleatorias podemos observar otro ejemplo representado en la figura 1.3. La figura del lado derecho representa las variables mezcladas por una matriz de mezcla $\mathbf{A} = \begin{bmatrix} 1 & 2 \\ 1 & 1 \end{bmatrix}$. En ésta puede apreciarse que una vez llevada a cabo la mezcla existe una concentración de puntos y un desplazamiento de los ejes el cual se encuentra limitado por el tipo de mezcla especificado.

A continuación se presenta los fundamentos matemáticos y estadísticos de ICA.

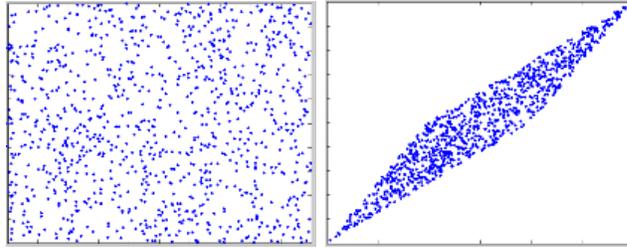


Figura 1.3: Izquierda: variables aleatorias. Derecha: señales mezcladas.

1.2. Análisis de Componentes Independientes (ICA)

El análisis componente independiente (ICA) es una técnica estadística para buscar factores base en sistemas de variables aleatorias, de medidas, o de señales.

Generalmente ICA define un modelo de datos multivariantes observados, que se presenta como muestras. En el modelo, las variables son mezclas lineares o no lineares de variables desconocidas, así como el sistema en que se mezclan. Para el modelo ICA, como se vera más adelante es de gran importancia que las variables sean no gaussianas y mutuamente independientes. Una vez obtenidos los datos, estos se conocen como las componentes independientes.

Retomando el ejemplo del efecto “Cocktail Party”, podemos asumir una mezcla lineal x_1, x_2, \dots, x_n de n componentes independientes [49][25], lo cual

se puede expresar matemáticamente:

$$x_j = a_{j1}s_1 + a_{j2}s_2 + a_{j3}s_3 + a_{jn}s_n \text{ para todo } j \quad (1.1)$$

donde cada x_j representa la mezcla y cada s_k representa las componentes independientes. Estas son variables aleatorias y tienen media cero.

Expresando lo anteriormente descrito de manera vectorial y matricial: sea \mathbf{x} el vector aleatorio con mezcla de los elementos x_1, x_2, \dots, x_n , y el vector aleatorio \mathbf{s} con s_1, s_2, \dots, s_n . Tenemos \mathbf{A} la matriz con los elementos a_{ij} . De esta forma obtenemos el modelo de mezcla expresado como:

$$\mathbf{x} = \mathbf{A}\mathbf{s} \quad (1.2)$$

Si denotamos las columnas de la matriz de \mathbf{A} como \mathbf{a}_j podemos escribir el modelo como:

$$\mathbf{x} = \sum_{i=1}^n \mathbf{a}_i s_i \quad (1.3)$$

El modelo estadístico de ICA se conoce como:

$$\mathbf{s} = \mathbf{W}\mathbf{x} \quad (1.4)$$

El modelo de ICA es un modelo generativo, esto es, los datos observados provienen de una proceso de mezcla de las componentes originales, donde dichas componentes independientes no pueden ser observadas de manera di-

recta y la forma en que se han mezclado se desconoce. Los únicos datos con los que contamos en la observación son \mathbf{x} , con estos datos se deben estimar \mathbf{A} y \mathbf{s} .

Con el fin de hacer el análisis más simple y poder llevar a cabo el estudio se deben de establecer 3 hipótesis:

Las fuentes son estadísticamente independientes entre sí. Esto es, un

conjunto de variables aleatorias y_1 y y_2 es independiente si la información proporcionada por cualquiera de las variables y_1 no proporciona ninguna información acerca de los valores de y_2 . De esta manera la independencia estadística puede ser formulada a partir de la densidad de probabilidad. Matemáticamente podemos denotemos mediante $p(y_1, y_2, \dots, y_n)$ la función de densidad de probabilidad conjunta de y_1, y_2, \dots, y_n y $p_i(y_i)$ como la función de densidad de probabilidad marginal de y_i . Para estos datos decimos que los y_i son independientes si y sólo si la función de densidad de probabilidad conjunta puede ser factorizada como:

$$p(y_1, y_2, \dots, y_n) = p_1(y_1)p_2(y_2)\dots p_n(y_n) \quad (1.5)$$

Resulta importante también resaltar que las variables no correlacionadas son solamente en parte independientes. Tal y como se ha visto desde el punto de vista estadístico la independencia estadística de las fuentes puede ser considerada restrictiva; pero a nivel práctico en mu-

chas situaciones puede suponerse que señales generadas por distintos procesos sean independientes entre sí. Esto hace a la independencia estadística mucho más restrictiva que la correlación ya que pueden tenerse señales mezcladas resultantes que están incorreladas; pero no son independientes.

Para la verificación de independencia de las fuentes se suele recurrir a métodos como la minimización de la información mutua o los momentos y cumulantes.

Las fuentes tienen distribución no-gaussiana. Esto es, como máximo solo una de ellas puede ser gaussiana, debido a que al presentarse variables con distribución gaussiana, los estadísticos de alto orden usados por ICA presentarían valores nulos. Matemáticamente, asumiendo la matriz de mezcla ortogonal y las fuentes \mathbf{s}_i gaussianas, tendríamos x_1 y x_2 también gaussianas, sin ninguna correlación y de varianza unitaria donde la densidad unitaria estaría dada por:

$$p(x_1, x_2) = \frac{1}{2\pi} \exp\left(-\frac{x_1^2 + x_2^2}{2}\right) \quad (1.6)$$

De acuerdo a la ecuación 1.6 y a la figura 1.4 donde se representa una señal gaussiana, puede observarse que no existe ninguna información sobre las direcciones de las columnas de la matriz de mezcla \mathbf{A} , haciendo imposible una solución.

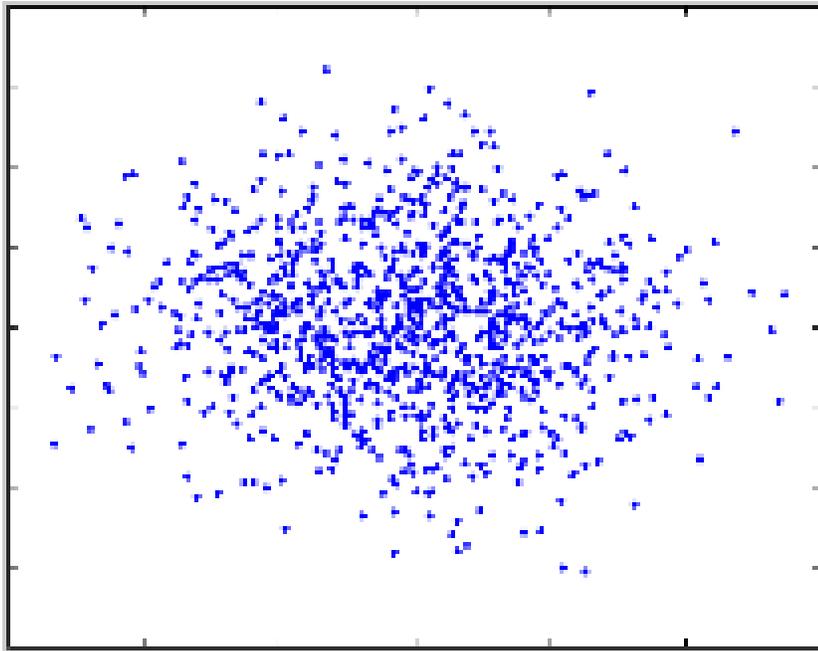


Figura 1.4: Gráfica de señales gaussianas.

El modelo de mezcla se conoce. Este modelo puede ser lineal, convolutivo o no lineal.

1.2.1. Ambigüedades de ICA

Pre-estableciendo éstas hipótesis, la técnica ICA puede ser llevada a cabo; pero existen ciertas indeterminaciones triviales relacionadas con el orden de las componentes y su amplitud [45], es por ello que generalmente los resultados que se obtienen desarrollando el modelo presentan a las señales recuperadas en diferente orden de las señales originales y afectas por un factor de escala que las puede aumentar o disminuir. De la ecuación 1.4 resulta fácil ver las dos siguientes ambigüedades:

El orden de las IC no se puede determinar. Esto se debe a que tanto \mathbf{s} como \mathbf{A} son desconocidos; la matriz de permutación \mathbf{P} puede ser sustituida obteniendo $\mathbf{x} = \mathbf{AP}^{-1}\mathbf{Ps}$, donde las variables originales independientes s_j se encuentran en diferente orden en los elementos de \mathbf{Ps} . De esta forma se pueden cambiar el orden de los términos.

Las variaciones de las IC no se pueden determinar. Esto se debe a que tanto \mathbf{s} como \mathbf{A} son desconocidos, cualquier multiplicador escalar en una de las fuentes s_i podría ser nulo al dividir la columna \mathbf{a}_i de \mathbf{A} por el mismo escalar. Por esta razón pueden ser preestablecidas las magnitudes de las componentes independientes; ya que son variables aleatorias, para esto asumimos $E(s_i^2) = 1$. Entonces la matriz \mathbf{A} es-

1.3. PRINCIPIOS MATEMÁTICOS PARA LA VALORACIÓN DE ICA11

tará adaptada a los métodos de la solución de ICA para considerar esta restricción.

1.3. Principios matemáticos para la valoración de ICA

1.3.1. Función de distribución acumulada

La función de distribución acumulada (cdf, cumulative distribution function), puede ser representada como $F_\chi(x_0)$. Corresponde a la probabilidad de que una variable aleatoria real χ pueda tomar un valor numérico menor o igual a x_0 . Matemáticamente se tiene:

$$F_\chi(x_0) = P(\chi \leq x_0) \text{ para } x_0 \in \mathbb{R} \quad (1.7)$$

Esta función de probabilidad cumple las siguientes propiedades:

1. $F_\chi(x_0)$ es no negativo:

$$F_\chi(x_0) \geq 0 \quad \forall \quad x_0 \in \mathbb{R}$$

2. F_χ no decrece:

$$F_\chi(x_1) \leq F_\chi(x_2) \quad \forall \quad x_1 \leq x_2, \quad x_1, x_2 \in \mathbb{R}$$

3. El recorrido de la función de probabilidad acumulada es:

$$0 \leq F_{\chi}(x_0) \leq 1 \quad \forall \quad x_0 \in \mathbb{R}$$

4. Cuando el valor de la variable es grande el valor de la función se tiene de 1:

$$F_{\chi}(\infty) = 1$$

5. Cuando el valor de la variable es pequeño el valor de la función se tiene de 0:

$$F_{\chi}(-\infty) = 0$$

1.3.2. Función de densidad de probabilidad

La función de densidad de probabilidad (fdp, pdf, probability density function) p_{χ} de una variable aleatoria continua χ es una función que se integra para obtener la probabilidad que la variable aleatoria toma un valor en un intervalo predefinido:

$$\int_{x_1}^{x_2} P(\chi)(t)dt = P(x_1 \leq \chi \leq x_2) \quad (1.8)$$

Esta función de densidad de probabilidad esta relacionada con la distri-

1.3. PRINCIPIOS MATEMÁTICOS PARA LA VALORACIÓN DE ICA13

bución de probabilidad de acuerdo a:

$$P(\chi)(x_0) = \left. \frac{dF\chi(x)}{dx} \right|_{x=x_0} \quad (1.9)$$

La relación inversas de ambas funciones es:

$$F\chi(x_0) = \int_{-\infty}^{x_0} P(\chi)(t) dt \quad (1.10)$$

Al igual que la función de distribución acumulada, la función de densidad de probabilidad cumple con las siguientes propiedades:

1. $P_\chi(x_o)$ es no negativo:

$$P_\chi(x_0) \geq 0 \quad \forall \quad x_0 \in \mathbb{R}$$

2. El recorrido de la función de densidad de probabilidad es:

$$0 \leq P_\chi(x_0) \leq 1 \quad \forall \quad x_0 \in \mathbb{R}$$

3. La función de densidad de probabilidad esta normalizada:

$$\int_{x_1}^{x_2} P(\chi)(t) dt = 1$$

1.3.3. Funciones de distribución y densidad de vectores aleatorios

Para el caso de multivariantes se debe de definir la función de distribución acumulada y la función de densidad de probabilidad en analogía con el caso univariante. Esto es:

Sea $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N]$ un vector aleatorio real compuesto por las variables aleatorias $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$, todas obtenidas de un mismo experimento estadístico. La función de distribución acumulada de X se define como:

$$F_{\chi}(x_1, x_2, \dots, x_N) = P(\chi_1 \leq x_1, \chi_2 \leq x_2, \dots, \chi_N \leq x_N) \quad (1.11)$$

De acuerdo con esta definición la función de probabilidad de un vector aleatorio puede ser determinado como en la ecuación (1.11) sustituyendo para este caso las derivadas totales por parciales respecto a cada una de las variables obtenemos:

$$P(\chi)(x_{1_0}, x_{2_0}, \dots, x_{N_0}) = \frac{\partial F_{\chi}(x)}{\partial x_1} \Big|_{x_1=x_{1_0}} \frac{\partial F_{\chi}(x)}{\partial x_2} \Big|_{x_2=x_{2_0}} \dots \frac{\partial F_{\chi}(x)}{\partial x_N} \Big|_{x_N=x_{N_0}} \quad (1.12)$$

1.3.4. Distribuciones y densidades conjuntas y marginales

Los conceptos anteriores pueden ser extendidos también a problemas con dos vectores aleatorios diferentes \mathbf{x} y \mathbf{y} donde la dimensión m del vector \mathbf{y} puede ser diferente que la dimensión n del vector \mathbf{x} . Si se concatenan los dos vectores para obtener un “supervector” $\mathbf{z} = [\mathbf{x}, \mathbf{y}]$ las fórmulas precedentes pueden ser utilizadas directamente.

Podemos definir la función de distribución conjunta como:

$$F_{\mathbf{x},\mathbf{y}}(\chi_0 y_0) = P(\mathbf{x} \leq \chi_0, \mathbf{y} \leq y_0) \quad (1.13)$$

y la función de densidad conjunta como:

$$P_{\mathbf{x},\mathbf{y}}(\chi_0 y_0) = \left. \frac{\partial F_{\mathbf{x},\mathbf{y}}(x, y)}{\partial x} \right|_{x=\chi_0} \left. \frac{\partial F_{\mathbf{x},\mathbf{y}}(x, y)}{\partial y} \right|_{y=y_0} \quad (1.14)$$

donde x_0 y y_0 son un vector constante de valores concretos del conjunto de variables que forman X y Y respectivamente. Asimismo, la función de distribución se deriva de la función de densidad conjunta.

$$F_{\mathbf{x},\mathbf{y}}(\chi_0 y_0) = \int_{-\infty}^{\chi_0} \int_{-\infty}^{y_0} P_{\mathbf{x},\mathbf{y}}(\mu\nu) d\mu d\nu \quad (1.15)$$

Dada la función de densidad conjunta las dos funciones de densidad de probabilidad individuales $p_X(x)$ de X y $p_Y(y)$ de Y , llamadas funciones de

densidad marginales, pueden ser también obtenidas. Por eso, se integra la función de densidad conjunta sobre uno de los dos vectores aleatorios como sigue:

$$P_x(x) = \int_{-\infty}^{\infty} P_{xy}(\chi\mu)d\mu \quad (1.16)$$

$$P_y(y) = \int_{-\infty}^{\infty} P_{xy}(\nu y)d\mu \quad (1.17)$$

1.3.5. Teorema del límite central

El teorema de límite central orientado hacia la teoría de probabilidades dice que la distribución de una suma de variables aleatorias independientes bajo ciertas condiciones tiende hacia una distribución gaussiana. Sea el vector \mathbf{x} una distribución idéntica de datos de una mezcla de componentes independientes según la ecuación 1.4. Para estimar una de las componentes independientes podemos considerar una combinación lineal de x_i para la ecuación 1.4. Sea $y = \mathbf{w}^T \mathbf{x} = \sum_i w_i x_i$, donde \mathbf{w} es el vector que se quiere encontrar. Si \mathbf{w} fuera una de las filas inversas de la matriz \mathbf{A} esta combinación lineal sería igual a una de las componentes independientes.

Sea $\mathbf{z} = \mathbf{A}^T \mathbf{w}$ tenemos $y = \mathbf{w}^T \mathbf{x} = \mathbf{w}^T \mathbf{A} \mathbf{s} = \mathbf{z}^T \mathbf{s}$, donde y es una combinación lineal de s_i con los pesos cerca de z_i . La suma de dos variables aleatorias, uniformes e independientes es más gaussiana que la de las variables originales, $\mathbf{z}^T \mathbf{s}$ es más gaussiana que s_i y cuando el factor s_i se iguala a uno no presenta gaussianidad. Es por esta razón que se puede tomar \mathbf{w} que un vector que maximiza la no gaussianidad de $\mathbf{w}^T \mathbf{x}$. Este vector corresponde

1.3. PRINCIPIOS MATEMÁTICOS PARA LA VALORACIÓN DE ICA17

a un \mathbf{z} el cual tiene solamente una componente distinta de cero; lo que hace $\mathbf{w}^T \mathbf{x} = \mathbf{z}^T \mathbf{s}$ y maximizando la no gaussianidad de $\mathbf{w}^T \mathbf{x}$ se puede encontrar una componente independiente. Encontrando todos estos máximos locales de la señal se pueden encontrar las demás componentes independientes.

1.3.6. Minimización de la información mutua

La información mutua es una medida de dependencia entre dos variables aleatorias, ésta se empleada como una medida de la cantidad de información que las variables de un vector aleatorio tienen sobre el resto de las variables del conjunto. Referenciándonos en el concepto de la entropía, la información mutua I entre las variables aleatorias (escalares) de $y_i, i = 1 \dots m$ es:

$$I(y_1, y_2, y_3, \dots, y_m) = \sum_{i=1}^m H(y_i) - H(y) \quad (1.18)$$

La información mutua se presenta siempre en valores positivos y es cero si y solamente si las variables son independientes. Con este tipo de información se puede medir la dependencia estadística de las variables aleatorias.

Una característica importante de la información mutua es que puede tener una transformación lineal inversible $\mathbf{y} = \mathbf{W}\mathbf{x}$:

$$I(y_1, y_2, y_3, \dots, y_m) = \sum H(y_i) - H(x) - \log |\det \mathbf{W}| \quad (1.19)$$

1.3.7. Valoración de la máxima probabilidad

Partiendo de un conjunto finito de medidas, la teoría de la aproximación tiene como objetivo aproximar una serie de parámetros poblacionales mediante estimadores. Suponiendo $\mathbf{W} = (\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_n)^T$ la matriz \mathbf{A}^{-1} podemos obtener [69]:

$$L = \sum_{t=1}^T \sum_{i=1}^n \log f_i(\mathbf{w}_i^T \mathbf{x}(t)) + T \log |\det \mathbf{W}| \quad (1.20)$$

donde f_i determina las funciones de densidad de s_i . A nivel general, para cualquier vector aleatorio \mathbf{x} con densidad de probabilidad p_x y para cualquier matriz \mathbf{W} , la densidad de $\mathbf{y} = \mathbf{W}\mathbf{x}$ es dada por $p_x(\mathbf{W}\mathbf{x})|\det \mathbf{W}|$. De la siguiente ecuación podemos ver la conexión entre la probabilidad y la información mutua:

$$\frac{1}{T} E\{L\} = \sum_{i=1}^n E\{\log f_i(\mathbf{w}_i^T \mathbf{x})\} + \log |\det \mathbf{W}| \quad (1.21)$$

Si f_i fuera igual a las distribuciones reales de $\mathbf{w}_i^T \mathbf{x}$, el primer término sería igual a $-\sum_i H(\mathbf{w}_i^T \mathbf{x})$

1.3.8. Momentos y Cumulantes

Es posible obtener una función genérica para la creación de momentos de cualquier orden mediante la transformada de Fourier continua de la función de densidad de probabilidad de x . Esta función se denomina función

1.3. PRINCIPIOS MATEMÁTICOS PARA LA VALORACIÓN DE ICA19

generadora de momentos:

$$\phi(w) = E\{\exp(iwx)\} = \int_{-\infty}^{\infty} \exp(iwx)px(x)dx \quad (1.22)$$

donde $i = \sqrt{-1}$ y w corresponde a la variable x después de la transformación.

Los cumulantes corresponden a otro tipo de estadísticos de alto orden derivados a partir del logaritmo neperiano de la función generadora de momentos, de la misma forma que los momentos se pueden obtener de dicha función:

$$\gamma(w) = \ln(\phi(w)) \quad (1.23)$$

Para una variable aleatoria x de media 0, los primeros 4 cumulantes son:

$$\begin{aligned} k_1 &= 0 \\ k_2 &= E\{x^2\} \\ k_3 &= E\{x^3\} \\ k_4 &= E\{x^4\} - 3[E\{x^2\}]^2 \end{aligned}$$

Los tres primeros cumulantes coinciden con sus respectivos momentos, mientras que el cuarto corresponde a la kurtosis la cual definiremos más adelante.

En el caso multivariado, es decir, cuando tratamos un vector x de variables aleatorias, extrapolamos las fórmulas de los cumulantes, obteniendo los

cumulantes cruzados. El segundo y tercer cumulante cruzado coincide con sus respectivos momentos cruzados. La expresión del cumulante cruzado de cuarto orden cambia respecto a su correspondiente momento cruzado:

$$\begin{aligned} cum(x_i, x_j) &= Ex_ix_j \\ cum(x_i, x_j, x_k) &= Ex_ix_jx_k \\ cum(x_i, x_j, x_k, x_l) &= Ex_ix_jx_kx_l - Ex_ix_jEx_kx_l - Ex_ix_kEx_jx_l - Ex_ix_lEx_jx_k \end{aligned}$$

Obviamente, al poder obtener los cumulantes a partir de los momentos, ambos contienen la misma información estadística. Sin embargo, es preferible tratar con cumulantes puesto que presentan una serie de propiedades deseables que no se encuentran en los momentos, entre ellas la de que todos los cumulantes de orden mayor que tres de un vector aleatorio x de distribución gaussiana son nulos.

1.3.9. Métodos basados en medidas estadísticas de alto orden

Los parámetros estadísticos de alto orden pueden definir funciones de contraste que resultan ser aproximaciones a las obtenidas a partir de los enfoques de máxima verosimilitud (ML). La información de alto orden puede ser expresada por medio de los cumulantes, en donde un conjunto de señales

y_1, \dots, y_p) son estadísticamente independientes si y sólo si sus cumulantes cruzados se anulan. Resulta común considerar las fuentes con funciones de densidad de probabilidad simétricas centradas en sus medias $E\{y_i\} = 0$, de aquí se puede ver que los momentos de orden impar son nulos, por ello se pueden tener las condiciones:

Cumulantes cruzados de orden 2 nulos: la matriz de correlación es diagonal. Esta condición equivale a imponer $p(p+1)/2$ restricciones ya que la matriz de correlación es pxp y simétrica.

Cumulantes cruzados de orden 4 nulos: esto es necesario para poder completar la información y poder tener hasta pxp relaciones con objeto de calcular los elementos de la matriz de reconstrucción.

Las señales con función de distribución gaussiana tienen todos los cumulantes de orden superior a 2 nulos, por lo que no se pueden separar este tipo de señales utilizando un enfoque estadístico (salvo en el caso de que tan sólo una de las fuentes sea gaussiana).

1.4. Medidas de no gaussianidad

Kurtosis. La kurtosis se calcula en relación al momento de cuarto orden.

Matemáticamente la kurtosis se define como:

$$kurt(y) = E\{y^4\} - 3(E\{y^2\})^2 \quad (1.24)$$

Asumiendo que y tiene varianza unitaria podríamos escribir la anterior expresión como:

$$kurt(y) = E\{y^4\} - 3 \quad (1.25)$$

Para variables aleatorias gaussianas el cuarto momento es cero; mientras que para variables aleatorias no gaussianas el valor del kurtosis es diferente de cero. Esta magnitud de cercanía al cero es lo que hace a la kurtosis interesante. Igualmente, ésta magnitud puede presentar un valor positivo o negativo. Para variables aleatorias con kurtosis negativo se conocen como subgaussianas, mientras que variables aleatorias con kurtosis positivo se conocen como supergaussianas.

Negentropía. La negentropía es una medida que se basa en la cantidad información teórica de entropía. En teoría de la información la entropía es un concepto básico y para una variable aleatoria se puede interpretar como el grado de información que la observación de la variable dada [26] [68].

La entropía H se define para una variable aleatoria Y como:

$$H(Y) = - \sum P(Y = a_i) \log P(Y = a_i) \quad (1.26)$$

donde todos los posibles valores de Y están dados por a_i .

La entropía diferencial de un vector \mathbf{y} con densidad $f(\mathbf{y})$ se define como:

$$H(\mathbf{y}) = - \int f(\mathbf{y}) \log f(\mathbf{y}) d\mathbf{y} \quad (1.27)$$

El concepto de negentropía se define a partir del de entropía para determinar una medida no negativa que para variables gaussianas es cero:

$$J(\mathbf{y}) = H(\mathbf{y}_{gauss}) - H(\mathbf{y}) \quad (1.28)$$

donde \mathbf{y}_{gauss} es una variable gaussiana aleatoria de la misma matriz de covarianza que \mathbf{y} . De lo anteriormente expuesto se puede observar que la negentropía siempre son valores positivos y es cero solamente si \mathbf{y} presenta una distribución gaussiana. La negentropía también tiene la propiedad de ser invariante frente a transformaciones lineales [26] [68].

Clasicamente la negentropía puede aproximarse utilizando momentos de alto orden [47]:

$$J(y) \approx \frac{1}{12} E\{y^3\}^2 + \frac{1}{48} kurt(y)^2 \quad (1.29)$$

donde y se asume con varianza unitaria; pero muchas veces las aproximaciones de estos métodos pueden resultar algo limitadas. Para resolver

este problema se recurre al principio de la máxima entropía:

$$J(\mathbf{y}) \approx \sum_{i=1}^p k_i [E\{G_i(\mathbf{y})\} - E\{G_i(v)\}]^2 \quad (1.30)$$

donde k_i pueden ser constantes positivas y v es una variable gaussiana con varianza unitaria.

Para funciones no cuadráticas unitarias G la aproximación queda:

$$J(\mathbf{y}) \propto [E\{G(\mathbf{y})\} - E\{G(v)\}]^2 \quad (1.31)$$

1.5. ICA y el pre-procesamiento previo de los datos

El propósito fundamental del pre-procesamiento de datos, independientemente de la rama en la que se quiera investigar, se convierte en la manipular de la información para transformar y preparar los datos para una correcta investigación. La importancia en el pre-procesado se encuentra sustentada en que datos reales pueden contener patrones impuros (ruido, datos incompletos, etc), que pueden conducir a resultados poco útiles.

1.5.1. Substracción de la media

Este proceso previo se hace solamente para simplificar los algoritmos usados por la técnica ICA. De acuerdo con la ecuación 1.4, por medio de los algoritmos ICA puede ser estimada la matriz de mezcla \mathbf{A} con datos centrados, una vez llevado a cabo este proceso podemos agregar el vector s de nuevo a las estimaciones centradas de s ; donde el vector s es dado por $\mathbf{A}^{-1}m$.

1.5.2. Blanqueamiento de datos

Este pre-procesamiento se lleva a cabo antes de uso de cualquiera de los algoritmos de ICA y después de haber centralizado los datos. Los vectores aleatorios blanqueados tienen como característica que para todas las componentes la media es cero y la matriz de covarianzas es la matriz identidad I .

Un vector lineal x de observaciones puede ser transformado para obtener un nuevo vector denominado $\tilde{\mathbf{x}}$ el cual está blanqueado lo que significa que sus componentes son de variación unidad y sin correlación. Esto es, la matriz de covariación $\tilde{\mathbf{x}}$ igual a la matriz de identidad:

$$E\{\tilde{\mathbf{x}}\tilde{\mathbf{x}}^T\} = I \quad (1.32)$$

Un método muy utilizado para el blanqueamiento de las observaciones es

la descomposición en los valores propios de la matriz:

$$E\{xx^T\} = EDE^T \quad (1.33)$$

donde \mathbf{E} es una matriz ortogonal de los vectores propios de $E\{xx^T\}$ y D es la matriz diagonal con los valores propios, $D = \text{diag}(d_1, d_2, \dots, d_n)$

Blanqueando se obtiene:

$$\tilde{\mathbf{x}} = ED^{-1/2}E^T x \quad (1.34)$$

donde la matriz $\mathbf{D}^{-1/2}$ es obtenida como $\mathbf{D}^{-1/2} = \text{diag}(d_1^{-1/2}, d_2^{-1/2}, \dots, d_n^{-1/2})$.
con lo que se hace más fácil $E\{\tilde{\mathbf{x}}\tilde{\mathbf{x}}^T\} = I$

Trasformando lo anterior dentro de una nueva matriz $\tilde{\mathbf{A}}$ se puede tener de 1.4:

$$\tilde{\mathbf{x}} = ED^{-1/2}E^T A s = \tilde{\mathbf{A}} s \quad (1.35)$$

La nueva ventaja que presenta el blanqueamiento es que la $\tilde{\mathbf{A}}$ esta blanqueada. Esto puede verse en:

$$E\{\tilde{\mathbf{x}}\tilde{\mathbf{x}}^T\} = \tilde{\mathbf{A}} E s s^T \tilde{\mathbf{A}}^T = \tilde{\mathbf{A}} \tilde{\mathbf{A}}^T = I \quad (1.36)$$

De aquí puede verse como el blanqueado reduce el parámetro de señales a estimar. Resumiendo podemos decir que para un vector observado x , se dice que éste esta blanqueado si $m_x = 0$, $Cov_x = Corr_x = I$

1.6. ICA vs PCA

El Análisis de Componentes Principales y el Análisis de Componentes Independientes son dos técnicas las cuales se basan en métodos estadísticos que presenta entre sí grandes diferencias. mientras que PCA es una técnicas de proyección, ICA se presenta más como un nuevo método que produce vectores espacial localizados y estadístico independientes.

1.6.1. PCA (Análisis de Componentes Principales)

A nivel estadístico cuando se desea obtener una transformación lineal de datos dentro de un análisis multivariante se suele utilizar el Análisis de Componentes Principales (PCA); el cálculo de estas componentes principales se lleva a cabo por medio de la matriz de varianza y covarianza. Ésta técnica suele usarse orientada a dos enfoques [89]:

En el sentido de maximizar la varianza, PCA busca direcciones incorreladas con el fin de encontrar nuevas variables no correlacionadas que sean función lineal de las variables originales con el fin de poder determinar las componentes principales.

PCA busca sistemas de referencia ortogonal del subespacio de modo que la diferencia entre las observaciones y sus proyecciones tengan el menor error cuadrático medio.

El objetivo de PCA es eliminar la redundancia de los datos encontrando un conjunto de variables menor. Para realizar esta transformación se impone la condición de que las nuevas señales estén incorreladas. Es por esta razón que para muchas aplicaciones de separación de fuentes PCA es usado como una forma de pre-procesamiento de datos en donde además de lo anteriormente descrito se pueden tener también como ventaja como filtración de ruido gaussiano.

Existen muchos métodos que permiten calcular las componentes principales. Tal y como se menciona con anterioridad matemáticamente esto corresponde a calcular los vectores propios y los valores propios de la matriz de covarianza del vector aleatorio:

$$\mathbf{D} = \mathbf{V}^T \text{cov}(\mathbf{x}) \mathbf{V} \quad (1.37)$$

donde \mathbf{V} es la matriz con los vectores propios de la matriz de covarianza de \mathbf{x} como columnas y \mathbf{D} es una matriz diagonal con los valores propios correspondientes en la diagonal. La ecuación para la matriz de PCA \mathbf{W} es:

$$\mathbf{W} = \mathbf{D}^{-\frac{1}{2}} \mathbf{V}^T \quad (1.38)$$

La prueba para demostrar esto se puede obtener de una manera muy simple:

$$\begin{aligned} \text{cov}(\mathbf{W}\mathbf{x}) &= E(\mathbf{W}\mathbf{x}\mathbf{x}^T\mathbf{W}^T) = \mathbf{W}\text{cov}(\mathbf{x})\mathbf{W} \\ &= \mathbf{D}^{-\frac{1}{2}}\mathbf{V}^T\text{cov}(\mathbf{x})\mathbf{V}\mathbf{D}^{-\frac{1}{2}} = \mathbf{D}^{-\frac{1}{2}}\mathbf{D}\mathbf{D}^{-\frac{1}{2}} \end{aligned}$$

= II

Otra posibilidad, es la optimización con la siguiente función de coste:

$$J(\vec{\omega}) = \left\langle \left(\mathbf{x}^T \frac{\vec{\omega}}{\|\vec{\omega}\|} \right)^2 \right\rangle = \frac{\vec{\omega}^T \langle \mathbf{x}\mathbf{x}^T \rangle \vec{\omega}}{\|\vec{\omega}\|^2}$$

donde \mathbf{x} es el vector de los datos ya centralizados y $\vec{\omega}$ los nuevos ejes del sistema de coordenadas. La reducción al mínimo de esta función de coste significa:

$$\vec{\nabla} J(\vec{\omega}) \stackrel{!}{=} 0 \quad (1.39)$$

Con lo que se obtiene:

$$\frac{\langle \mathbf{x}\mathbf{x}^T \rangle \vec{\omega} \|\vec{\omega}\|^2 - (\vec{\omega}^T \langle \mathbf{x}\mathbf{x}^T \rangle \vec{\omega}) \vec{\omega}}{\|\vec{\omega}\|^4} \stackrel{!}{=} 0 \quad (1.40)$$

y

$$\langle \mathbf{x}\mathbf{x}^T \rangle \vec{\omega} = \frac{(\vec{\omega}^T \langle \mathbf{x}\mathbf{x}^T \rangle \vec{\omega})}{\|\vec{\omega}\|^2} \vec{\omega} = \lambda \vec{\omega} \quad (1.41)$$

con

$$\lambda = \frac{(\vec{\omega}^T \langle \mathbf{x}\mathbf{x}^T \rangle \vec{\omega})}{\|\vec{\omega}\|^2}.$$

Es también posible utilizar una red neuronal para PCA: Sea \mathbf{x}' el vector de entrada, \vec{w} los pesos y \vec{v} el vector de salida. La regla del aprendizaje Hebbiano es:

$$\vec{v} = \vec{w}^T \mathbf{x}' \quad (1.42)$$

con

$$\Delta w_i = \eta v_i x'_i \quad (1.43)$$

Es importante conocer que resulta necesario la normalización de los pesos. Una forma para hacer esto es la normalización propuesta por Oja con el paso de tiempo k :

$$\vec{w}^{k+1} = \frac{\vec{w}^k + \eta \vec{v}^k \mathbf{x}'^k}{\|\vec{w}^k + \eta \vec{v}^k \mathbf{x}'^k\|} \approx w^4 + \eta v^4 (x'^4 - v^4 w^4) + O(\eta^2) \quad (1.44)$$

De lo anterior se obtiene:

$$\Delta w_i = \eta v_i x_i - \eta v^2 w_i \quad (1.45)$$

\vec{w} forma entonces un vector de valores propios $\langle \mathbf{x}\mathbf{x}^T \rangle$ con λ_{\max} .

1.6.2. ICA (Análisis de Componentes Independientes)

A lo largo del desarrollo de este capítulo hemos venido hablando de la técnica ICA. Normalmente ICA es usado para extraer componentes de diferentes clases de datos, como vídeo, imágenes, etc. En el espacio de las transformaciones obtenidas por ICA se pueden formar asociaciones fácilmente apreciables en el nuevo espacio; pero que no son apreciables en el espacio original.

1.6.3. Comparación ICA y PCA

Tal cual como hemos expuesto, mientras PCA es utilizado como un preprocesamiento para eliminar la información redundante, el objetivo de ICA no es el de reducir la dimensión, obteniendo tantas componentes independientes como variables observadas. El principal uso de ICA es el de encontrar las fuentes originales a partir de unas observaciones (mezclas) lineales.

Mientras que PCA pretende encontrar aquellas combinaciones que proporcionen más información medida en términos de la varianza, ICA obtiene combinaciones independientes, esto es, menor entropía, menor información. La figura 1.5 demuestra este efecto. Muestra la uniformidad en fuentes distribuidas s_1 y s_2 , mezclado por la matriz de mezcla

$$\mathbf{A} = \begin{pmatrix} 0,90 & 0,80 \\ 0,04 & 0,30 \end{pmatrix}.$$

Como puede verse los ejes de PCA y de ICA no señalan en la misma dirección, de hecho después de aplicar PCA a los datos las fuentes serán más dependientes que antes. Igualmente con ICA no tomará las componentes resultantes basadas en la contribución a la variación, las componentes resultantes contribuirán de forma absoluta en mayor o menor cantidad de variación a las mezclas.

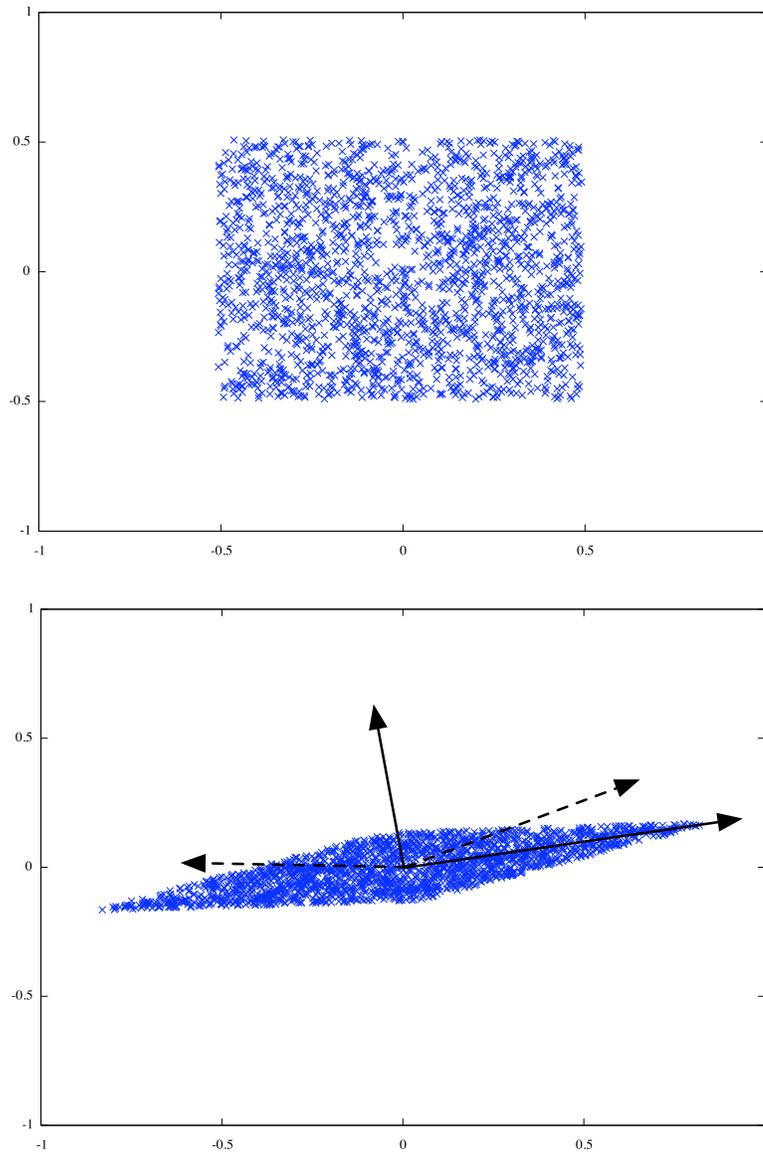


Figura 1.5: Arriba se muestran las fuentes antes de aplicar la matriz de mezcla. Abajo se muestra las mezclas obtenidas y los ejes de PCA (sólido) y de ICA (punteado).

1.7. Modelos de mezcla lineales

Se puede asumir los datos como un conjunto de variables recibidas de manera conjunta. Sea n el número de variables y T el número de observaciones. Los datos disponibles como $x_i(t)$ donde los índices toman los valores $i = 1, \dots, n$ y $t = 1, \dots, T$. Suponemos, también, que el número de observaciones $x_i(t)$ es igual al número de fuentes $s_i(t)$.

1.7.1. Modelos lineal de mezcla instantáneo

Este modelo supone que el valor de las observaciones $x_i(t)$ en un instante dado es una función lineal de los valores de las fuentes originales $s_i(t)$ en ese mismo instante de tiempo y no influyen en la mezcla en ningún caso los valores de las fuentes en instantes de tiempo anteriores. Una función lineal sobre un conjunto de variables se reduce a un conjunto de coeficientes reales que multiplican a los datos originales. Matemáticamente tenemos:

$$x_i(t) = \sum_{j=1}^n a_{ij}s_j(t) \quad (1.46)$$

donde $i = 1, \dots, n$.

Utilizando vectores de variables:

$$x(t) = \mathbf{A}s(t) \quad (1.47)$$

donde \mathbf{A} es la matriz de mezcla y por medio de ésta se realiza la combi-

nación lineal de las fuentes s .

De acuerdo a lo anterior, en el caso de tener la matriz \mathbf{W} de separación cuya inversa es similar a la matriz de mezcla \mathbf{A} y de mismas dimensiones, puede ser resuelto el problema de separación de señales, multiplicando \mathbf{A} por \mathbf{W} se podría obtener una reconstrucción o estimación $\mathbf{y}(t)$ de las fuentes originales. De esta forma, la ecuación del modelo para la separación es:

$$\mathbf{y}(t) = \mathbf{W}\mathbf{x}(t) \quad (1.48)$$

La separación ciega de señales en medios lineales e instantáneos se reduce según la ecuación 1.47 a la obtención de los coeficientes de la matriz de separación \mathbf{W} , teniendo en cuenta que los resultados se presentaran con las ambigüedades mencionadas en la sección 1.2.1 respecto a la matriz original:

$$\mathbf{W}\mathbf{A} = \Lambda\mathbf{P} \quad (1.49)$$

donde Λ es una matriz diagonal no singular y \mathbf{P} es una matriz de permutación (matriz con un único 1 por fila y columna).

1.7.2. Modelos con distinto número de mezclas y de fuentes

En aplicaciones de la vida real el número de las fuentes se desconoce. Muchos de los algoritmos de separación de señales resultan muy eficientes

con el número de fuentes igual al número de sensores. Si denominamos como s al número de fuentes y q al número de observaciones o de sensores, cuando no se presenta igualdad entre estas variables pueden existir dos casos:

Menos observaciones que fuentes ($q < s$): se denomina separación ciega de señales sobrecompleta (overcomplete blind source separation). En principio el problema no tiene solución ya que las q observaciones no contienen información suficiente para separar exactamente las s fuentes. Se ha estudiado teóricamente este problema consiguiendo separar las s fuentes en q grupos distintos y se obtuvieron las fuentes casi puras o combinaciones lineales de ellas. Se pueden encontrar algunos resultados experimentales con el algoritmo PCA-no lineal. Asimismo, según recientes investigaciones con métodos geométricos pueden detectarse las s componentes independientes [86] [85].

Más observaciones que fuentes ($q > s$): para este caso se recurre en primera instancia al blanqueo que transformará las q observaciones en las s señales incorreladas con mejor relación señal/ruido. Las s señales proyectadas sobre los vectores propios correspondientes a las s valores propios mayores resultan ser las más adecuadas (las $q - s$ señales resultantes pueden considerarse ruido).

1.7.3. Mezclas no estacionarias

En muchas ocasiones puede ocurrir que las características del medio material donde se produce la mezcla o los sensores varíen con el tiempo. Para este caso los elementos de la matriz de mezcla \mathbf{A} deben de cambiar con el tiempo $\mathbf{A}(t)$. Igualmente se puede presentar cambios en la densidad de probabilidad de las fuentes lo que complica la resolución de la separación. Por este motivo, es importante en la práctica tener algoritmos de separación que puedan adaptarse a cambios de los coeficientes de la mezcla y de las fuentes. Para este caso, las redes neuronales artificiales resultan ser muy eficientes [33]; es posible conseguir que los elementos de matriz estimada se auto-adapten en el tiempo a los cambios de la matriz de mezcla, aplicando constantemente la regla de aprendizaje y seleccionando adecuadamente el parámetro de ganancia de aprendizaje en el tiempo.

1.7.4. Conocimiento de las fuentes

Toda información sobre las fuentes conlleva a simplificaciones en los algoritmos de separación de señales. En cuanto a la información que se puede tener sobre las fuentes, esta puede ser:

Conocimiento sobre las distribuciones de probabilidad de las fuentes [14] [13] [71].

Conocimiento sobre la naturaleza discreta de los datos (binarios o multivaluados), en cuyo caso se pueden utilizar algoritmos muy eficientes [73]

[75] [74].

En general saber también que las fuentes estén acotadas y conocer los rangos máximo y mínimo entre los cuales pueden estar limitadas en amplitud [4] [3].

1.8. Modelos de mezcla no lineales

Generalizando el modelo de mezcla lineal instantánea, se puede suponer el valor de las observaciones en un instante como una función de cualquier tipo que depende solamente de los valores de las fuentes originales en ese mismo instante de tiempo, de esta forma el modelo de mezcla será no lineal e instantáneo. Así, se puede generalizar 1.47 como:

$$\mathbf{x}(t) = \mathbf{F}(\mathbf{s}(t)) \quad (1.50)$$

donde $\mathbf{F} = [f_1, f_2, \dots, f_n]^T$ son funciones no lineales de mezcla que transforman las señales originales que no se conocen en las mezclas observadas.

La separación ciega de señales no lineales consiste en encontrar las funciones $\mathbf{G} = [g_1, g_2, \dots, g_n]^T$ que al aplicarse sobre las mezclas observadas, entregan un conjunto de señales \mathbf{y} equivalentes en su forma de onda a las señales originales. Matemáticamente esto puede ser expresado como:

$$\mathbf{y}(t) = \mathbf{G}(\mathbf{x}(t)) \quad (1.51)$$

Para el caso general de mezclas no lineales 1.50 siempre existen muchas soluciones diferentes que, a partir de un conjunto de variables dependientes, proporcionan un conjunto de variables independientes. Pero al mismo tiempo se presenta el problema de que estas soluciones pueden no tener ninguna relación con las señales originales buscadas.

1.8.1. El modelo de mezcla post no lineal

En este tipo de modelo de mezcla se asume una mezcla en dos fases para cada una de las fuentes desconocidas.

1. Mezcla lineal: donde las fuentes se mezclan de acuerdo al modelo lineal, obteniendo un conjunto de mezclas desconocidas.

$$\mathbf{z}(t) = \mathbf{A}(\mathbf{s}(t)) \quad (1.52)$$

2. Mezcla no lineal: donde se aplica una función no lineal f_i con el fin de obtener observaciones o mezclas:

$$x_i(t) = f_i z_i(t) \quad (1.53)$$

Escribiendo lo anterior en forma vectorial se tiene:

$$\mathbf{x} = \mathbf{F}(\mathbf{A}\mathbf{s}) \quad (1.54)$$

donde \mathbf{A} es la matriz de mezcla de coeficientes lineales y $\mathbf{F} = (f_1, f_2, \dots, f_n)$ las distorsiones no lineales.

El problema que presenta este tipo de mezcla no lineal es que al querer encontrar las componentes independientes presenta mucha más complejidad ya que el espacio de búsqueda es mucho mayor, lo que conlleva a que se complique encontrar soluciones satisfactorias de acuerdo a la dimensionalidad del problema.

1.9. Soluciones al problema BSS basados en la implementación de algoritmos

1.9.1. Métodos de eliminación de ruido

Este tipo de método considera [55]:

Métodos basados en una observación. - Supresión espectral: se basa en

la estimación espectral de la señal: sus hipótesis principales son: señal y ruido no correlacionados, ruido estacionario en el tiempo, gran importancia del modulo del espectro ya que es quien interviene en la supresión de ruido y aplicación a señales formadas por voz y ruido aditivo. (sustracción espectral, descomposición espectral.

- Reducción de ruido basado en el modelo de producción de voz: modelo que resulta un filtro “todo polos”, con parámetros y variables en el tiempo. El ruido es eliminado por medio de filtros AR. Como factores

de desventaja en este modelo de puede resaltar la necesidad del conocimiento previo sobre la señal de voz y la complejidad en el calculo de parámetros.

- Supresión de ruido con una red de neuronas multicapa: conexión de red que se obtiene por medio del aprendizaje de un conjunto de muestras tanto de la señal que esta afectada por el ruido, como la señal original. La desventaja de este modelo esta dad en la necesidad de conocer previamente la señal, además de la complejidad de su calculo.

Métodos basados en dos observaciones. - Aproximación basada en máxi-

ma similitud. Usada para estimar para metros de la voz eliminando el ruido. La aplicación de estos algoritmos requiere conocer información previa de la señal de voz, entre lo que se puede destacar la necesidad de conocer si el ruido es aleatorio y gaussiano. Los sistemas lineales se modelan como filtros de respuesta impulsiva finita. La desventaja esta en su alto costo y su aplicación solo a señales de voz, debido a la necesidad de conocer los parámetros espectrales.

- Eliminación de ruido en presencia de una referencia de ruido: método que implica conocer previamente la fuente de ruido o medirla muy bien. El ruido y la señal de referencia están corraelacionadas, mientras que en las señales originales no.

- Estimación adaptativa de dos señales con una estructura recursiva: consisten en utilizar filtros adaptativos con el fin de decorrelacionar las señales; a pesar de que esta correlación en la salida no es garantía de la

separación de las señales no se necesita de hipótesis de referencias de ruido.

- Reducción de ruido por medio de funciones de coherencia. Enfoque estudiado por medio de etapas, en donde en primer lugar se genera una referencia de ruido y en segundo lugar se usa este como identificación entre ruidos prefiltrados. Presenta dos hipótesis: Ruidos correlacionados y para señales que vienen de una misma fuente, el modulo de la función de coherencia es igual a 1.

Métodos basados en más de dos observaciones. Método con red de micrófonos. Utilización de micrófonos para captar señales de audio.

- Red de micrófonos autodirectivos: con un número mínimo de micrófonos y a distancias cortas se logra una separación de señales aceptable.
- Red adaptativa de micrófonos: requiere de cálculos complejos y de información previa de señales de voz, ruido y entorno

1.9.2. Métodos basados en redes neuronales

Utilización de redes neuronales artificiales en donde los elementos del separador se denominan pesos sinápticos y la matriz de parámetros se conoce como matriz de pesos. Se recurre a los algoritmos adaptativos de aprendizaje para la adaptación de los pesos. Otros algoritmos de aprendizaje de redes neuronales son: el método de la máxima entropía que utiliza el método del ascenso más abrupto y el análisis de compo-

entes principales (PCA), que usa un algoritmo hebbiano generalizado. Más información puede ser encontrada en [70] [36] [48].

Red neuronal recursiva de Cichocki y Amari , en donde se proponen algoritmos de aprendizaje robustos, no supervisados y que resultan eficientes al usarlos en problemas mal condicionados o escalados, su objetivo es conseguir una convergencia de los pesos partiendo de una eliminación de los momentos de orden superior con el fin de obtener una independencia estadística [21] [5].

1.9.3. Métodos basados en la utilización estadística

Basado en las propiedades de las distribuciones de probabilidad de las fuentes y de las observaciones, lo cual se caracteriza tanto por cumulantes como por distribuciones respectivas. Debe de cumplir la hipótesis de que las señales originales $s(t)$ son estadísticamente independientes, así las señales recuperadas cumplirán con criterios estadísticos o de teoría estadística de la información.

- Métodos basados en estadísticas de alto orden: usado para definir funciones de contraste que sean aproximaciones obtenidas por medio del enfoque de máxima verosimilitud. Esta función se puede expresar por medio de cumulantes.
- Estimación de la máxima verosimilitud: función que resulta ser la densidad

de probabilidad de un conjunto de datos de un modelo dado, vista como función de los parámetros desconocidos. Su desarrollo hace necesario definir una función logarítmica de máxima verosimilitud para después optimizarla con respecto al vector de parámetros.

Métodos basados en el Análisis de Componentes Principales (PCA)

Permite proyectar en espacios de características ortogonales un espacio de datos, con el fin de que la varianza de las proyecciones sea mayor esto se logra teniendo como condición la decorrelación de las nuevas señales. Busca las componentes principales de las observaciones que consiste en los autovalores propios de la matriz de covarianza de dichas observaciones, para ello impone la condición de independencia estadística de segundo orden y que la base de vectores del espacio de características sea ortogonal.

- Blanqueado espacial: tiene como objetivo someter a los vectores $e(t)$ a una transformación lineal V para obtener nuevos vectores $x(t)$. El utilizar PCA para el blanqueado permite que se pueda comprimir información para optimizar el error cuadrado medio, la transformación de un espacio de datos a un espacio de características permite mantener la información intrínseca, se puede filtrar el ruido gaussiano y realizar las separaciones por medio del procedimiento de PCA no lineal y decorrelacionar señales.

- PCA no lineal: propuesto por Oja, puede ser implementado mediante redes neuronales artificiales a través de la regla de aprendizaje.

Métodos basados en el Análisis de Componentes Independientes (ICA)

Introducidos por Jutten y Herault, tienen como objetivo buscar la máxima independencia entre las componentes que están conformando las observaciones, la independencia estadística permite componentes del vector de salidas individuales reconstruidas sin la condición de ortogonalidad [55].

1.9.4. Métodos basados en el principio de la Máxima Entropía

Las técnicas del procesamiento de imágenes son aplicadas cuando se hace necesario realzar o modificar una imagen para mejorar su apariencia o para destacar algún aspecto de la información contenida en la misma, o cuando se requiere medir, contrastar o clasificar algún elemento contenido en la misma. Igualmente estas técnicas se usan al combinar imágenes o trozos de las mismas o reorganizar su contenido.

1.10. Medidas de prestaciones

Con el fin de saber si la separación de las fuentes se ha realizado adecuadamente, se recurre a unas medidas de prestaciones; pero en BSS no se tiene información ni de las fuentes originales ni de cómo se han mezclado, luego estas medidas no tienen sentido ya que idealmente se debería poder

comparar las señales reconstruidas, con las fuentes originales. Con el fin de poder comprobar si los resultados son correctos, se llevan a cabo procesos completos de simulaciones, para poder disponer de las señales originales, las cuales se mezclan para luego aplicar los algoritmos de separación de fuentes y obtener las señales estimadas. Con estas últimas y las originales, se realizan las medidas de prestaciones que indican la calidad de la separación. A continuación enunciaremos los índices más comúnmente usados, para más información puede referirse:

Índice de Prestaciones de Amari: Denominado $E(\mathbf{C})$, se puede determinar la calidad de la separación de fuentes con un único valor numérico. El índice de prestaciones de Amari se define como [21] [5]:

$$E(\mathbf{C}) = \sum_{i=1}^p \left(\left(\sum_{j=1}^p \frac{|c_{ij}|}{\max_k |c_{ij}|} \right) - 1 \right) + \sum_{j=1}^p \left(\left(\sum_{i=1}^p \frac{|c_{ij}|}{\max_k |c_{ij}|} \right) - 1 \right) \quad (1.55)$$

donde $\mathbf{C} = (c_{ij})$ es la matriz de prestaciones, a menor valor de índice se presentan mejores prestaciones. La separación es perfecta cuando $E(\mathbf{C}) = 0$

Diafonía: La diafonía (crosstalk) es un estimador que se define para dar información sobre el nivel de diferencia entre dos señales. Cuanto menor es la diafonía, mayor será la similitud entre dichas señales. Es una

medida inversa a la relación señal/ruido. El crosstalk se define como:

$$Cross_i = 10 \log \frac{\langle x_j - s_j \rangle^2}{\langle s_j \rangle^2} \quad (1.56)$$

Error cuadrático medio (MSE) entre los elementos de \mathbf{A} y \mathbf{W} : se define cuando se presenta un error cuadrático medio entre los elementos de \mathbf{A} y \mathbf{W}) haciéndose necesario primero calcular las diferencias entre los elementos de $\mathbf{w}_{ij} - \mathbf{a}_{ij}$ ($i, j = 1, \dots, p$):

$$MSE(t) = \frac{1}{p^2} \sum_{i=1}^p \sum_{j=1}^p (\mathbf{a}_{ij}^*(t) - \mathbf{w}_{ij}^*(t))^2 \quad (1.57)$$

donde p es el número de fuentes originales.

Crosstalk error: se define como:

$$Cross_e(G) = \sum_i \left(\sum_j \frac{|g_{ik}|}{\max_k |g_{ik}|} - 1 \right) + \sum_j \left(\sum_i \frac{|g_{ik}|}{\max_k |g_{ik}|} - 1 \right) \quad (1.58)$$

Rechazo o interferencia de una fuente s sobre la q : se define como:

$$Rej_{sq} = \langle g_{sq}^2 \rangle \quad (1.59)$$

Indice global (GlobInd): siendo s el número de fuentes

$$\rho(G(k)) = 100 \sum_i \left(\max_i \left\{ \frac{|g_{ik}|}{\sum |g_{ik}|} \right\} - \frac{1}{p} \right) \quad (1.60)$$

Otras medidas de prestaciones: existen también otras medidas de prestaciones como: la Norma de Error (ErrN), Razón de Error (SER).

1.11. Aplicaciones

A nivel práctico existen infinidad de campos donde la técnica del análisis de componentes independientes puede ser aplicado a la separación de fuentes.

A continuación se enumeran algunas de estas aplicaciones:

Reducción de ruido en imágenes naturales: Eliminación del ruido y procesamiento de imágenes [63] [55] [21] [34] [45] [15] [22] [44] [79] [2] [61].

Aplicaciones biomédicas: Electroencefalogramas (EEG), Resonancia Magnética Nuclear (MNR), magnetoencefalógrafo (MEG), Electrocardiograma (ECG), etc [63] [55] [34] [52] [45] [59] [12].

Señales de audio: señales de audio y voz donde puede suprimirse ruido e interferencias [63] [55] [34] [45] [31] [17] [71] [4] [93].

Telecomunicaciones: antenas, radares, señales de banda ancha, etc [63] [55] [34] [45] [54] [28].

Control de máquinas: sensores piroeléctricos y ópticos, vibración en máquinas rotativas [63] [55] [34] [45][29] [94] [87].

Análisis sísmicos: separación de señales sísmicas [63] [55] [34] [45] [87].

MOS-VLSI: algoritmos implementables en digital y VLSI [63] [55] [34] [45]

1.12. Conclusiones

ICA es una técnica estadística en la cual las variables aleatorias llamadas observaciones son linealmente transformadas a componentes independientes con una máxima independencia y características de distribución interesante. Para poder llevar a cabo el análisis de componentes independientes deben de establecerse 3 hipótesis: fuentes independientes, fuentes no gaussianas y modelo de mezcla predeterminado. La idea de no gaussianidad puede usarse para derivar diferentes funciones que permitan maximizar la estimación del modelo de ICA.

Las mezclas u observaciones pueden presentarse en 3 tipos de modelos: el modelo lineal instantáneo de mezcla, el modelo convolutivo y el modelo no lineal. Este último suele presentarse en mezclas no lineales "puras" mezclas post no lineales. Para cada uno de estos modelos han sido desarrollados diferentes algoritmos de investigación y de los cuales se hace necesario recurrir a fundamentos matemáticos como la teoría de la información, estadísticos de segundo y alto orden, métodos geométricos, etc.

Capítulo 2

CONEXIÓN ENTRE EL SISTEMA VISUAL HUMANO (SVH), EL ANÁLISIS DE COMPONENTES INDEPENDIENTES (ICA) Y EL RUIDO

El desarrollo de este capítulo pretende introducir el campo del procesado de imágenes digitales, el ruido y la conexión entre éstos con la técnica ICA. El procesado digital de imágenes se encuentra inspirado por el procesamiento

visual humano. La representación de una imagen consiste en descomponerla en un conjunto de imágenes, donde cada uno de estos conjuntos cuenta con características diferenciables y sensibles a una estructura de la escena original. Los estudios llevados a cabo por David H. Hubel y Torsten N. Wiesel han despertado un gran interés basado en el hecho de que en la corteza visual gran parte de las neuronas parecen responder a líneas o bordes con distintas orientaciones.

2.1. Bases del procesamiento de imágenes digitales

Tal y como se menciona con anterioridad el procesado digital de imágenes se encuentra inspirado por el procesamiento visual humano, de esta forma resulta útil entender como se realiza el proceso receptivo en el sistema visual humano, el cual se produce en una serie de pasos los cuales permiten que una persona pueda observar un objeto que esta delante.

2.1.1. El Sistema Visual Humano (SVH)

La luz que es visible por el ojo humano pertenece a una pequeña zona del espectro electromagnético comprendida entre la franja de infrarrojos y los ultravioleta. De esta forma, puede decirse que el ojo es un órgano especializado en la recepción de la luz [30]. En la tabla 2.1 se pueden observar las

diferentes longitudes de onda y frecuencias del espectro.

Cuadro 2.1: Espectro electromagnético

Denominación	Longitud de onda [m]	Frecuencia [Hz]
Rayos gamma	< 10 pm	> 30,0 EHz
Rayos X	< 10 nm	> 30,0 PHz
Ultravioleta extremo	< 200 nm	> 1,5 PHz
Ultravioleta cercano	< 380 nm	> 789 THz
Luz visible	< 780 nm	> 384 THz
Infrarrojo cercano	< 2,5 um	> 120 THz
Infrarrojo medio	< 50 um	> 6,00 THz
Infrarrojo lejano/submilimétrico	< 1 mm	> 300 GHz
Microondas	< 30 cm	> 1,0 GHz
Ultra alta frecuencia radio	< 1 m	> 300 MHz
Muy alta frecuencia radio	< 10 m	> 30 MHz
Onda corta radio	< 180 m	> 1,7 MHz
Onda media (AM) radio	< 650 m	> 650 kHz
Onda larga radio	< 10 km	> 30 kHz
Muy baja frecuencia radio	> 10 km	< 30 kHz

Referenciandonos un poco en la historia, la naturaleza de la luz ha sido explicada de diferentes formas [6]. En 1671 Newton hacía referencia a la Teoría Corpuscular, la cual dice que la luz esta compuesta por corpúsculos que viajan por el espacio en línea recta. En 1678 Huygens, Young y Fresnel postularon la Teoría Ondulatoria la cual también requiere del Éter para poder transportarse. Maxwell en 1860 postula la Teoría Electromagnética la cual plantea características parecidas a las ondas de radio y Plank la esbozo como paquetes de energía a los cuales llamo Cuantos. En 1924 se unifica la Teoría Electromagnética y la de los Cuantos debido a que se demuestra la na-

turalidad doble de la luz. En la figura 2.1 puede observarse una representación del espectro electromagnético [35] [32] [53] [77].

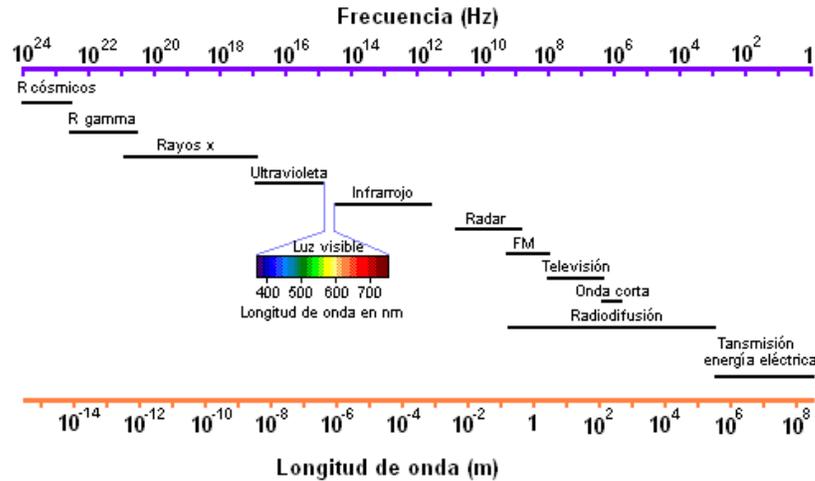


Figura 2.1: Gráfica espectro electromagnético adaptado de: http://edison.upc.es/curs/llum/luz_vision/luz.html

El ojo humano es aproximadamente un globo de 2 centímetros de diámetro, (ver figura 2.2). La luz entra a través de la cornea, para pasar por el humor acuoso, llega al cristalino y a través del humor vítreo forma una imagen invertida en la retina la cual esta compuesta por dos tipos de receptores, los conos que proporcionan una visión en color y los bastones que proporcionan la visión en la oscuridad [60] [80] [56]. Estos fotorreceptores al ser excitados transmiten señales eléctricas que llegan al nervio óptico para pasar de ahí a la corteza visual y finalmente al cerebro.

La retina es una capa de tejido neural, en la cual se presentan reacciones específicas que dependen de las radiaciones luminosas. En su superficie

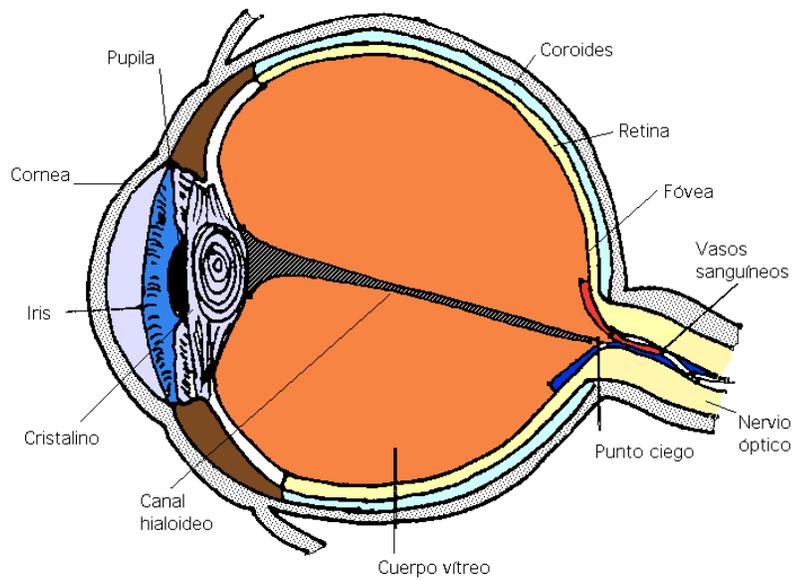


Figura 2.2: Gráfica del ojo humano. Vista lateral, adaptado de: <http://webvision.med.utah.edu/spanish/anatomia.html>

pueden ser observadas diferentes estructuras [78]:

Papila o disco óptico: en la papila o disco óptico se encuentran reunidas todas las fibras nerviosas de las células ganglionares de la retina que abandonan el globo ocular para formar el nervio óptico. En esta superficie de la retina no existen fotorreceptores, por lo que se le conoce como punto ciego.

Ora serrata: borde o línea en zig-zag de la retina. Es el límite de la misma. Existe una ora serrata nasal o medial y una ora serrata lateral o temporal. Por delante de la ora serrata la retina que compone la porción ciliar y la porción irídea de la retina no es fotosensible.

Fóvea: la fóvea forma una pequeña porción de la retina. Presenta un mayor número de células ganglionares, con una distribución más regular y precisa de los elementos estructurales, especialmente de conos y bastones. Los vasos sanguíneos rodean la fóvea tanto por arriba como por abajo. Dentro de ella se encuentran pequeños capilares.

Área central de la retina: es la porción de la retina que rodea a la fóvea y donde se produce la mayor fotorrecepción. La fóvea y la pequeña área que la rodea contiene un pigmento amarillo es por ello que se les denomina mácula lútea.

Área periférica de la retina: donde los elementos de la retina son de menor número, mayor tamaño y distribuidos menos regularmente. Tiene

menos capacidad de fotorrecepción.

En la retina se pueden encontrar seis tipos diferentes de células: células fotorreceptoras (conos y bastones), células horizontales, células bipolares, células amacrinas, células interplexiformes y células ganglionares. Éstas están distribuidas en tres capas. En la primera capa se distinguen los conos y bastones, los cuales no tienen el mismo número a lo largo de toda la retina, sino que varían en cantidad respondiendo de forma diferente a la luz. El espectro visible en el SVH tiene tres tipos de colores asociados cada uno a una longitud de onda diferente (ver figura 2.4), el ancho de banda en el que se encuentran estos tres colores son [80]:

Azul: entre 400nm - 500nm.

Verde: entre 500nm - 600nm.

Rojo: entre 600nm - 700nm.

Para ver cualquier color, el cerebro debe comparar el estímulo entrante de diferentes clases de células (cono), además de otras relaciones. Reconocer un color comienza en la retina que cuenta con tres capas de células. Las señales de los conos rojos y verdes de la primera capa son comparadas por las de células rojas y verdes presentes en la segunda capa. Estas células computan el balance entre la luz roja y verde que viene de una parte determinada del campo visual. Otras células comparan señales de los conos azules con las señales combinadas de los conos rojos y verdes. Las comparaciones de los

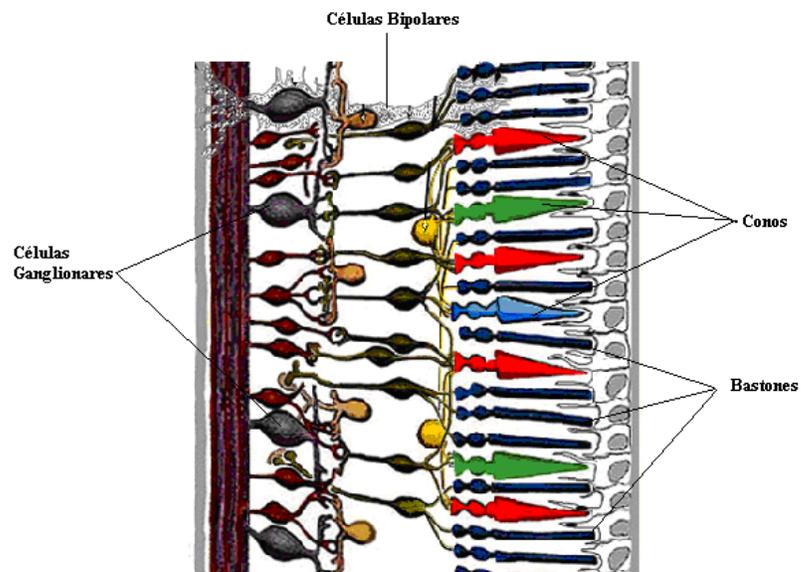


Figura 2.3: Organización de las distintas capas en la retina, adaptado de: <http://webvision.med.utah.edu/spanish/vgeneral.html>

diferentes trozos vecinos de una imagen permiten tener una gran habilidad para ver los colores como constantes. “La constancia del color es la propiedad más importante del sistema del color”, expreso Semir Zeki [96]. Si se usara el color para clasificar objetos percibidos, y este cambiara bajo diferentes condiciones, el color se convertiría en una característica poco adecuada para realizar una clasificación. La clave de la constancia del color es que no se determina el color de un objeto aislado; el color del objeto se determina de una comparación de las longitudes de onda reflejadas desde el objeto y de sus alrededores.

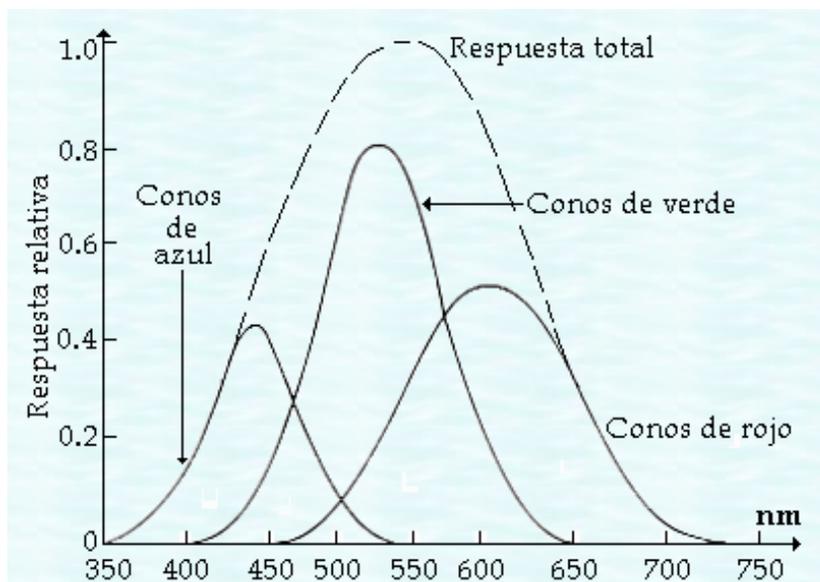


Figura 2.4: Respuestas de los tipos de conos (rojo, verde y azul) en la visión a color

A la segunda capa de la retina la cual esta compuesta por células bipolares, se conectan los axones de los fotorreceptores. La tercer capa se compone

de células ganglionares y se encuentra unida a la segunda capa por medio de las células bipolares (ver figura 2.4) [80].

La distribución de conos y bastones en la retina forman una capa sináptica exterior con las células bipolares. La sinapsis es un lugar en el cual una célula del sistema nervioso trasmite una señal a otra, de esta forma se obtienen caminos visuales del ojo al cerebro. La figura 2.5 muestra el recorrido visual que la imagen realiza desde que es captada por los ojos, donde las fibras que forman cada nervio óptico son divididas en dos grupos en las cuales las más próximas a las fosas nasales atraviesan el quiasma óptico, en el que las fibras nerviosas procedentes de la mitad nasal de la retina cruzan al lado derecho y las procedentes de la mitad temporal cruzan al lado izquierdo, esto es, se dirigen al lado contrario al que fueron captadas, componiendo los tractos ópticos derecho e izquierdo. En los núcleos geniculados externos se producen las últimas sinapsis, desde éste punto las radiaciones ópticas son transmitidas hasta la corteza estriada donde son reflejadas [80].

El ojo está conectado al cerebro a través del nervio óptico que le envía mensajes por medio de las células fotosensibles de la retina. Cuando la imagen es reflejada en la retina, ésta lo hace en forma compuesta o de mosaico, a partir de una gran cantidad de elementos separados o distintos. Así, cuando en el ojo se proyecta una imagen del entorno o medio sobre los fotorreceptores, cada uno de ellos responde a la intensidad de luz (la cual genera impulsos eléctricos sobre la retina) que incide sobre él. El cerebro no llega a ver una imagen definida; por detrás de la retina hay neuronas (fibras nerviosas) que

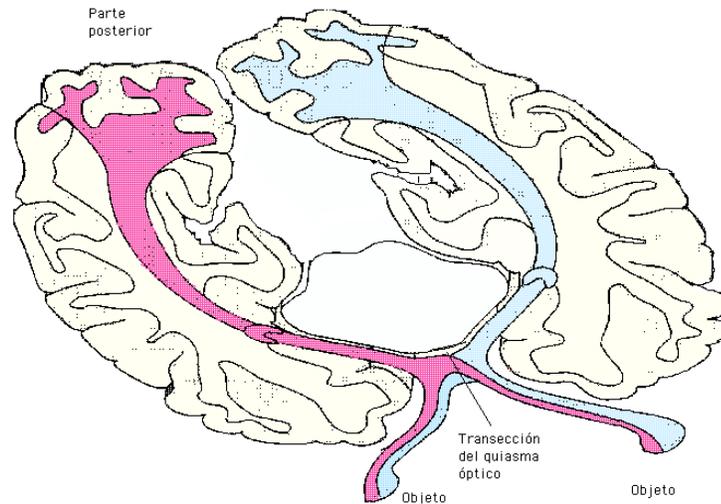


Figura 2.5: Recorrido visual de la imagen desde el ojo al cerebro.

están conectadas cada una a un fragmento de la retina y ven una parte de la escena global. De esta forma, cada una de las fibras nerviosas se encarga de analizar la información, combinar las señales llegadas y enviar mensajes al cerebro, no la imagen total.

Toda la información sensorial que penetra a través de los sentidos viaja por el sistema nervioso en forma de señales eléctricas, las cuales seguirán distintos caminos dentro de la médula espinal y el cerebro llegando a diferentes poblaciones de neuronas (núcleos), donde las señales son relevadas a otros centros nerviosos y así sucesivamente.

Estas fibras nerviosas se unen en el nervio óptico hasta una parada intermedia en el cerebro el cual se conoce como cuerpo geniculado lateral en el tálamo. Desde aquí hacia la corteza visual que contiene un mapa detallado

de la imagen vista por el ojo. La parte de la corteza que analiza los datos es de 2.5 mm de grosor (la materia gris). Debajo existe una capa de 1 mm compuesta de materia blanquecina con cables fibras que van de una célula a otra y que se conoce como materia blanca. Cada parte contiene un rústico mapa de la imagen: una zona con un mapa de color, otra con la distancia, otra con la profundidad, otra con el movimiento, etc [64] [80].

En la corteza V1, están localizadas las neuronas claves para el procesamiento de las formas. En la corteza se distinguen 6 capas; la capa 4 representa el principal destino de llegada de los axones que vienen del geniculado lateral y se divide en las subcapas 4A, 4B, 4C (4Ca, 4Cb). A ellas llegan los axones de las neuronas parvocelulares y magnocelulares. El área V4, esta llena de células que se descargan activamente cuando son expuestas a colores diferentes. Zeki [96] descubrió que algunas de las células en esta área, responden consistentemente al color real de la superficie de una porción del Mondrian, a pesar de las condiciones de iluminación y específico que posiblemente sean estas células las que llevan a cabo la constancia del color.

Los impulsos nerviosos que se reciben en las cortezas sensoriales tienen determinados patrones espacio-temporales, los cuales provocan la inhibición o excitación de determinadas neuronas corticales. Una vez que las señales alcanzan la corteza primaria, éstas son enviadas hacia otras cortezas visuales las cuales se conocen como secundarias, donde se produce el procesamiento de aspectos más dificultosos de la información visual. Posteriormente, la información viaja hacia otras regiones conocidas como cortezas asociativas,

donde se produce la asociación de los estímulos visuales con estímulos de otras formas sensoriales. El flujo entre cada una de estas cortezas (primaria, secundaria y asociativa), se produce de forma bi-direccional, estableciéndose múltiples vías de interacción entre ellas y permitiendo que las señales visuales sean procesadas en varios pasos, en cada uno de los cuales la abstracción se hace más compleja.

En la corteza visual primaria, existen neuronas que tienen la propiedad de ser sensitivas a la dirección del movimiento. Estas células fueron descubiertas por David H. Hubel y Torsten N. Wiesel, quienes describieron que la corteza visual primaria está organizada en una serie de columnas en donde las células de esta zona sólo se excitan cuando se les muestra líneas orientadas [37] [38] [40] [39].

Las neuronas son las unidades básicas del sistema nervioso (células nerviosas). Su principal propiedad está en la excitabilidad de su membrana que puede ser modificada por la acción de otras neuronas. El funcionamiento del sistema nervioso depende de la transmisión de estas señales de neurona en neurona, las cuales se llevan a cabo gracias a reacciones químicas y eléctricas que llegan a través de las dendritas y son emitidas por medio del axón produciendo una conexión funcional denominada sinapsis [18] [57]. Esto es, la transmisión en las neuronas se realiza por medio de impulsos eléctricos en donde son liberadas moléculas denominadas neurotransmisores los cuales alcanza la membrana de otra neurona provocando en ella un cambio de potencial eléctrico, el cual se transforma en un impulso capaz de excitar o inhibir

la continuación del mensaje [18] [57].

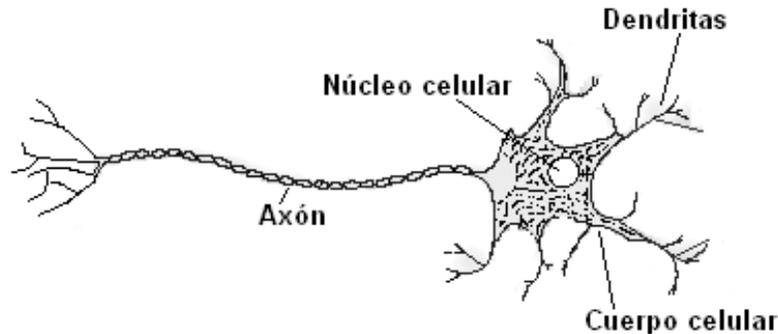


Figura 2.6: Partes de una neurona.

Para ver imágenes en colores el cerebro se ocupa de la forma el color y el movimiento de manera separada. En la retina, los bastones permiten poder responder a un único fotón o partícula de luz. Este tipo de células, cuenta con una proteína denominada rodopsina [82], [?], receptora sensible a la luz la cual está presente en las membranas de los discos y es la que indica llegadas del pulso de luz. Los fotorreceptores denominados conos trabajan con luz brillante y son los responsables de la gran agudeza visual además de la visión de los colores, se encuentran concentrados en la fovea.

Los objetos parecen ser de un color en particular porque reflejan más algunas longitudes de onda que otras. En 1802, Young, concluyó que la retina no podía tener un receptor diferente para cada una de estas longitudes de onda, las cuales cruzan todo el espectro de colores, que va desde el violeta al rojo y propuso que los colores eran percibidos por un código de tres colores. En el procesamiento de señales, resulta fundamental la búsqueda de mode-

los generativos estadísticos de las diferentes señales observadas ya que las componentes de estos modelos muestran una representación de los datos, pudiendo usar estas muestras en trabajos como el reconocimiento de patrones, la compresión de imágenes, etc.

En la actualidad, se han llevado a cabo diferentes investigaciones enfocadas en el área del procesamiento de imágenes teniendo como principal referencia el SVH [58] [15] [21], debido a la particularidad que presenta la fácil adaptación a los diferentes procesos de información que está constantemente recibiendo. El tratamiento de estas imágenes para obtener la información deseada requiere de ciertos procesos que implican una manipulación la cual permita extraer datos subyacentes; un análisis que determina estructuras elementales como bordes, regiones, texturas, etc [91], y unas aplicaciones para poder dar solución a problemas como el reconocimiento, la reconstrucción en 3D, etc.

El papel de la corteza visual primaria en la representación de imágenes resulta muy importante ya que la estimulación de una determinada región en la retina excita las neuronas de una región específica de la corteza visual, al igual que la estimulación de regiones adyacentes excita regiones corticales adyacentes. En esta región se encuentran dos tipos de células las cuales se caracterizan por la estructura de sus campos receptivos: [39] [40]

Células simples: se presentan en gran abundancia. Se caracterizan por campos receptivos denominados simples, que responden principalmente a contornos paralelos claro-oscuro con una determinada orientación.

Células complejas: se caracterizan por campos receptivos complejos que responden principalmente a contornos claro-oscuro con una determinada orientación y extensión espacial, interrupciones en contornos y esquinas. Presentan su principal activación con patrones de estímulos móviles.

El sistema visual central se caracteriza por una organización “retinotópica” no lineal, donde la información que proviene de puntos adyacentes de la imagen de la retina es procesada por neuronas adyacentes [30], [82] [61]. Esta proyección retinotópica tiene una orientación horizontal paralela a la superficie del cerebro. Igualmente existe un principio organizativo vertical orientado perpendicularmente a la superficie del cerebro, según el cual las neuronas de la corteza visual se agrupan en columnas corticales [40], las cuales se consideran como la unidad mínima de procesamiento de la corteza y se caracterizan por: en la misma región de la retina se encuentran los campos receptivos de las neuronas dentro de una columna cortical, los cuales son sensibles a los estímulos luminosos de un mismo ojo y responden a orientaciones muy definidas de los contornos del estímulo.

2.2. ICA e Imágenes

A lo largo del desarrollo de este capítulo se ha visto que resulta de gran valor la cantidad de campos en los que las imágenes juegan un papel muy importante e igualmente existen diferentes técnicas para su procesado. A nivel

general el procesamiento de imágenes siempre tiene como objetivo la mejora de la misma ya que estas pueden encontrarse corrompidas por diferentes clases de ruido, o estar distorcionadas, entre otros problemas.

Las percepciones que cada individuo tiene sobre determinadas situaciones, a través de los sentidos (vista, oído, tacto, gusto y olfato), permite que se pueda asociar el concepto básico de imagen al de un espacio temporal. El recibir información a través del sistema visual involucra la recepción de señales luminosas (fuente de energía electromagnética), con rangos de frecuencias espectrales y mínimas intensidades; pero no implica que este se convierta en la única manera posible de captar imágenes, ya que existen sensores capaces de obtener información que no dependen de la iluminación. Debido a que los rangos de sensibilidad de los diferentes sensores son muy pequeños comparándolo con la información que puede proveer el espectro electromagnético, el problema de una correcta identificación partiendo de valores espectrales debe también resolverse recurriendo al uso de información complementaria de tipo estructural extraída de la propia imagen, información experimental previa o información de la misma experiencia. Existen diferentes aplicaciones del análisis de componentes independientes, al procesado digital de imágenes. Múltiples autores como [15] [63] [55] [21] [34] [45] [15] [22] [44] [79] [2] [61], han referido el uso de ICA para la extracción de determinadas características de una imagen.

Por medio del uso de modelos estadísticos se pueden determinar características de imágenes naturales. En los modelos lineales, las columnas de la

matriz de mezcla proporcionan los vectores base que se adaptan a la estructura estadística de las imágenes naturales [45] [50]. El modo de actuación de las neuronas de la corteza visual primaria tiene un comportamiento similar a la naturaleza de ICA, de esta forma cuando se aplica dicha herramienta a las imágenes naturales se obtienen aproximaciones muy buenas de los campos receptivos de las neuronas a través de los coeficientes de ponderación del modelo lineal de ICA, además del comportamiento disperso de dicho modelo en la respuesta de las neuronas del sistema visual humano [45] [50].

2.2.1. Relación entre ICA y las imágenes

En las imágenes representadas en la figura 2.7, se puede apreciar dos tipos de imágenes, arriba esta la imagen lena de 256 x 256 pixeles, a la cual hemos aplicado ICA obteniendo sus características típicas, representadas en la figura de abajo. En ésta última puede apreciarse la aparición de bordes y otras características de interés. Para este caso, en primera instancia los datos han sido blanqueados y ha sido utilizado en algoritmo de fastICA para estimar la matriz de mezcla \mathbf{A} . Cada parche corresponde a una columna \mathbf{a}_i de la matriz de mezcla estimada [45]

Para el tratamiento de datos en imágenes se suele recurrir a dos tipos diferentes de soluciones. La primera alternativa se presenta como una solución local donde la matriz de blanqueamiento $\mathbf{V}_{ZCA} = E\{\mathbf{xx}^T\}^{-1/2}$ filtra de manera idéntica una determinada región local de entrada, de manera similar como los campos receptivos en la retina y en el núcleo geniculado



Figura 2.7: Arriba: Imagen original cameraman 256 x 256 píxeles. Abajo: características típicas de la imagen, obtenidas aplicando ICA a bloques de 8 x 8 píxeles.

lateral (LGN) actúan. La segunda alternativa se presenta con el análisis de componentes principales, donde los filtros son ortogonales lo que hace que las fuentes estimadas no tengan correlación. Aquí, $\mathbf{V}_{PCA} = \mathbf{D}^{-1/2}\mathbf{E}^T$ donde $\mathbf{E}\mathbf{D}\mathbf{E}^T = E\{\mathbf{x}\mathbf{x}^T\}$ es un sistema propio de la matriz de correlaciones $E\{\hat{\mathbf{x}}\hat{\mathbf{x}}^T\}$. Además PCA permite reducir la dimensión seleccionando solamente un subconjunto de los componentes $\mathbf{z} = \mathbf{V}_{PCA}\mathbf{x}$, con lo que se obtiene ganancia en bajos costes de cómputo, menor tiempo de consumición y de memoria, etc.

Una vez con los datos pre-procesados ICA transforma \mathbf{W} reduciendo al mínimo la dependencia estadística entre las fuentes estimadas.

$$\hat{\mathbf{s}} = \mathbf{W}\mathbf{z} = \mathbf{W}\mathbf{V}_{PCA}\mathbf{x} = \mathbf{W}\mathbf{D}_n^{-1/2}\mathbf{D}_n^T\mathbf{x} \quad (2.1)$$

donde \mathbf{D}_n es una matriz diagonal que contienen los n valores propios de la matriz de correlación $E\{\mathbf{x}\mathbf{x}^T\}$ y \mathbf{E}_n es la matriz con los vectores propios correspondientes a las columnas.

Es importante también señalar que la semejanza que presentan las características de ICA con los campos receptivos de neuronas en corteza visual primaria sugiere que las neuronas son capaces de realizar un cierto tipo de análisis componente independiente y que los campos receptivos están optimizados para las imágenes naturales de proceso [45] [66] [67] [15].

2.3. ICA y el ruido

El ruido a nivel general se presenta como una mezcla de señales aleatorias las cuales pueden tener diferentes frecuencias (mayor o menor) a la señal de interés. El ruido en imágenes digitales se presenta como valores de un píxel que no corresponden con la realidad. Generalmente, al adquirir imagen digitales éstas vienen contaminadas por ruido el cual puede ser introducido por los equipos utilizados para la captación de la imagen, o ruido que se va añadiendo tramo a tramo cuando se envía la información, etc [45].

En la sección 1.9.1 hacía referencia a soluciones al problema de la separación ciega de fuentes teniendo en cuenta los métodos de eliminación de ruido. Generalmente, cuando se habla de ruido para la técnica ICA, éste es siempre asumido de forma aditiva, de la tal forma que la ecuación 1.2 puede ser expresada matemáticamente como:

$$\mathbf{x} = \mathbf{A}\mathbf{s} + \mathbf{n} \quad (2.2)$$

donde el ruido se encuentra representado por el vector $\mathbf{n} = (n_1, n_2, \dots, n_n)$.

Igualmente suelen considerarse dos problemas:

- El ruido se presenta de manera independiente de las componentes independientes.
- El ruido es gaussiano.

Tal como hemos mencionado con anterioridad, en la vida real se presentan muchos casos donde el ruido esta siempre presente en las observaciones. Al

igual que para el caso ideal de la no presencia de ruido, para encontrar la matriz de mezcla del modelo de ICA con ruido se deben de recurrir también a las restricciones de independencia y no gaussianidad [62] [46] [7]. Asumiendo el ruido como conocido, con matriz de covarianza Σ y de forma $\sigma^2\mathbf{I}$ se pueden presentar diferentes casos [45]:

2.3.1. Ruido de las fuentes

Este presenta como característica que el ruido es adicionado en las componentes independientes y no en las mezclas de las observaciones. De la ecuación 2.2 y asumiendo ruido adicionado a las componentes independientes de las fuentes, se obtiene:

$$\mathbf{x} = \mathbf{A}(\mathbf{s} + \mathbf{n}) \quad (2.3)$$

donde la covarianza del ruido es diagonal. Considerando el ruido de las componentes independientes dado por: $\hat{s}_i = s_i + n_i$ se puede obtener:

$$\mathbf{x} = \mathbf{A}\hat{\mathbf{s}} \quad (2.4)$$

de la ecuación anterior se puede observar claramente que ésta representa el modelo base de ICA representado en 1.2 con las componentes \hat{s} conservando las restricciones de independencia y no gaussianidad. Esto conduce a poder estimar la matriz de mezcla \mathbf{A} y el ruido de las componentes a través del modelo básico de ICA. La dificultad en este caso se presenta al querer estimar las componentes independientes originales del ruido.

Asumiendo la covarianza del ruido de la forma [45]:

$$\Sigma = \mathbf{A}\mathbf{A}^T\sigma^2 \quad (2.5)$$

El ruido puede ser transformado en un vector de la forma $\hat{\mathbf{n}} = \mathbf{A}^{-1}\mathbf{n}$ con lo que obtenemos [45]:

$$\mathbf{x} = \mathbf{A}\mathbf{s} + \mathbf{A}\hat{\mathbf{n}} = \mathbf{A}(\mathbf{s} + \hat{\mathbf{n}}) \quad (2.6)$$

donde la covarianza de $\hat{\mathbf{n}}$ es $\sigma^2\mathbf{I}$ lo que permite independencia en $\mathbf{s} + \hat{\mathbf{n}}$.

2.3.2. Pocas fuentes de ruido

Este caso se presenta cuando el número de las componentes de ruido y de las componentes independientes no es tan grande. De esta forma asumiendo el vector de componentes independientes como $\hat{\mathbf{s}} = (s_1, s_2, \dots, s_k, n_1, n_2, \dots, n_l)^T$ donde las componentes independientes están representadas por $s_i, i = 1, \dots, k$ y el ruido por $n_i, i = 1, \dots, l$ si se asume las mezclas como la sumatoria de las componentes independientes y el ruido $k + l$ se puede obtener:

$$\mathbf{x} = \mathbf{A}\hat{\mathbf{s}} \quad (2.7)$$

donde la matriz \mathbf{A} contiene la mezcla de las componentes independientes y la estructura de covarianza del ruido; $\hat{\mathbf{s}}$ son las mezclas observadas, las cuales son iguales al número de componentes independientes.

2.3.3. Técnicas de diagonalización para la estimación de la matriz de mezcla

Con la introducción de ruido en los datos originales existen casos para los cuales el estimar la matriz de mezcla se convierte en una tarea difícil. En estos casos suele recurrirse a técnicas de diagonalización en las cuales se tiene como objetivo la modificación del método de “ruido libre” de ICA con el fin de eliminar o reducir el ruido [45]:

Denotando la ecuación del “ruido libre” como:

$$\mathbf{v} = \mathbf{A}\mathbf{s} \quad (2.8)$$

Se puede recurrir a la búsqueda de proyecciones, sea $\mathbf{w}^T \mathbf{v}$ es no gaussiano y esta localmente maximizado con el blanqueamiento de los datos con $\|\mathbf{w}\| = 1$. En [45] se demuestra que proyecciones en estas direcciones se pueden hallar con la estimación de las componentes independientes. Este camino presenta ciertas ventajas como es el de ser en muchos casos inmune a ruido gaussiano, así como la facilidad los datos originales de la señales con ruido. Esto es $\mathbf{w}^T \mathbf{x} = \mathbf{w}^T \mathbf{v} + \mathbf{w}^T \mathbf{n}$, donde se puede ver claramente que la medida de no gaussianidad de $\mathbf{w}^T \mathbf{v}$ desde las observaciones $\mathbf{w}^T \mathbf{x}$ no se ve afectada por el ruido $\mathbf{w}^T \mathbf{n}$.

Existen diferentes técnicas para la diagonalización como son la diagonalización por kurtosis, diagonalización general para medidas no gaussianas, el algoritmo fastICA orientado a ruido en los datos, métodos de cumulantes de

alto orden, métodos de maximización de la vecindad. Para más información mirar: [45].

2.3.4. Separación por medio de código “sparse”

A pesar de que el propósito básico de la estimación del ruido ICA es estimar las componentes independientes, el modelo se puede utilizar para desarrollar un método también de separación. Asumiendo una matriz observada \mathbf{x} :

$$\mathbf{x} = \mathbf{v} + \mathbf{n} \quad (2.9)$$

donde los datos han sido previamente modelados por:

$$\mathbf{v} = \mathbf{A} + \mathbf{s} \quad (2.10)$$

La eliminación de ruido en \mathbf{x} computando las estimaciones de $\hat{\mathbf{s}}$ de las componentes independientes por medio del método de la estimación máxima [45], de donde se puede obtener:

$$\hat{\mathbf{v}} = \mathbf{A} + \hat{\mathbf{s}} \quad (2.11)$$

Aquí la condición expuesta en la ecuación 2.5 puede ser satisfecha ya que \mathbf{A} es ortogonal y la covarianza del ruido es de la forma $\sigma^2\mathbf{I}$. A este tipo de modelo los autores le llaman sparse code shrinkage [41] [42]. Más información en [45].

Para este trabajo vamos a centrarnos en tres clases diferentes de ruido: [90]

Ruido gaussiano: La característica principal de esta clase de ruido se presenta en que todas las frecuencias presentan un espectro de energía constante, esto provoca variaciones de los valores de los diferentes píxeles cada vez que se captura la imagen, provocando que se sume o reste un valor al nivel de gris real para el caso de imágenes en escala de grises. Matemáticamente esto puede expresarse como:

$$\mathbf{G}(i, j) = \mathbf{I}(i, j) + \mathbf{n}(i, j) \quad (2.12)$$

donde para nuestro caso, \mathbf{G} va a representar las observaciones de la imagen original representada por \mathbf{I} y \mathbf{n} representa el ruido que es generalmente $\sim N(m, \sigma^2)$. La función de densidad $f(x)$ para una gaussiana de media m y varianza σ^2 es:

$$f(x) = \frac{1}{2\pi\sigma^2} \exp\left[-\frac{1}{2\sigma^2}(x - m)^2\right] \quad (2.13)$$

El ruido gaussiano tiene un efecto general en toda la imagen, es decir, la intensidad de cada píxel de la imagen se ve alterada en cierta medida con respecto a la intensidad en la imagen original.

Ruido multiplicativo: Se caracteriza por tener el ruido dependiente de la señal, esto es, cuanta más señal más ruido. Matemáticamente esto se

expresa como:

$$\mathbf{G}(i, j) = \mathbf{I}(i, j) + \lambda \mathbf{I}(i, j) \quad (2.14)$$

donde $\lambda \sim U([a, b])$. Generalmente b es el inverso de a , ($b = -a$).

Existen también otros modelos de ruido multiplicativo:

Gaussiano multiplicativo:

$$\mathbf{G}(i, j) \sim N(\mathbf{I}(i, j), \alpha \mathbf{I}(i, j)) \quad (2.15)$$

con $\alpha > 0$.

Poisson

$$\mathbf{G}(i, j) \sim P(\mathbf{I}(i, j)) \quad (2.16)$$

Ruido de sal y pimienta: Un tipo común de ruido impulsivo es el ruido de sal y pimienta, el cual en muchas ocasiones es causado por mal funcionamiento de píxeles en sensores de cámaras, o por transmisiones en canales con mucho ruido [16]. En este tipo de ruido existen sólo dos posibles valores, q y p . Para una imagen de 256 niveles de gris el valor típico de la pimienta es 0 y de la sal 255. Matemáticamente se tiene:

$$\begin{aligned} \mathbf{G}(i, j) &= \mathbf{I}(i, j) \quad \text{si } x < p \\ &= A \quad \text{si } p \leq x < q \\ &= B \quad \text{si } q \leq x < 1 \end{aligned}$$

donde $x \sim U[0, 1)$, $0 \leq p \leq q \leq 1$.

Vale la pena resaltar que $(q - p)$ es la probabilidad de pasar al valor A (255 y por tanto sal) y $(1 - q)$ es la probabilidad de pasar a B (0 y por tanto pimienta), p es la probabilidad de permanecer igual.

2.4. Conclusiones

A nivel particular el sistema visual humano cumple tareas específicas que comprenden todo tipo de manejo de información. Es en este sistema en el que se han basado por muchos años gran cantidad de autores con el fin de encontrar nuevos métodos que permitan un mejor procesamiento de datos de imagen.

El recorrido visual para la percepción de objetos puede resumirse como [80]:

Formación de la imagen: se produce cuando la luz llega al ojo y se refleja sobre la retina con el fin de crear la imagen retinal.

Codificación: la imagen retinal se codifica y hace que se generen señales en los fotorreceptores (conos y bastones), con el fin de poder enviar información a la corteza visual.

Representación: en la corteza visual dicha imagen codificada se procesa, permitiendo operaciones como reconocimiento, detección, etc.

Interpretación: a través de los nervios, son transmitidos impulsos eléctricos en dirección al cerebro los cuales, una vez llegados al mismo son procesados para permitir finalmente, la percepción del objeto. Los mecanismos de decodificación del sistema receptor (cerebro), permiten asociar propiedades como color, forma, movimiento, etc.

A nivel general mucha de la información que se maneja comienza con la evaluación de la calidad y eliminación de ruido existente ya que a menudo, las imágenes digitales se corrompen con ruido durante la transmisión o en otras partes del sistema y existen diferentes razones para querer eliminar el ruido como fenómeno parásito, de la señal de interés, principalmente el que se ocasionen pérdidas en la información y para el caso particular de imágenes obteniendo una mala calidad de las mismas. Esto es a nivel general una regla básica ya que cualquier sistema de visión artificial comienza con el procesamiento de los valores de intensidades. El método ICA al ser un modelo libre permite un gran desarrollo en el área de la eliminación de ruido; pero para muchos autores resulta aún especulativo.

Capítulo 3

Algoritmos ICA

El desarrollo de este capítulo pretende introducir al campo de los algoritmos de ICA que son usados para llevar a cabo la tarea final de esta investigación. En primera instancia de llevará a cabo un repaso de los algoritmos ICA en general para terminar introduciendo los algoritmos que nos interesan a nivel particular los cuales son: fastICA, JADE, AMUSE, fastGEO y Denoising ICA.

3.1. Breve reseña algoritmos ICA

A manera de información se pretende introducir en el campo de los algoritmos ICA, así como mostrar un breve resumen como reseña. En la tabla 3.1 se pueden observar de manera general los diferentes algoritmos y el tipo de mezcla a la que son orientados.

Cuadro 3.1: Tipos de mezclas de información y algoritmos orientados a ellas

Mezclas lineales
Estadísticos de 2° orden
Estadísticos de alto orden
Teoría de la información
Teoría de la máxima verosimilitud
Métodos geométricos
Señales no estacionarias
Mezclas convolutivas
Estadísticos de 2° orden
Estadísticos de alto orden
Aproximaciones en frecuencia
Mezclas no lineales (o lineales puras)
Estadísticos de alto orden
Redes Neuronales Artificiales
Métodos geométricos
Mezclas no lineales (post no lineales)
Entropía
Funciones de base radial
Repulsión de patrones
Aprendizaje conjunto

El análisis de componentes independientes (ICA), fue planteado por primera vez por Barlow en 1961, enfocando su trabajo a las unidades neuronales de una red las cuales tenían como principal característica la independencia estadística. Tomando como referencia esta idea, Linsker, Nadal y Parga, propusieron el aprendizaje sobre redes neuronales no supervisado, en el cual aplicaron los conceptos de minimización de la información mutua entre las salidas y entradas.

En la década de los 80 fue introducido el concepto de BSS fue por Heroult, Jutten y Ans [36], sección 1.1 quienes trataban de modelar el problema biológico de la codificación del movimiento, debido a que estos sistemas tienen como característica la posibilidad de extraer información partiendo de mezclas. Así, de este modo, se dispone de un conjunto de fuentes p , las cuales envían señales q , a través de un medio material, que hace que dicha información se mezcle entre si, de forma desconocida y que se puede observar por medio de unos sensores a los cuales esta llegando la información transmitida. El problema de BSS consiste en delimitar las fuentes desconocidas p , a partir de las observaciones q , reconstruyendo las señales originales sin conocer estas, ni como se han mezclado. Es por esta razón que se dice que BSS es “ciega”.

A principios de los años 90, algunos investigadores franceses, entre ellos Cardoso [20] proponen un método algebraico para el tratamiento de señales en el cual se emplean estadísticos de alto orden, este método lo llevaron a cabo desarrollando el algoritmo JADE. A mediados del mismo año, es cuando ICA despierta un gran interés en el campo de la investigación. A. Cichocki y

R. unbehauen [24] proponen un algoritmo adaptativo, mientras que E. Oja, J. Kerhunen y col. proponen una generalización basada en redes neuronales que pretende dar solución por medio del análisis de componentes principales en el campo del análisis de componentes independientes, sustituyendo la hipótesis de decorrelación por la de independencia [51]. Igualmente C.G. Puntonet [72] [70] y E. Lang [86] [85] propone una solución basada en algoritmos geométricos, los cuales tratan de explotar las propiedades geométricas y algebraicas de las señales originales y de las observaciones.

En 1995 se presenta el algoritmo infomax [14] [13], el cual rnlaza redes neuronales con la entropía; éste algoritmo fue presentado por A.J. Bell y T. Sejnowski. S.I. Amari [5] presenta el algoritmo infomax pero bajo el concepto del gradiente natural.

La primera versión del algoritmo de punto fijo fastICA [43] es presentada en 1997 y con él se abre un campo del análisis de componentes independientes enfocado a problemas de mayor dimensionalidad.

Actualmente las investigaciones en el área siguen realizándose, para mayor información puede referirse a [5] [19] [23] [43] [65]

Todos estos avances a través de los años han permitido que se desarrollen diferentes algoritmos que permiten la obtención de las señales originales en un proceso de mezcla, para las múltiples aplicaciones que se pueden presentar a nivel de biomedicina, aplicaciones de audio, supresión de ruido e interferencias, telecomunicaciones, etc. Estos algoritmos de separación de señales para mezclas instantáneas suelen consistir en una regla de aprendizaje iterativo,

la cual consiste en una adaptación serie se la matriz de separación \mathbf{W} .

3.2. Algoritmo fastICA

FastICA es un algoritmo muy popular de la técnica ICA, que se ha aplicado ya con éxito a muchos problemas fastICA. Este algoritmo ha sido propuesto por A. Hyvärinen y E. Oja [43] y busca las componentes independientes de la matriz de mezcla basándose en diferentes conceptos como la estimación de la máxima verosimilitud, maximización de la no gaussianidad mediante la kurtosis, minimización de la información mutua mediante la negentropía, etc.

Para este algoritmo las observaciones tienen que tener una media de cero tal cual como la matriz diagonal de covariación. Esto puede llevarse a cabo cuando las observaciones son blanqueadas. Para este propósito, la descomposición del valor propio de la matriz de la covariación de las observaciones tiene que ser computada. Haciendo \mathbf{C} la matriz de covarianza de la matriz de observaciones \mathbf{X} :

$$\mathbf{C} = \mathbf{V}\mathbf{D}\mathbf{V}^T \quad (3.1)$$

donde \mathbf{V} es la matriz del vector propio y \mathbf{D} es una matriz diagonal que contiene los valores propios de \mathbf{X} en su diagonal. De acuerdo con las matrices \mathbf{D} y \mathbf{V} , la matriz de blanqueamiento puede ser determinada por:

$$\mathbf{Q} = \mathbf{D}^{-1/2}\mathbf{V}^T \quad (3.2)$$

Multiplicando esta matriz con la matriz de observaciones \mathbf{X} puede conducir a variables uncorrelacionadas de la observación. Para este caso asumimos observaciones con una media de cero y blanqueadas.

El objetivo de fastICA es encontrar una matriz \mathbf{W} tal que las filas de $\mathbf{Y} = \mathbf{W}\mathbf{X}$ sean estadísticamente independientes. Para el algoritmo fastICA la independencia de las componentes de un vector aleatorio se miden por medio de la información mutua \mathbf{I} . Esta medida de la independencia se puede expresar en términos de la negentropía J :

$$\mathbf{I}(y_1, y_2, \dots, y_N) = J(\mathbf{y}) - \sum_i J(y_i). \quad (3.3)$$

\mathbf{I} es una función no negativa que desaparece si los elementos \mathbf{y} son estadísticamente independientes. Como la negentropía $J(\mathbf{y})$ es independiente bajo transformaciones lineares (por ejemplo $J(\mathbf{W}\mathbf{x}) = J(\mathbf{x})$ para cualquier matriz \mathbf{W} y cualquier vector \mathbf{x}) Necesitaría optimizarte sólo el segundo término.

El algoritmo fastICA puede ser descrito como:

$$\text{Maximice } \sum_{i=1}^N J(\mathbf{w}_i) \text{ w.r.t. } \mathbf{w}_i, i = 1, \dots, n \quad (3.4)$$

$$E\{(\mathbf{w}_k^T \mathbf{x})(\mathbf{w}_j^T \mathbf{x})\} = \delta_{jk}. \quad (3.5)$$

como $y_i = \mathbf{w}_i^T \mathbf{X}$. Es necesaria la normalización de $E\{(\mathbf{w}_k^T \mathbf{x})(\mathbf{w}_j^T \mathbf{x})\} = \delta_{jk}$ ya que las fuentes subyacentes se asumen para ser estadísticamente indepen-

dientes y sin correlación.

Para una buena optimización en fastICA hay que tener una buena estimación de la negentropía de cualquier variable aleatoria. De acuerdo a [43] puede recurrirse a la siguiente fórmula para una correcta valoración:

$$J(y_i) = \frac{1}{\delta^2} (E\{G(y_i)\} - E\{G(\nu)\})^2. \quad (3.6)$$

donde G es cualquier función no lineal ν es una variable gaussiana estandarizada y δ es una constante de normalización.

La expresión para $J(y_i)$ puede ser optimizada bajo las condiciones de Kuhn-Tucker y del método del Newton. De acuerdo a ref podemos llegar a:

$$\mathbf{w}^+ = E\{\mathbf{x}G'(\mathbf{w}^T \mathbf{x})\} - E\{G''(\mathbf{w}^T \mathbf{x})\} \mathbf{w} \quad (3.7)$$

$$\mathbf{w}^* = \frac{\mathbf{w}^+}{\|\mathbf{w}^+\|} \quad (3.8)$$

Este procedimiento se utiliza iterativo hasta que todas las filas de \mathbf{W} son determinadas. Un punto importante aquí, es que debe de evitarse que el algoritmo converja varias veces a los mismos máximos de la negentropía. Para este caso, Esto se puede alcanzar, después de ser desarrollada cada iteración del método del Newton son nuevamente determinadas las filas decorrelacionadas de \mathbf{w}_{q+1} con respecto a las filas de $\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_q$ de \mathbf{W} .

Una vez determinada la matriz \mathbf{W} , las fuentes se pueden obtener multiplicando el inverso de \mathbf{W} con las observaciones.

3.3. Algoritmo JADE

El algoritmo de diagonalización conjunta aproximada de las automatrices ha sido propuesto por J.F. Cardoso [20] y se basa en el cálculo de estadísticos de alto orden y en la diagonalización de la descomposición de los autovalores de las mezclas.

JADE se define como el tensor \mathbf{F} de cumulantes de cuarto orden donde las entradas son dadas mediante los cumulantes cruzados de cuarto orden $\mathbf{X} = [X_1, X_2, \dots, X_N]^T$

$$F_{ijkl} = cum(X_i, X_j, X_k, X_l), \quad 1 \leq i, j, k, l \leq N \quad (3.9)$$

\mathbf{F} es un operador lineal definido por el espacio de matrices $N \times N$. Si $\mathbf{M} \in \mathbb{R}^N$ entonces:

$$\mathbf{F}_{ij}(\mathbf{M}) = \sum_{kl} m_{kl} cum(X_1, X_j, X_k, X_l) \quad (3.10)$$

También \mathbf{F} es simétrica, con lo que se puede descomponer en los valores propios (EDV). De este forma el escalar λ se llama valor propio y la matriz \mathbf{M} de $N \times N$ se llama matriz de vectores propios \mathbf{F} si

$$\mathbf{F}(\mathbf{M}) = \lambda \mathbf{M} \quad (3.11)$$

De acuerdo al modelo blanqueado de separación de fuentes:

$$\mathbf{z} = \mathbf{V} \mathbf{A} \mathbf{s} = \mathbf{W}^T \mathbf{s} \quad (3.12)$$

donde \mathbf{V} es la matriz de blanqueamiento, \mathbf{W}^T la matriz de mezcla blanqueada y z el vector de datos blanqueados. Para este caso se puede demostrar que cada matriz de la forma:

$$\mathbf{M} = \mathbf{w}_m \mathbf{w}_m^T, \quad (3.13)$$

donde \mathbf{w}_m es m filas de la matriz \mathbf{W} , es una matriz de valores propios correspondiente a $\lambda = kurt(s_m)$ de \mathbf{F} :

$$\mathbf{F}(\mathbf{M}) = kurt(s_m) \mathbf{M}. \quad (3.14)$$

También el resto de los valores propios \mathbf{F} son cero. Es por esto que la matriz \mathbf{W} puede ser determinada por las matrices de valores propios de \mathbf{F} . Vale la pena resaltar que esta idea sólo trabaja si los valores propios de espectro de \mathbf{F} no se degeneran.

De acuerdo a lo anterior, para \mathbf{W} las diagonales $\mathbf{F}(\mathbf{M})$ para *cualquier* $N \times N$ matriz \mathbf{M} ,

$$\mathbf{W} \mathbf{F}(\mathbf{M}) \mathbf{W}^T \quad (3.15)$$

es diagonal. Por lo tanto, \mathbf{W} puede determinarse como la diagonal común de un sistema de K matrices $\mathbf{F}(\mathbf{M}_k)$, $1 \leq k \leq K$. Para esto se necesita una medida cualitativa para la diagonalización de la matriz $\mathbf{Q}_i = \mathbf{W} \mathbf{F}(\mathbf{M}_i) \mathbf{W}^T$.

Una idea puede ser tomar los elementos \mathbf{Q}_i . Como alternativa, también la suma de los cuadrados de los elementos diagonales se puede maximizar como la suma total de cuadrados de una matriz que no cambia bajo transformaciones ortogonales. La última medida conduce a la función de la blanco usada en JADE

$$E_{JADE} = \sum_i \|\text{diag}(\mathbf{W} \mathbf{F}(\mathbf{M}_i) \mathbf{W})\|^2 \quad (3.16)$$

mientras que es maximizado w.r.t \mathbf{W} (aquí $\|\text{diag}(\cdot)\|^2$ denota la suma de los elementos diagonales).

Para el sistema de las matrices de n éstas dan toda la información relevante sobre los cumulantes, en el sentido que atraviesan el mismo subspace en el tensor de cumulantes. Por ello puede ser demostrado que en tal caso la ecuación 3.16 se convierte en:

$$E_{JADE} = \sum_{iklm \neq iikl} \text{cum}(y_i, y_k, y_l, y_m). \quad (3.17)$$

Por lo tanto, JADE se puede interpretar como un método que reduce al mínimo correlaciones no lineales.

3.4. Algoritmo AMUSE

El algoritmo AMUSE (Algorithm for Multiple Unknown Signals Extraction) utiliza una estructura temporal; aplica estadísticos de segundo orden con el fin de buscar y obtener las componentes independientes. La prime-

ra idea del desarrollo del algoritmo se baso en superar las dificultades que presentan algunos algoritmos de separación ciega de señales basados en estadísticos de cuarto orden cuando las fuentes son gaussianas [88].

El algoritmo AMUSE puede formularse de una manera muy simple:

1. Sea $\mathbf{x}(t)$ blanqueado y sea $C_\tau^{\vec{x}}$ con n diferentes valores propios.
2. Se calcula la descomposición del valor propio de $C_\tau^{\vec{x}}$:

$$C_\tau^{\vec{x}} = \mathbf{W}^T D \mathbf{W} \quad (3.18)$$

con $\mathbf{W} \in O(n)$ y D diagonal.

3. Entonces \mathbf{W} es la matriz de separación:

$$\mathbf{W}^T = \mathbf{W}^{-1} \sim \mathbf{A} \quad (3.19)$$

Sin embargo, la condición que todos los n valores propios existen resulta a menudo estricta y es un un problema. Los valores propios son $\text{Cov}(s_i(t), s_i(T - \tau))$ y tienen que ser muy diferentes, lo cual resulta especialmente problemático con señales que tienen similares espectros de energía. Una solución posible es utilizar múltiples traslados τ ; pero esto también dan lugar a problemas en la diagonalización simultánea.

3.5. Algoritmo fastGEO

Los algoritmos geométricos han recibido en los últimos años una cierta atención debido a su descripción ilustrada y a su facilidad para la puesta en práctica. El acercamiento geométrico fue propuesto primero por [71]. ICA geométrico en una demostración teórica que los puntos fijos de ICA geométrico satisfacen una condición geométrica de la convergencia (GCC), que las imágenes mezcladas de los vectores de la unidad satisfacen también. Esto conduce a una conjetura que demanda que en el caso simétrico unimodal de no gaussianidad, hay solamente un punto fijo estable, implicando la unicidad de ICA geométrico después de la convergencia.

De acuerdo a lo anteriormente descrito, el algoritmo fastGEO es un algoritmo geométrico lineal basado en histogramas. Comparado frente a otros algoritmos geométricos presenta más estabilidad y es más eficiente.

Tenga μ dos ceros los cuales representan el punto estable e inestable de un algoritmo neuronal. En la práctica se extraen los puntos fijos relacionados con la matriz de separación \mathbf{A}^{-1} escogiendo φ_0 tal que:

$$f_Y(l_1(\varphi_0)) + f_Y(l_2(\varphi_0)) \tag{3.20}$$

es máxima. Las distribuciones de las fuentes unimodales y super-gaussianas dan lugar a encontrar un punto fijo estable [84]. Para este caso se pueden

observar distribuciones unimodales:

$$\frac{l_1(\varphi_0) + l_2(\varphi_0)}{2} = \varphi_0 + \frac{\pi}{2} \quad (3.21)$$

y

$$\frac{l_1(\varphi_0) + l_2(\varphi_0) - \pi}{2} = \varphi_0 \quad (3.22)$$

donde se busca α_i .

Para el caso de fuentes sub-gaussianas se elige:

$$f_Y(l_1(\varphi_0)) + f_Y(l_2(\varphi_0)) \quad (3.23)$$

el mínimo índice de la matriz de separación.

3.6. Denoising Source Separation DSS

DSS presenta nuevos algoritmos enfocados en el tema de la separación de fuentes propuesto por Jaakko Särelä y Harri Valpola [81], el cual se basa en procedimientos de eliminación de ruido, presentando como principal ventaja el fácil desarrollo de nuevos algoritmos optimizables para tareas específicas.

Con la introducción de DSS los autores [81] buscan demostrar que es posible construir los algoritmos de separación de fuentes basándose en métodos de eliminación de ruido “denoising”, que tienen como objetivo la búsqueda

da de algoritmos rápidos y exactos en la separación de fuentes. Igualmente, con DSS los autores sugieren que varios de los métodos de separación ya existentes pueden ser formulados en términos de denoising tal cual como se demuestra en [92].

Partiendo del algoritmo de la máxima expectativa (expectation - maximisation EM) [27] como solución a un modelo lineal generativo en [81] los autores interpretan la no linealidad como denoising denominando al algoritmo como DSS. Las características de trabajo de este algoritmo EM se desarrollan en dos pasos, en primer lugar el teorema de Bayes permite el cálculo para la distribución posterior de las fuentes basándose en los datos conocidos y en la matriz de mezcla. Paso seguido, la matriz de mezcla se añade a las nuevas estimaciones. Para más información ver [81].

El algoritmo EM resulta bastante ventajoso debido a que permite el cálculo de componentes simultáneamente o una a una tal como se demuestra en [45], por medio de cuatro pasos donde interviene la probabilidad del cálculo de ruido 3.24, la expectativa de las fuentes 3.25, la revaloración 3.26 y la normalización 3.27:

$$\mathbf{s} = \mathbf{w}^T \mathbf{X} \quad (3.24)$$

$$\mathbf{s}^+ = \mathbf{f}(\mathbf{s}) \quad (3.25)$$

$$\mathbf{w}^+ = \mathbf{X}\mathbf{s}^T \quad (3.26)$$

$$\mathbf{w}_{new} = \frac{\mathbf{w}^+}{\|\mathbf{w}^+\|} \quad (3.27)$$

Särelä y Valpola en [81] interpretan la ecuación 3.25 como denoising, permitiendo determinar todos los algoritmos donde 3.25 como denoising, obteniendo finalmente los algoritmos DSS de la forma de la forma como se expresan en las ecuaciones 3.24, 3.25, 3.26 y 3.27.

Las cuatro ideas básicas del marco DSS desarrollado por Särelä y Valpola son:

- El blanqueamiento de datos, junto con otras formas de normalización crean una especie de aprendizaje sensitivo dirigido a la eliminación del ruido u otras operaciones como por ejemplo la combinación de varios datos.
- DSS resulta en un paquete de programas que se comportan a nivel general de forma flexible, robusta y rápida a la hora que querer analizar grandes cantidades de datos.
- Los algoritmos de DSS implementan una característica de expansión no lineal la cual puede agruparse a fin de conseguir un extractor fuerte de características no lineales.
- El marco de DSS está desarrollado para combinar atención y aprendizaje.

Para más información, un completo desarrollo matemático es llevado aca-

bo en el artículo [81], y en la página web [81].

3.7. Otros algoritmos ICA

Algoritmo ICA de Bell & Sejnowski Generalización del principio de información máxima (InfoMax). Recurre al principio de la máxima entropía como principal estrategia de separación.

InfoMax establece que la transformación de un vector de entradas $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$, en un vector de salidas $\mathbf{z} = (z_1, z_2, \dots, z_n)^T = f(\mathbf{W}\mathbf{x})$, caracterizado por una matriz \mathbf{W} de dimensiones $n \times n$ y una función no lineal acotada e invariante $f = (f_1, f_2, \dots, f_n)^T$, debe ser tal que maximice el promedio de la información mutuamente entre entradas y salidas. La matriz de separación \mathbf{W} representa la transformación de un conjunto de mezclas \mathbf{x} , en un conjunto de señales estadísticamente independientes \mathbf{z} . El método aplicado para estimar esta matriz de separación \mathbf{W} , es el del descenso del gradiente de la función de entropía.

Método geométrico por sectores Introducido por C.G.Puntonet y colaboradores en 1995, trata de explotar las propiedades geométricas y algebraicas tanto de señales originales como de las observaciones. Sigue fundamentos con procedimientos muy sencillos y directos de aplicar, que dan lugar a algoritmos de baja complejidad. La mayor parte de los resultados y conclusiones sobre su convergencia sólo se han obtenido empíricamente, y o no son equi-

variantes o su equivarianza no ha sido probada.

Al estar las fuentes acotadas, los valores generados por las mismas estarán circunscritos en el interior de un rectángulo, que después de la mezcla lineal se transforma en paralelepípedo.

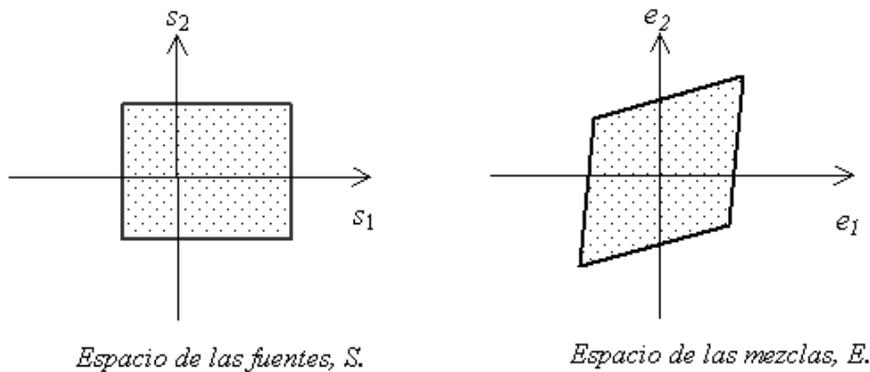


Figura 3.1: Transformación resultante de rectángulo a paralelepípedo a producirse la mezcla.

Para conseguir realizar la separación geométrica deben ser estimadas las pendientes de los bordes del paralelepípedo. Como principales hipótesis usadas para este tipo de separación están:

- Tanto las señales originales como sus transformaciones son desconocidas.
- El número de sensores y fuentes es igual.
- Las fuentes están acotadas, esto se compara con los límites de amplitud de las diferentes señales de radar, voz, sonar, etc.
- Se suponen todos los elementos de la diagonal principal de la matriz A son distintos de cero.

- Igualmente se supone la proximidad de los sensores a las diferentes fuentes.

LatticeIca Método de separación geométrica de señales que estudia el espacio de representación de las señales dividiéndolo en un conjunto de rectángulos, formando un entramado, y se usan el estudio de estas divisiones y sus relaciones para obtener la matriz de mezcla de la separación.

Método de separación de neuronas Método geométrico desarrollado para separar N señales mezcladas mediante una mezcla lineal instantánea. Pretende asociar un punto en cada uno de los extremos de los ejes de las señales, al que se le llama “neurona”, y que permite conocer la pendiente para la matriz de separación. Los cuadrantes que neurona es la que se va a desplazar según el punto en el que se este trabajando.

Algoritmos ICA genéticos Introducidos por John Holland en 1970, inspirándose en el proceso observado en la evolución natural de los seres vivos, métodos estocásticos de búsqueda ciega. El objetivo principal de un algoritmo genético es evolucionar a partir de una población de soluciones para poder dar solución a un problema específico, produciendo nuevas soluciones mejores a las anteriores. A nivel general, cualquier tarea a resolver se puede ver como una búsqueda a través de un espacio de posibles soluciones, donde encontrar la mejor solución se puede convertir en un problema de optimización. Estructura fundamental de un algoritmo genético:

- Forma de iniciar un conjunto de individuos o soluciones en la primera generación.
- Representación genética de posibles soluciones.
- Función que evalúa la calidad de los individuos.
- Operadores que alteren la composición de las nuevas generaciones.
- Uso de distintos parámetros (población, probabilidades, etc), usados por los algoritmos genéticos.

Método de medias ponderadas cuadradas Divide el espacio asociado a las señales que forman la mezcla en hipercuadrantes, en donde se pretende determinar la pendiente en cada uno de los hipercuadrantes, con el fin de obtener la matriz de mezcla. Una de las hipótesis en el que se basa es que las señales de la mezcla tienen el máximo de la distribución de probabilidad se encuentra en el centro de la señal (gaussianas y supergaussianas).

Simulated Annealing Surge del interés en sistemas con un gran número de grados de libertad. Se estudia las integrales de dimensión elevada, en donde la técnica Monte Carlo (evaluación de las integrales solo en algunos puntos aleatorios representativos del sistema), surge como una posible solución. Este algoritmo se basa en la forma mediante la cual, un líquido congela, o un metal cristaliza en el proceso de recocido annealing. Durante el proceso una sustancia a alta temperatura y desordenada, es lentamente enfriada, permitiendo que el sistema se encuentre en estado de equilibrio durante todo el proceso.

Ascenso de colinas dinámico Algoritmo de búsqueda local más conocido, que resulta sencillo y eficiente. El cual se basa en dos bucles, uno interno (técnica eficiente de localización del óptimo local) y otro externo (dirige la actuación del lazo interno). Su idea principal se basa en la selección aleatoria de un estado, cambio al vecino con mejor función de evaluación y si se alcanza un mínimo local estricto, se reinicia el proceso. [17]. Evita problemas como la rigidez del sistema de coordenadas en el cambio entre estados e implementa heurísticas que explotan la búsqueda de óptimos locales.

Algoritmo de búsqueda aleatoria controlada Técnica de búsqueda directa totalmente heurística. Proceso de contracción en un conjunto muestral N de posibles soluciones en X . Esta contracción se consigue reemplazando la peor solución mediante técnicas globales o locales. La técnica global es un proceso iterativo en el que se define el nuevo estado en términos de un subconjunto de puntos pertenecientes al conjunto muestral. [18]

3.8. Conclusiones

Desde sus inicios y hasta la fecha se puede apreciar que la técnica ICA cobra gran interés con el paso de los años. En este capítulo se ha hecho una revisión en algunos de los algoritmos ICA más utilizados, los cuales por sus características presentan un gran interés de estudio. Igualmente se ha visto a nivel general la variedad de algoritmos en la actualidad que recurren a la

técnica ICA con el fin de resolver ciertos problemas reales.

Capítulo 4

El ruido en ICA

El desarrollo de este capítulo pretende mostrar la influencia del ruido en imágenes reales. A nivel general se pueden encontrar imágenes del tipo natural como las que representan bosques, árboles, hojas, etc; e imágenes de estructuras como por ejemplo edificios, puertas, ventanas, etc, que serán las que se trabajaran en esta memoria. Tal y como se mencionó en el anterior capítulo el desarrollo de este trabajo pretende estudiar el ruido como señal parásita y el comportamiento de los algoritmos fastICA, JADE, AMUSE, fastGEO y Denoising ICA bajo la influencia de éste. A nivel general este estudio se llevará a cabo con 15 imágenes diferentes de estructuras, las cuales serán estudiadas primero en su forma original, para después pasar a ser analizadas bajo la influencia de 3 clases diferentes de ruido como son: ruido gaussiano, multiplicativo y de sal y pimienta.

Una definición de ruido al hablar de sistemas electrónicos considera a

éste, como aquella señal ajena a la señal de interés y que es susceptible de provocar un error en el sistema de medida. Así el ruido se puede clasificar según sea una señal determinista o aleatoria, o bien según su origen: externo o interno al sistema de medida. Normalmente se utiliza el término ruido cuando el origen es interno al propio sistema de medida y la naturaleza de la señal suele ser aleatoria. El término interferencia se aplica a aquellas señales externas al sistema de medida, cuya evolución temporal suele seguir una ley preestablecida que puede ser conocida de antemano; aunque su valor en un instante determinado pueda venir caracterizado por una variable aleatoria. Se debe de tener muy presente que la aplicación de ruido implica especificar conceptos de varianza y media.

4.1. Introducción general al análisis

Con el fin de ubicar al lector en un esquema global de la forma de evaluación que se va a realizar en esta sección, se presenta a continuación el diagrama general mostrado en la figura 4.1.

La secuencia general de la información representada en el diagrama se explica a continuación. El primera instancia se toman las imágenes de la matriz de fuentes \mathbf{S} (ver figura 4.2) y se transforman en la matriz de observaciones \mathbf{X} . A estos datos se les agrega ruido, el cual este caso va a ser de 3 clases: gaussiano, multiplicativo y de sal y pimienta. Con ello se va a tener 4 diferentes matrices de observaciones: matriz de observaciones de muestras

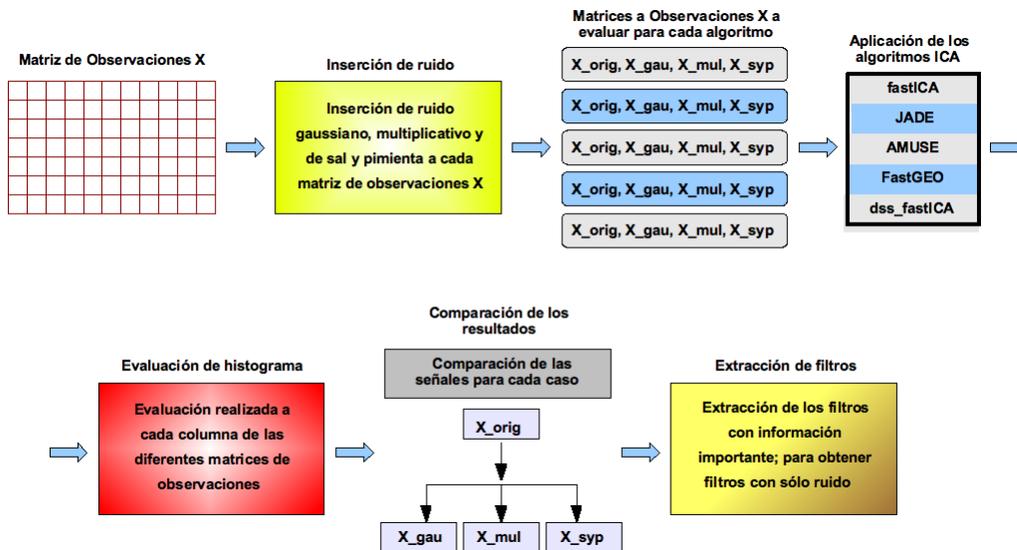


Figura 4.1: Diagrama general. Esquema que introduce a la forma en que se hace el desarrollo de la evaluación en este capítulo.

originales (X_orig), matriz de observaciones contaminadas con ruido gaussiano (X_gau), matriz de observaciones contaminadas con ruido multiplicativo (X_mul) y matriz de observaciones contaminadas con ruido de sal y pimienta (X_syp). Una vez listas las observaciones, se aplican los diferentes algoritmos ICA, para el siguiente paso realizar la evaluación del histograma y la comparación de los diferentes resultados con el fin de llegar a obtener los filtros que contienen sólo ruido.

4.2. Preparación de la información

Descripción general

El trabajar con el algoritmo fastICA implica una manipulación previa de los datos (ver sección 3.2). Del mismo modo, para cada uno de los procesos de información con o sin ruido, se tiene que cada iteración de fastICA entrega diferentes resultados con lo cual, se hace necesario llevar a cabo un análisis de estabilidad elaborando 10 diferentes iteraciones de los datos.

En la literatura existente muchos autores que restringen el conjunto de imágenes naturales a imágenes donde no aparecen representadas estructuras creadas por el hombre, como por ejemplo calles, edificios, etc. A pesar de esta restricción, en el desarrollo de este trabajo las imágenes con las que se va a trabajar van a ser consideradas igualmente en el grupo de imágenes naturales y a las cuales sólo caracterizaremos por propiedades como: reflejos, brillos, sombras, transparencias, colores no uniformes, etc.

El grupo de imágenes de muestra han sido obtenidas con una cámara Fujifilm FinePix S5500, con zoom óptico Fujinon 10x, sensor CCD de 4,0 millones de pixels efectivos, monitor LCD TFT de alta resolución de 115,000 pixel con cobertura del cien por ciento de la imagen. Estas imágenes se encuentran representadas en la figura 4.2 y tal como puede apreciarse son imágenes de intensidad donde los píxeles corresponden a una escala de grises, los cuales representan la brillantes de la imagen. Este tipo de imágenes contiene datos de 8 bits/pixel con lo que se puede tener 256 diferentes niveles de brillantes,

donde el negro se encuentra representado por 0 y el blanco es representado por 255. Igualmente cada una de las imágenes tiene un tamaño de 256 x 256 píxeles. Igualmente, se ve que el ruido que introducir la cámara puede considerarse como insignificante, con lo que se puede decir que las imágenes no poseen ninguna perturbación de ruido.

De forma aleatoria se selecciona la imagen 3 y 7 tal cual como se muestra en la figura 4.3. Esta figura representa 6 sub-imágenes las cuales contienen la información de las imágenes anteriormente pre-seleccionadas y en las cuales puede apreciarse, en la parte superior de la figura se representan las imágenes muestra con la información más una contaminación de ruido gaussiano; la parte del centro representa las misma imágenes muestra, pero esta vez la contaminación se ha llevado a cabo con ruido multiplicativo y las parte inferior muestra la contaminación con ruido de sal y pimienta. A nivel particular, en cada una de estas sub-imágenes se está representando en la primera fila y de izquierda a derecha la imagen de muestra (bien sea 3 o 7), sin ninguna señal parásita, el corte horizontal de esta imagen y su respectiva distribución de histograma. En la segunda fila y también con una organización de derecha a izquierda, se muestran las imágenes muestra contaminadas con ruido, el corte horizontal, que esta representando la influencia que el ruido tiene la imagen y finalmente la distribución del histograma. En la tercera fila de cada sub-imagen se representa sólo una distribución de histograma; este es el error, el resultado obtenido después de realizar la substracción de la información contaminada con ruido menos la información original. Este gráfico es par-



Figura 4.2: Imágenes originales de estructuras.

particularmente interesante debido a que en general, para pre-procesamientos permite al observador hacerse una idea inmediata del comportamiento y naturaleza del ruido presente en los datos a analizar.

Algoritmo fastICA Para inicial se toman las muestras representadas en la figura 4.2, éstas se convertirán en la matriz de observaciones que en un principio, para cada imagen es de de 256 x 256 píxeles. Paso seguido dividimos la imagen en bloques de 8 x 8 píxeles con el fin de buscar los bloques que serán las columnas de la matriz de observaciones \mathbf{X} , a la cual se le podrá aplicar ICA considerando cada una de las columnas como observaciones diferentes, esto es, retomando lo visto en el capítulo 1:

$$\mathbf{X} = \mathbf{AS} \quad (4.1)$$

donde \mathbf{A} son los filtros representados en las columnas y \mathbf{S} son las componentes representadas en las filas las cuales serán tan independientes entre sí como sea posible.

Al ser fastICA un algoritmo que trabaja de forma estadística, en cada iteración las componentes resultantes no guardan igual posición que el proceso anterior, por ejemplo una iteración de fastICA puede entregar el filtro 1 primero en la posición 5 de la matriz de mezcla \mathbf{A} y para la siguiente iteración puede estar este mismo en la posición 20 de la misma matriz. Como solucionar este problema? Este proceso es resuelto con la creación de una matriz de almacenamiento la cual tiene un tamaño de 64 filas por 64 columnas por

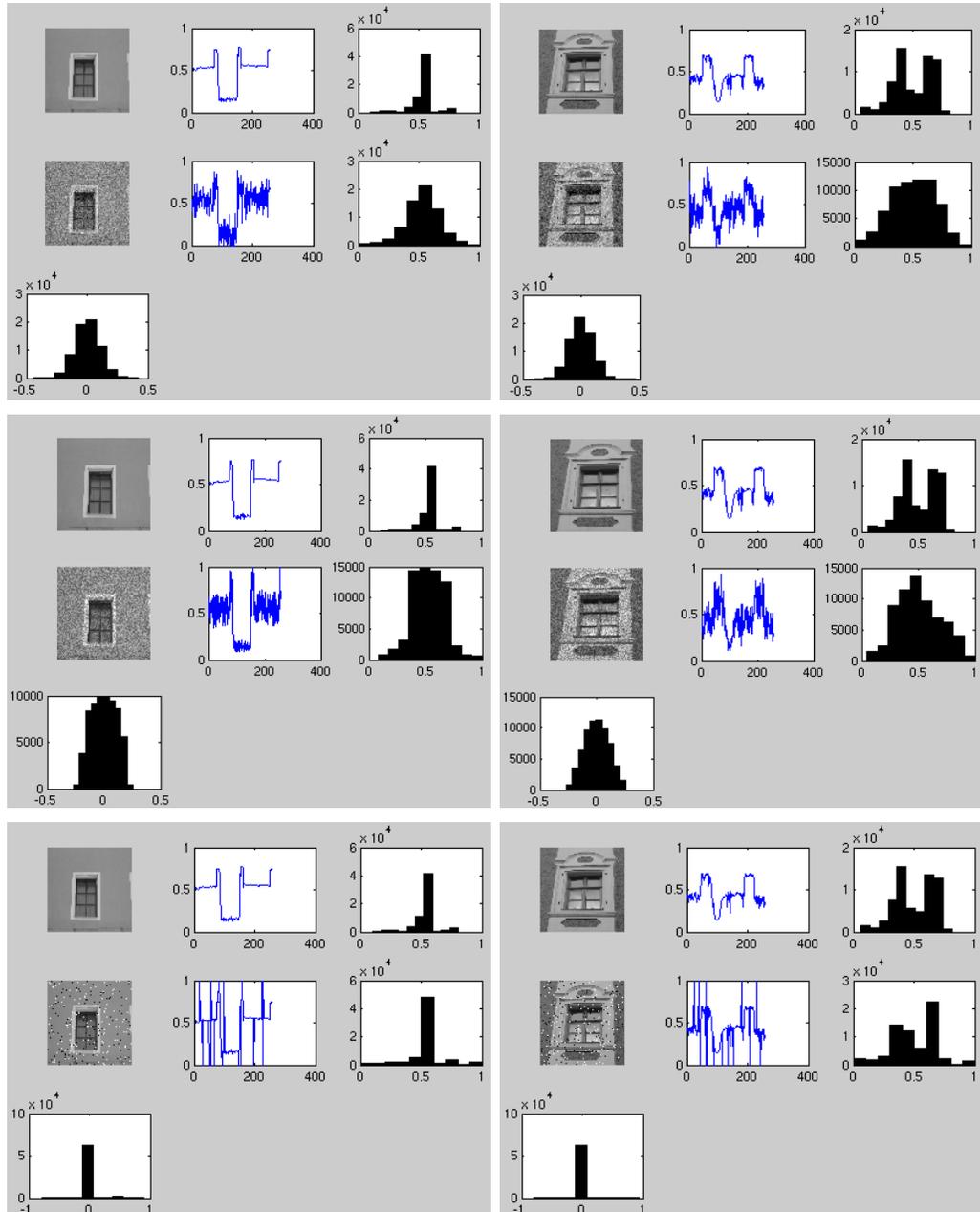


Figura 4.3: Imágenes de estructuras después de la inserción de ruido gaussiano (arriba), multiplicativo (centro) y de sal y pimienta (abajo).

10 de profundidad, o sea, tantas filas como columnas hay en la matriz de mezcla \mathbf{A} , representadas en la grafica 4.4 con las variables j , k y tantas matriz como iteraciones para la prueba de estabilidad que hemos desarrollado, representadas con la variable p .

Con los resultados de las 10 iteraciones se pretende evaluar la correlación entre las imágenes con el fin de conocer que filtros se comportan de manera estable. A nivel teórico una alta correlación implica relación entre filtros, esto es alta dependencia, para nuestro caso se ha seleccionado una correlación mayor o igual a 0,9 con el fin de garantizar la correcta elección de los diferentes filtros, e incluso con este valor de correlación tan elevado se presentan filtros estables que parecer contener ruido.

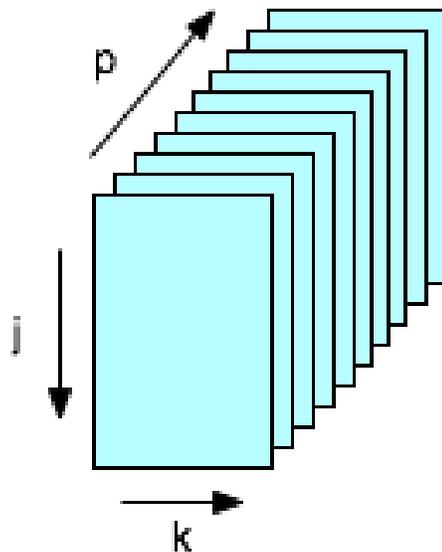


Figura 4.4: Estructura de la matriz de almacenamiento.

Las ventajas que conlleva la creación de esta matriz se pueden apreciar a la hora de llevar a cabo la evaluación del comportamiento de los diferentes filtros de la matriz de mezcla \mathbf{A} , ya que permite conocer donde están las componentes con filtros importantes. Con la datos listos se lleva a cabo el proceso anterior obteniendo una matriz de 64×45 , tal cual como se muestra en la figura 4.5.

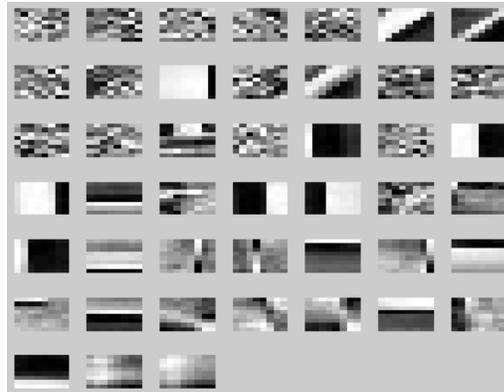


Figura 4.5: Algoritmo fastICA. Matriz de filtros estables resultante de la muestra original. Tamaño de la matriz: 64×45 .

Igualmente se desarrolla para cada uno de los tipos de ruido gaussiano, multiplicativo y de sal y pimienta procesos similares, los cuales arrojan información diversa la cual se puede ver reflejada a continuación en las figuras 4.6 para el caso de ruido gaussiano, 4.7 para el caso de ruido multiplicativo y 4.8 para ruido de sal y pimienta.

Algoritmo JADE El algoritmo JADE se encuentra descrito en la sección 3.3. El proceso llevado a cabo para la organización de la matriz de obser-

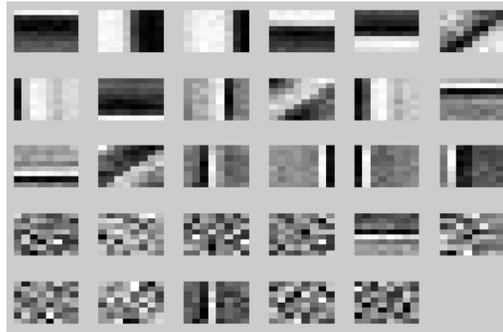


Figura 4.6: Algoritmo fastICA. Matriz de filtros estables resultante de la muestra original con ruido gaussiano. Tamaño de la matriz: 64 x 29.

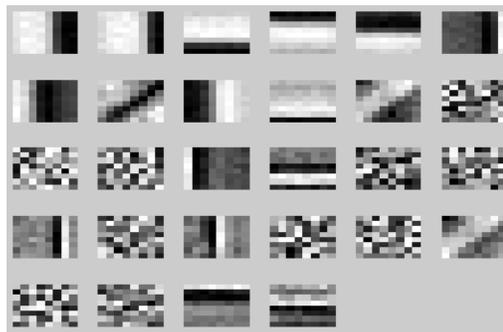


Figura 4.7: Algoritmo fastICA. Matriz de filtros estables resultante de la muestra original con ruido multiplicativo. Tamaño de la matriz: 64 x 28.

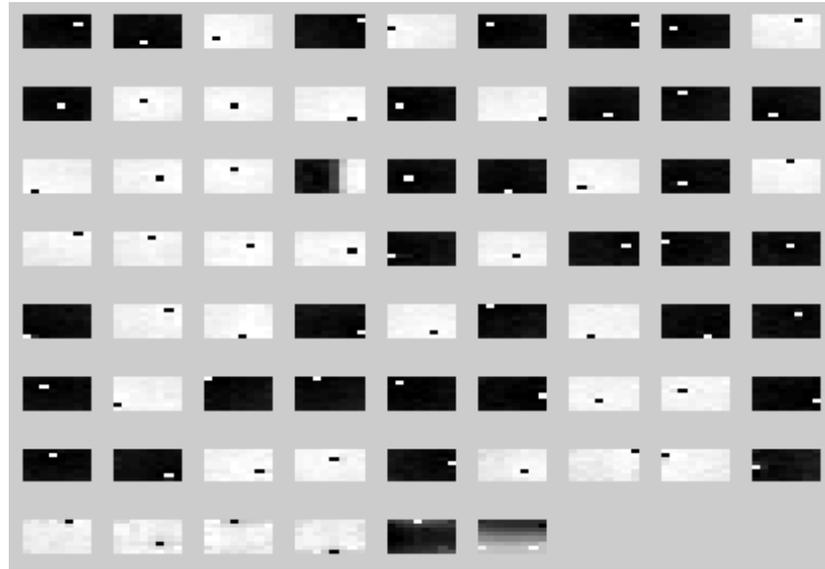


Figura 4.8: Algoritmo fastICA. Matriz de filtros estables resultante de la muestra original con ruido de sal y pimienta. Tamaño de la matriz: 64 x 69.

vaciones en fastICA cumple los mismos criterios para JADE, con lo que nuevamente se trabaja con las mismas muestra y con una matriz de 64 filas por 15360 columnas. Con dicho algoritmo no se hace necesaria la prueba de estabilidad; pero como desventaja del algoritmo es que presenta un tiempo de computo bastante alto comparado con otros.

Con cada unas de las observaciones listas, que en su caso son, observaciones de muestras originales y observaciones con cada una de las clases de ruido utilizadas, el algoritmo inicia su proceso de evaluación arrojando los resultados representados en la figura 4.9. La parte superior izquierda muestra los resultados obtenidos de la evaluación de la información sin ruido, mientras que los resultados arrojados con información con ruido gaussiano se mues-

tran en la parte superior derecha. La información representada en la parte inferior izquierda y derecha, respectivamente corresponden a las muestras contaminadas con ruido multiplicativo y de sal y pimienta.

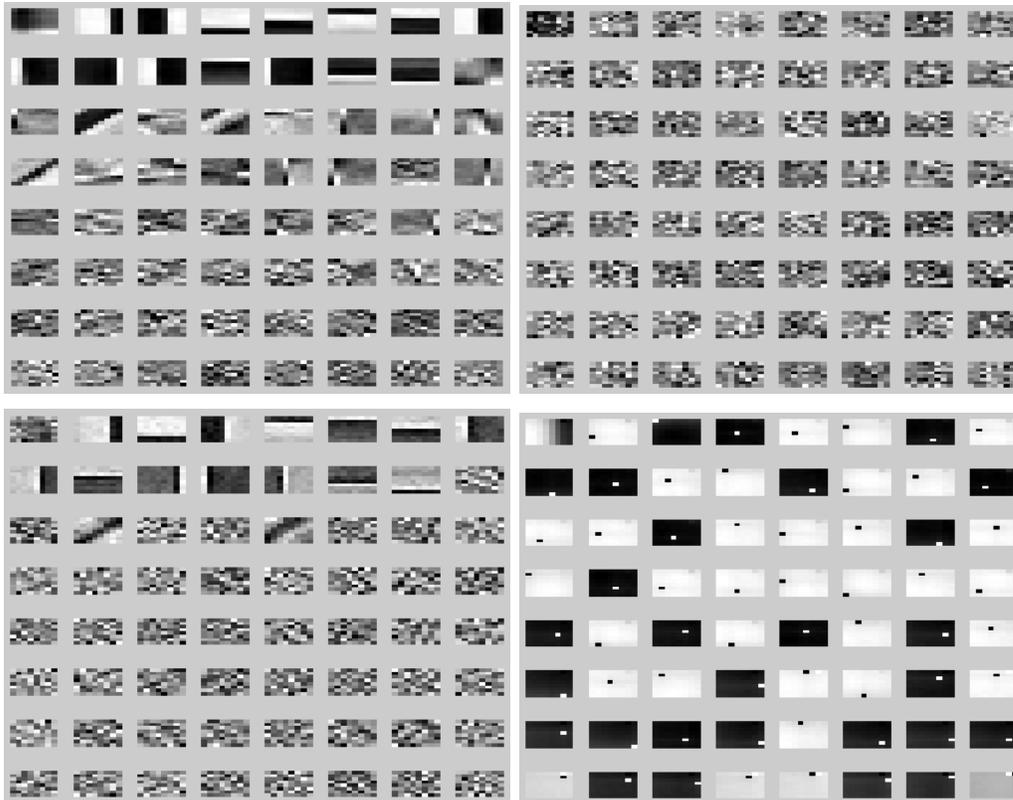


Figura 4.9: Algoritmo JADE. Matriz de mezcla \mathbf{A} resultante después de la evaluación de las muestras sin ninguna perturbación (esquina superior izquierda), muestras con inserción de ruido gaussiano (esquina superior derecha), muestras con ruido multiplicativo (esquina inferior izquierda) y muestras con ruido de sal y pimienta (esquina inferior derecha).

Algoritmo AMUSE En la sección 3.4 se describe el algoritmo AMUSE.

Con la matriz de observaciones que se ha trabajado hasta el momento, se

evalúan los resultados de los datos con la información original y con ruido. Todos ellos representados en la figura 4.10. Al contrario de JADE, AMUSE se presenta como un algoritmo relativamente rápido.

Algoritmo fastGEO Fastgeo se encuentra descrito en la sección 3.5. Las imágenes 4.11, contienen los resultados de aplicar fastGEO a la matriz de observaciones \mathbf{X} descrita con anterioridad.

Denoising ICA En 3.6 se describe Denoising ICA. Como se ha mencionado con anterioridad, DSS se basa en procedimientos de eliminación de ruido. Tal y como explican los autores en [81], DSS cuenta con diferentes paquetes. En especial para el desarrollo de este trabajo se trabajo con `dss_fastica` con el fin de poder evaluar al final resultados.

Al igual que para el algoritmo fastICA desarrollado en 4.2, para `dss_fastica` se hizo también necesario recurrir a un análisis de estabilidad, el cual se llevo a cabo para los datos de muestras originales, y de muestras con ruido gaussiano, ruido multiplicativo y ruido de sal y pimienta. Los resultados obtenidos pueden observarse en las imágenes 4.12, 4.13, 4.14 y 4.15 respectivamente.

4.2.1. Discusión de resultados

En esta primera parte se han organizado las respectivas matrices de observaciones \mathbf{X} tal cual como se ha descrito en la sección 4.2 con el fin de ser entregadas a cada uno de los algoritmos con los que se esta llevando a

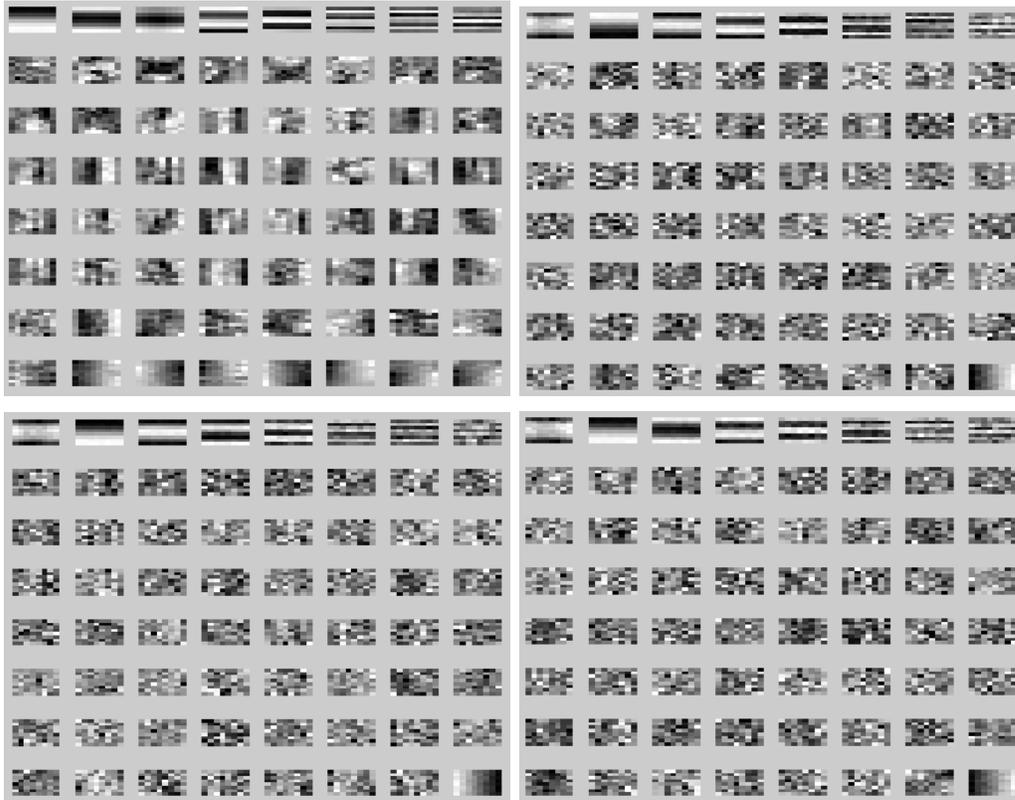


Figura 4.10: Algoritmo AMUSE: Resultado de la matriz de mezcla \mathbf{A} con la evaluación de las muestras sin ninguna perturbación. Imagen superior izquierda. Tamaño de la matriz: 64×64 . Resultado de la matriz de mezcla \mathbf{A} con la evaluación de las muestras original con ruido gaussiano. Imagen superior derecha. Tamaño de la matriz: 64×64 . Resultado de la matriz de mezcla \mathbf{A} con la evaluación de las muestras original con ruido multiplicativo. Imagen inferior izquierda. Tamaño de la matriz: 64×64 . Resultado de la matriz de mezcla \mathbf{A} con la evaluación de las muestras original con ruido de sal y pimienta. Imagen inferior derecha. Tamaño de la matriz: 64×64 .

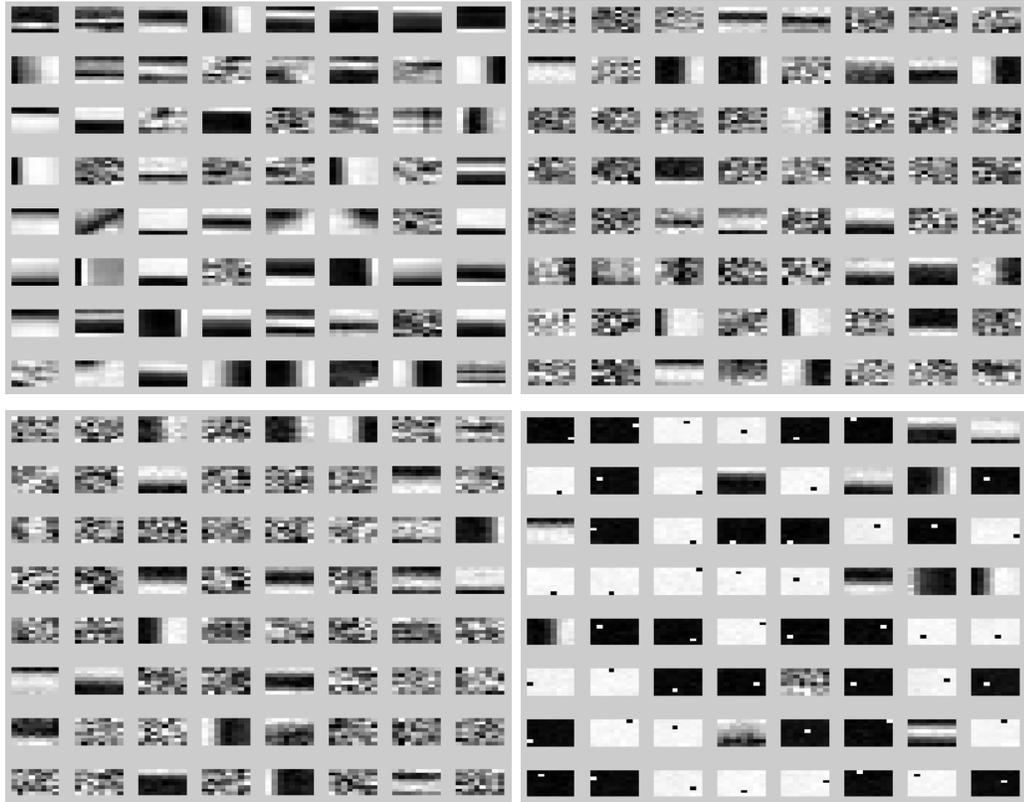


Figura 4.11: Algoritmo fastGEO. Resultado de la matriz de mezcla \mathbf{A} con la evaluación de las muestras sin ninguna perturbación. Imagen superior izquierda. Tamaño de la matriz: 64 x 64. Resultado de la matriz de mezcla \mathbf{A} con la evaluación de las muestras original con ruido gaussiano. Imagen superior derecha. Tamaño de la matriz: 64 x 64. Resultado de la matriz de mezcla \mathbf{A} con la evaluación de las muestras original con ruido multiplicativo. Imagen inferior izquierda. Tamaño de la matriz: 64 x 64. Resultado de la matriz de mezcla \mathbf{A} con la evaluación de las muestras original con ruido de sal y pimienta. Imagen inferior derecha. Tamaño de la matriz: 64 x 64.

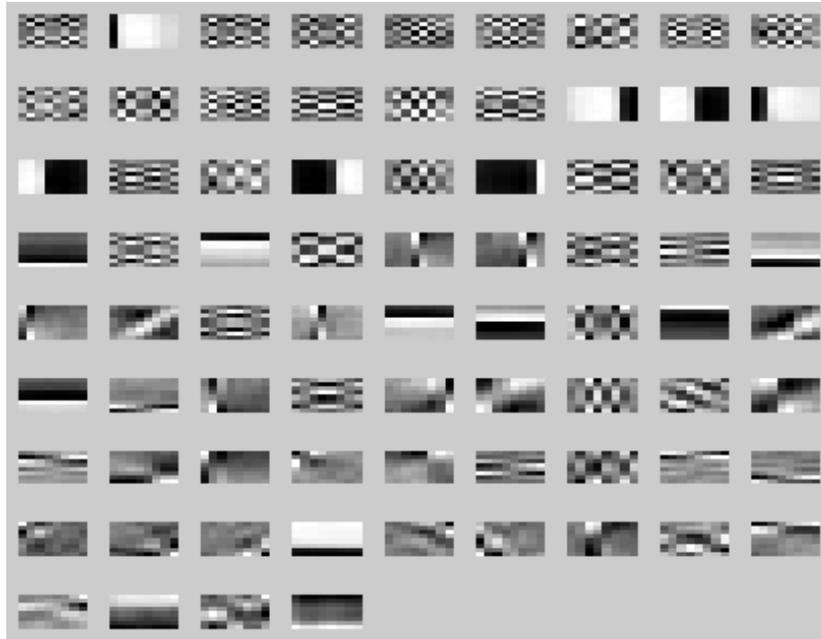


Figura 4.12: dss_fastICA. Matriz de filtros estables resultante de la muestra original. Tamaño de la matriz: 64 x 76.

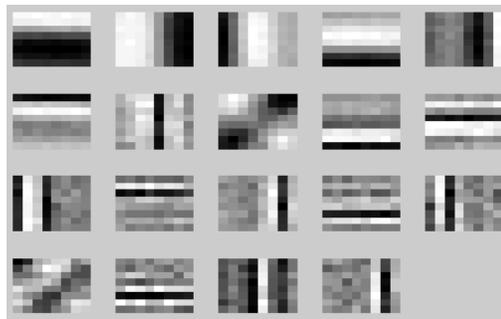


Figura 4.13: dss_fastICA. Matriz de filtros estables resultante de la muestra original con ruido gaussiano. Tamaño de la matriz: 64 x 19.

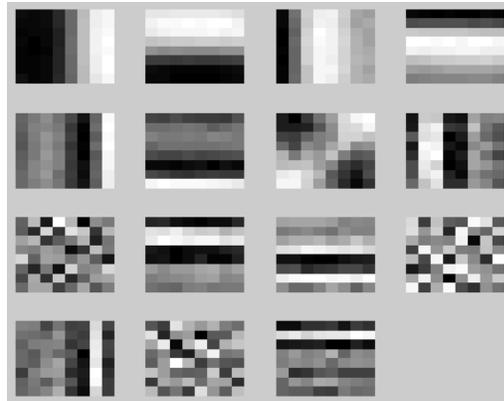


Figura 4.14: `dss_fastICA`. Matriz de filtros estables resultante de la muestra original con ruido multiplicativo. Tamaño de la matriz: 64 x 15.

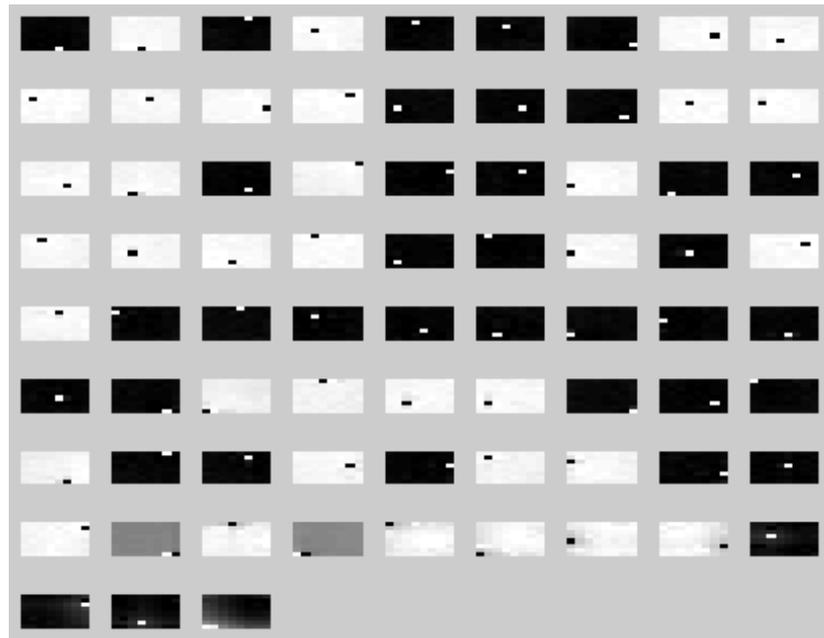


Figura 4.15: `dss_fastICA`. Matriz de filtros estables resultante de la muestra original con ruido de sal y pimienta. Tamaño de la matriz: 64 x 75.

cabo este análisis comparativo. Estos filtros obtenidos representan lo que los investigadores Bell y Sejnowski [15], definían como una conexión entre ICA aplicado a imágenes naturales y el comportamiento de ciertas neuronas de la corteza visual primaria. Igualmente, las investigaciones llevadas a cabo por Barlow [8], [9] y [10], sobre el proceso que llevan a cabo distintos sistemas sensoriales del cuerpo humano en cuanto a la reducción de redundancia se suma a la idea que el sistema visual podría realizar una especie de análisis de componentes independientes de la información que llega a la retina. Tal cual como se describía con anterioridad en el capítulo 2, sección 2.1.1, el cerebro no llega a ver una imagen definida; ya que las fibras nerviosas conectadas a la retina ven una parte de la escena global.

Para Barlow, ésta reducción de redundancia (caracterizada por una dependencia estadística entre estímulos, los cuales son minimizados para después ser combinados los de más dependencia), estaba representada en patrones repetitivos que distinguen estímulos de interés de datos aleatorios y el reconocimiento de estos estímulos permite al cerebro la elaboración de diferentes modelos los cuales vienen determinados de acuerdo al entorno de nuestra vida diaria.

Sobre las imágenes que venimos discutiendo, puede verse lo descrito por Hubel y Wiesel cuando explicaban que los detectores del sistema visual miran sólo a una pequeña parte del campo visual, demostrando que las neuronas de la corteza visual primaria estaban selectivamente sintonizadas para responder a bordes orientados en ángulos diversos tal cual y como se describe en

la figura 4.16. De esta manera puede decirse que para el caso visual, líneas con cierta orientación son usadas como estímulos, permitiendo a las neuronas responder a estos estímulos. Otra característica muy importante que se demuestra con estos resultados es que en general, las imágenes están formadas por segmentos los cuales tienen diferentes longitudes y distintas orientaciones que van variando, con lo cual se puede decir que la mayor parte de la información por la que se encuentra formada en conjunto una imagen esta compuesta de bordes y líneas.

Para la prueba de estabilidad llevada a cabo con las imágenes en los algoritmos de `fastICA` y `dss.fastICA`, nos hemos basado en esta idea de Barlow con el fin de reducir la redundancia en los datos. Una vez desarrollado este proceso y de acuerdo a los resultados obtenidos para cada una de las respuestas de los algoritmos, se puede observar en las imágenes (4.5, 4.6, 4.7, 4.8, 4.9, 4.10 y 4.11), donde la información que se encuentra representada es obtenida de las diferentes columnas de cada matriz de mezcla \mathbf{A} , están representando diferentes patrones los cuales en su mayoría, tienen un gran parecido con los campos receptivos observados por el sistema visual humano por la forma de sus estructuras localizadas y orientadas. La información representada en las imágenes 4.12, 4.13, 4.14 y 4.15, esta muestra igualmente los filtros obtenidos; pero con la previa aplicación de PCA (análisis de componentes principales) de acuerdo a las propiedades que maneja el algoritmo de `dss.fastICA`. De la misma forma que con la información anterior en este caso se reflejan también patrones que parecen representar algún tipo de borde o esquina.

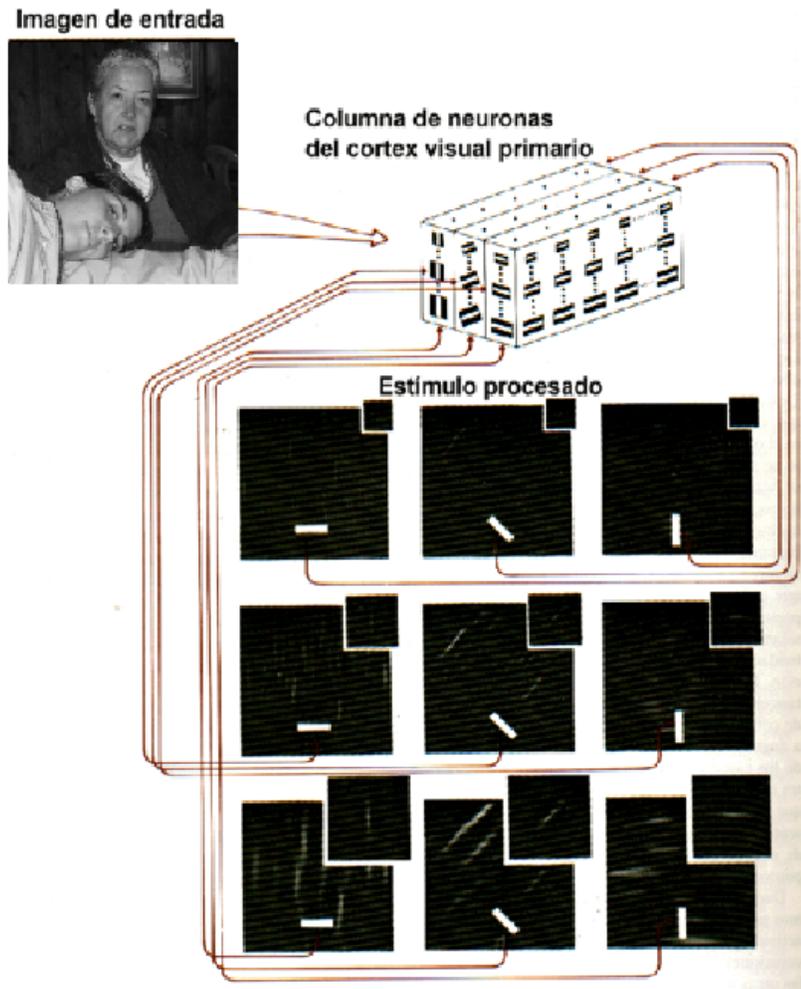


Figura 4.16: Descripción de las neuronas de la corteza visual primaria respondiendo selectivamente a bordes orientados en ángulos diversos. Imagen adaptada de la página web: <http://www.ucm.es/info/pslogica/mente/topo.htm>

Otro punto interesante que vale la pena destacar en este primer paso es lo ineficientes que parecen comportarse la mayoría de los algoritmos aquí trabajados (a excepción de AMUSE), cuando se presentan datos contaminados con ruido impulsivo de sal y pimienta, cosa que hablaremos en la sección 4.3.

4.3. Evaluación de resultados

El objetivo principal del presente trabajo es conocer como se comportan las diferentes clases de ruido y su influencia en las muestras originales. Teniendo como base principal la información recolectada en la sección anterior 4.2, el paso siguiente pretende evaluar los datos para cada un de los algoritmos así como realizar comparaciones con el fin de determinar las características de ruido.

En primera instancia se recurre a la evaluación del histograma para saber como se comporta cada uno de los filtros de las diferentes matrices de mezcla \mathbf{A} obtenidas. Se quiere analizar de forma independiente el comportamiento del ruido. A nivel particular, existen distribuciones propia de identificación para cada tipo de ruido.

Ruido Gaussiano. Para el caso de distribuciones gaussianas, se presenta como característica que la función de densidad de probabilidad tiene una concentración de sus valores más altos alrededor de la media, mientras que hacia las colas decrece de forma suave. La función de densidad de probabilidad de

una distribución gaussiana esta determinada por la función representada en la ecuación 4.2.

$$p_x(x) = \frac{1}{\sqrt{2\pi\sigma_x^2}} \exp \frac{(x - m_x)^2}{-2\sigma_x^2} \quad (4.2)$$

El ruido gaussiano en las imágenes de muestra se ha obtenido de la función “imnoise” de matlab y se ha trabajado con ruido blanco gaussiano de media 0 y varianza 0.01. De la figura 4.3 parte superior, se extraen las dos funciones de densidad de probabilidad gaussianas, obtenidas de las imágenes 3 y 7 respectivamente, en éstas puede observarse que efectivamente cumplen con una función gaussiana.

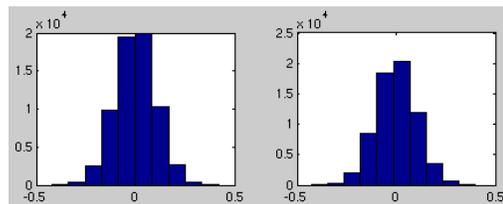


Figura 4.17: Características típicas de una función gaussina, obtenidas de las imágenes de muestras 3 y 7 después de la inserción del ruido gaussiano.

Ruido Multiplicativo. Para el ruido multiplicativo al igual que en el caso anterior, recurre a “imnoise” en matlab el cual trabaja usando la ecuación representada en 4.3 donde n es ruido aleatorio uniformemente distribuido con media 0 y la varianza v , la cual es para nuestro caso de 0.04.

$$J = I + nI \quad (4.3)$$

De la figura 4.3 parte media, se extraen para las gráficas de las muestras originales 3 y 7 el error obtenido después de la adición de ruido multiplicativo. Como puede verse en la figura 4.18, éstas parecen comportarse de forma subgaussiana.

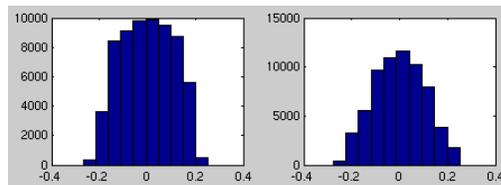


Figura 4.18: Función de densidad obtenidas de las imágenes de muestras 3 y 7 después de la inserción del ruido multiplicativo.

Ruido Impulsivo: Sal y Pimienta. Las características principales con las que trabaja “imnoise” para agregar ruido de sal y pimienta a una imagen se basan en las características de la densidad de ruido, lo cual afecta a los píxeles contenidos en $d * prod(size(I))$. Para nuestro caso se trabajó con una densidad de ruido de 0.05. De la figura 4.3 parte inferior, para las gráficas de las muestras originales 3 y 7 se extrae el error obtenido después de aplicar ruido de sal y pimienta. Como puede verse en la figura 4.19.

4.3.1. Evaluación de resultados presentados con fastICA

Desarrollada la evaluación con el análisis del histograma para cada filtro obtenido de la matriz de mezcla \mathbf{A} se tiene los resultados mostrados en

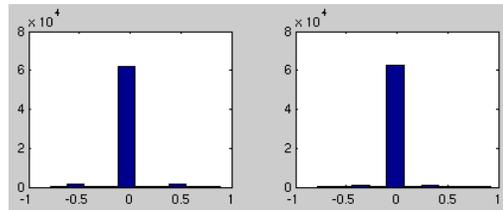


Figura 4.19: Función de densidad obtenidas de las imágenes de muestras 3 y 7 después de la inserción del ruido de sal y pimienta.

4.20, para señales originales, 4.21 para señales con ruido gaussiano, 4.22 para señales con ruido multiplicativo y 4.23 para señales con ruido de sal y pimienta.

Los filtros de las columnas de la matriz de mezcla \mathbf{A} que se representan en las gráficas 4.20, 4.21, 4.22 y 4.23 tienen un valor de interés a la hora de querer evaluar el comportamiento del ruido con el fin de conocer de que clase es y de catalogarlo para un futuro preprocesamiento. Propiamente, en la figura 4.21 puede observarse en los filtros de la matriz de mezcla \mathbf{A} que representan ruido describen una función de distribución gaussiana típica, ocurriendo lo mismo para las otras clases de ruido y sin perder sus características.

Ahora, que pasa cuando comparamos la señal original con cada una de las diferentes señales con ruido? Que filtros siguen presentándose estables? Para este proceso empezamos comparando la señal original con la señal de ruido gaussiano; la figura 4.24 representa los datos obtenidos como ejemplo los ocho primeros filtros que presentan alta correlación. Como puede verse esta información de comparación esta mostrando datos que representan bordes o esquinas que describen información importante. Cada una de las

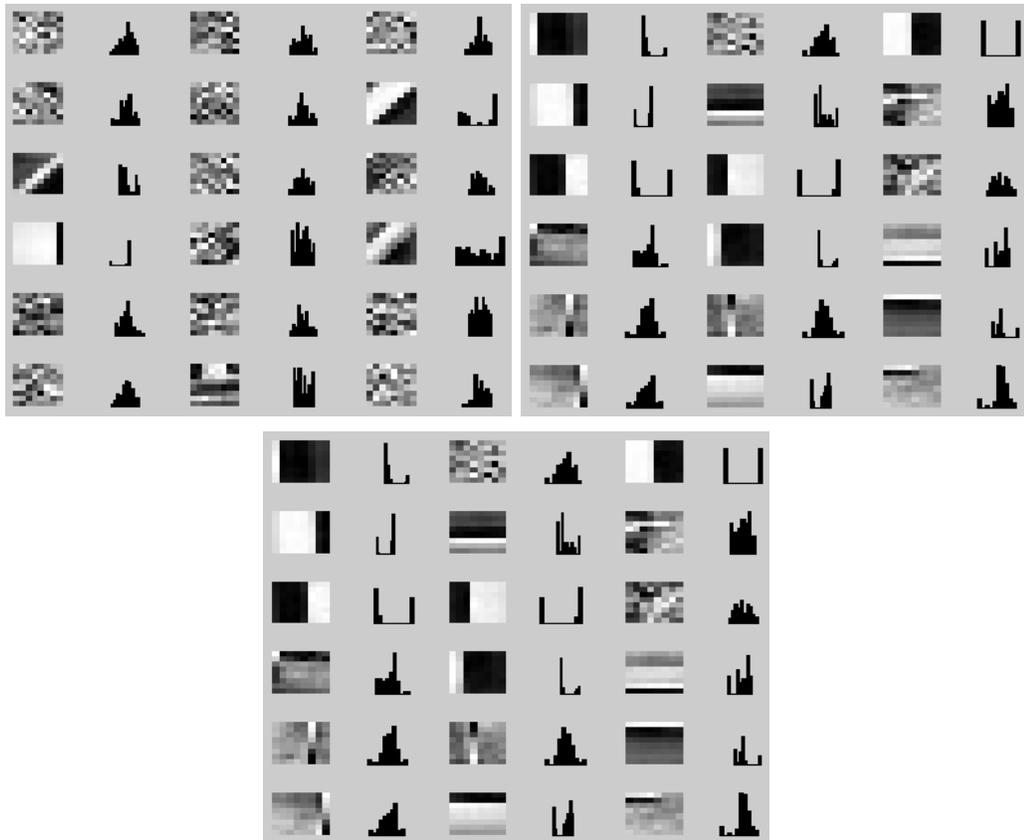


Figura 4.20: Algoritmo fastICA. Evaluación de cada columna de A para datos sin ruido.

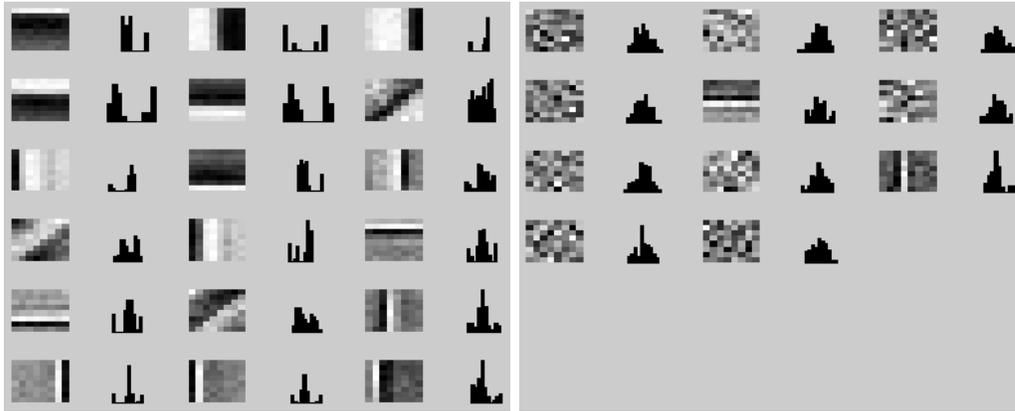


Figura 4.21: Algoritmo fastICA. Evaluación de cada columna de \mathbf{A} para datos con ruido gaussiano.

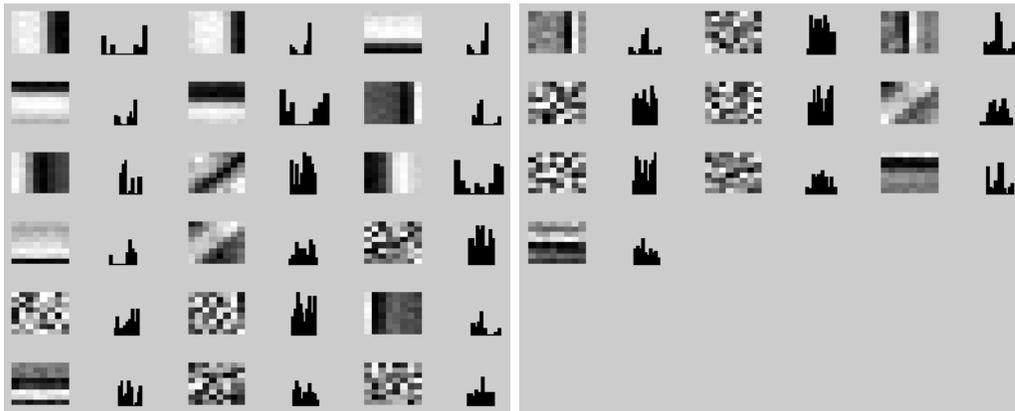


Figura 4.22: Algoritmo fastICA. Evaluación de cada columna de \mathbf{A} para datos con ruido multiplicativo.

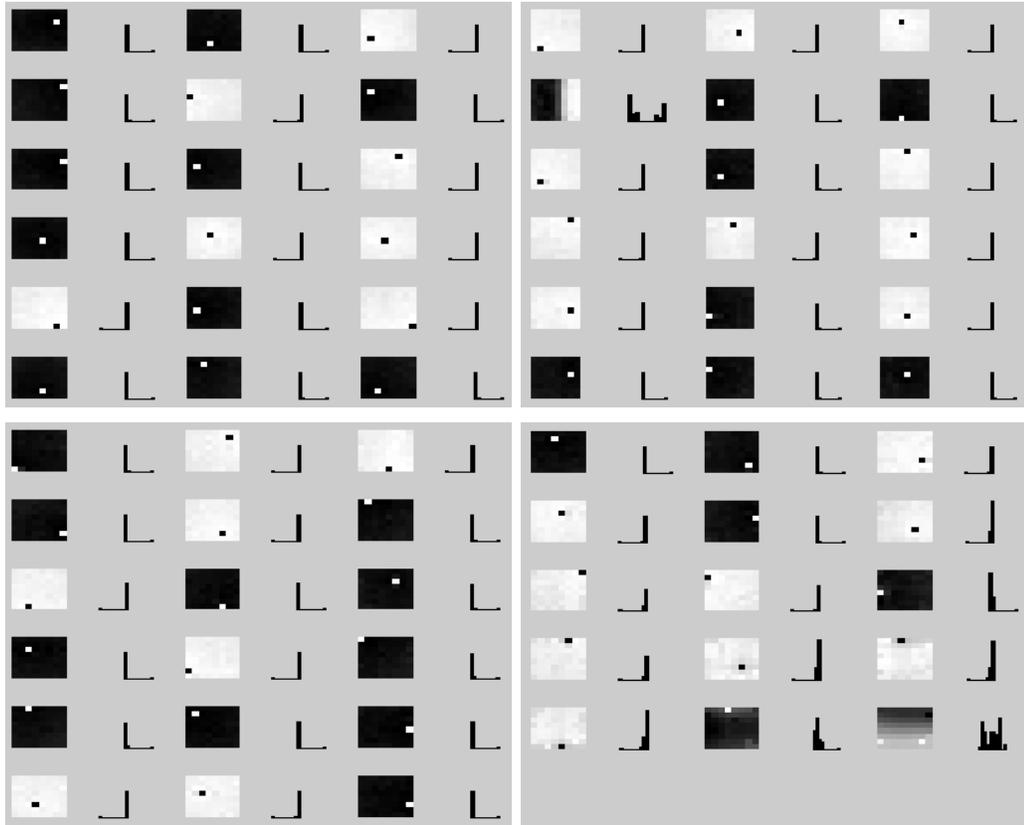


Figura 4.23: Algoritmo fastICA. Evaluación de cada columna de \mathbf{A} para datos con ruido de sal y pimienta.

sub-imágenes representadas en esta figura muestran la comparación llevada a cabo entre la matriz de mezcla \mathbf{A} de las muestras (parte superior izquierda), con su respectiva distribución de histograma y la matriz de mezcla \mathbf{A} de la información con ruido gaussiano (parte superior derecha), igualmente con su distribución de histograma. En la parte inferior de cada sub-imagen se representa el resultado de realizar la sustracción de la señal sin ruido y la señal con ruido.

Las figuras 4.25 y 4.26 muestran los resultados obtenidos para la comparación entre la señal original y la señal con ruido multiplicativo y ruido de sal y pimienta respectivamente. Estos resultados también muestran como ejemplo cada los primeros ocho filtros. Vale la pena mencionar nuevamente que la información que en un principio se está obteniendo es información con alta correlación, esto es información con datos importantes que no contienen ruido.

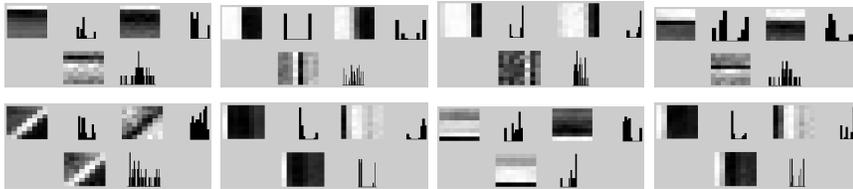


Figura 4.24: Algoritmo fastICA. Filtros de componentes entre la señal original y la señal con ruido gaussiano de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación.

En un principio para este proceso se comenzó tomando valores de correlación por encima de 0,9 entre las señales original y con ruido, con lo cual

se observo que en el resultado final todavía se conservaban filtros en la señal gaussiana que representaban alguna característica importante como un borde. Con ello se bajo el nivel de búsqueda entre la comparación hasta hasta llegar a una correlación de 0,5, valor con el cual se llega al objetivo de obtener en su mayoría, filtros que representan solamente ruido.

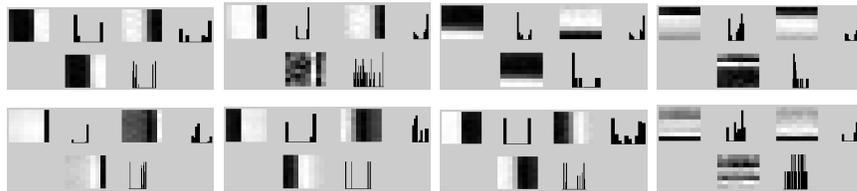


Figura 4.25: Algoritmo fastICA. Filtros de componentes entre la señal original y la señal con ruido multiplicativo de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación.

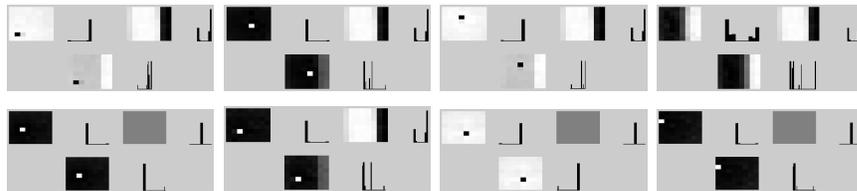


Figura 4.26: Algoritmo fastICA. Filtros de componentes entre la señal original y la señal con ruido de sal y pimienta de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación.

Ahora bien, una vez obtenidas cada una de las comparaciones entre las señales, el siguiente paso es eliminar de cada matriz de mezcla \mathbf{A} estas componentes para poder quedarnos con filtros que en su mayoría debes de mostrarnos solamente ruido. Con esta información se puede evaluar si cada tipo

de ruido insertado en las muestras se comporta como describe su propia distribución típica representadas en las figuras 4.17, 4.18 y 4.19.

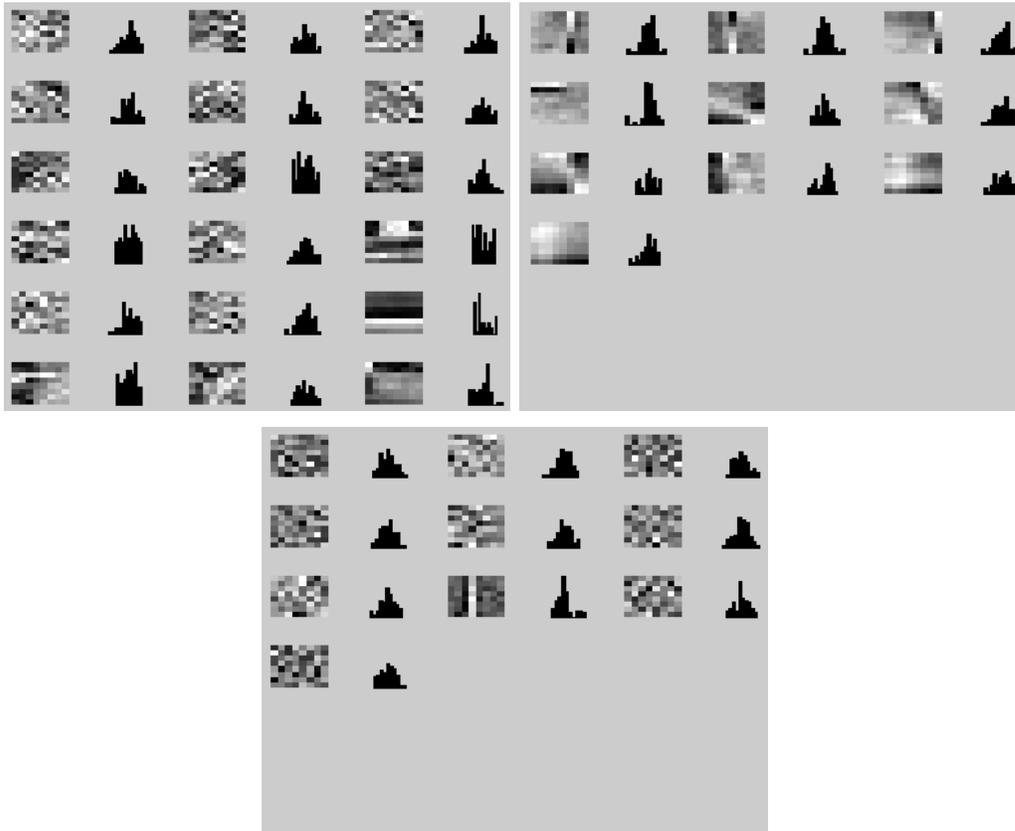


Figura 4.27: Algoritmo fastICA. Resultado de comparar la matriz de mezclas originales sin perturbaciones (parte superior) con la matriz de mezclas con ruido gaussiano (parte inferior).

En la figura 4.27 se presentan los resultados obtenidos de las comparación entre la matriz de mezcla no contaminada, esto es, matriz de datos originales, con la matriz de mezcla con ruido gaussiano. En la parte superior de esta imagen se representan los filtros resultantes de la muestras originales, mien-

tras que la parte inferior contienen la información de los filtros resultantes de la información a la que en un principio se le fue añadido ruido gaussiano. En esta primera muestra de resultados puede verse claramente como las columnas de la matriz de mezcla presentan una distribución típica de ruido gaussiano en prácticamente todas sus componentes.

La siguiente pregunta que surge es, que pasa con las otras clases de ruido? Las figuras 4.28 y 4.29, muestran los resultados obtenidos. Al igual que para el caso anterior se pueden observar que el comportamiento para cada uno de los casos sigue con la característica típica de distribución. Para el caso de ruido de sal y pimienta se observa como punto adicional que el algoritmo fastICA parece presentar problemas con esta clase de ruido, ya que estos resultados de obtener cuadros en su totalidad blancos o negros con un punto diferente, son típicos cuando el algoritmo solamente trabaja con datos de ruido.

En concreto, en cuanto a los resultados obtenidos en 4.29 con ruido de sal y pimienta, como saber con seguridad que la información contenida en cada uno de los filtros finales no presenta alguna característica importante? Para este proceso se busca en cada uno de los filtros de la matriz de mezcla \mathbf{A} los máximos valores de las mismas y las respectivas posiciones que ocupan, paso seguido son colocan estos puntos en 0 con lo cual se obtiene la información representada en la figura 4.30, donde se puede observar que la información disponible no presenta ninguna característica de bordes o esquinas. Este es un resultado esperado en esta parte del proceso ya que las componentes con alta correlación han sido retiradas con anterioridad.

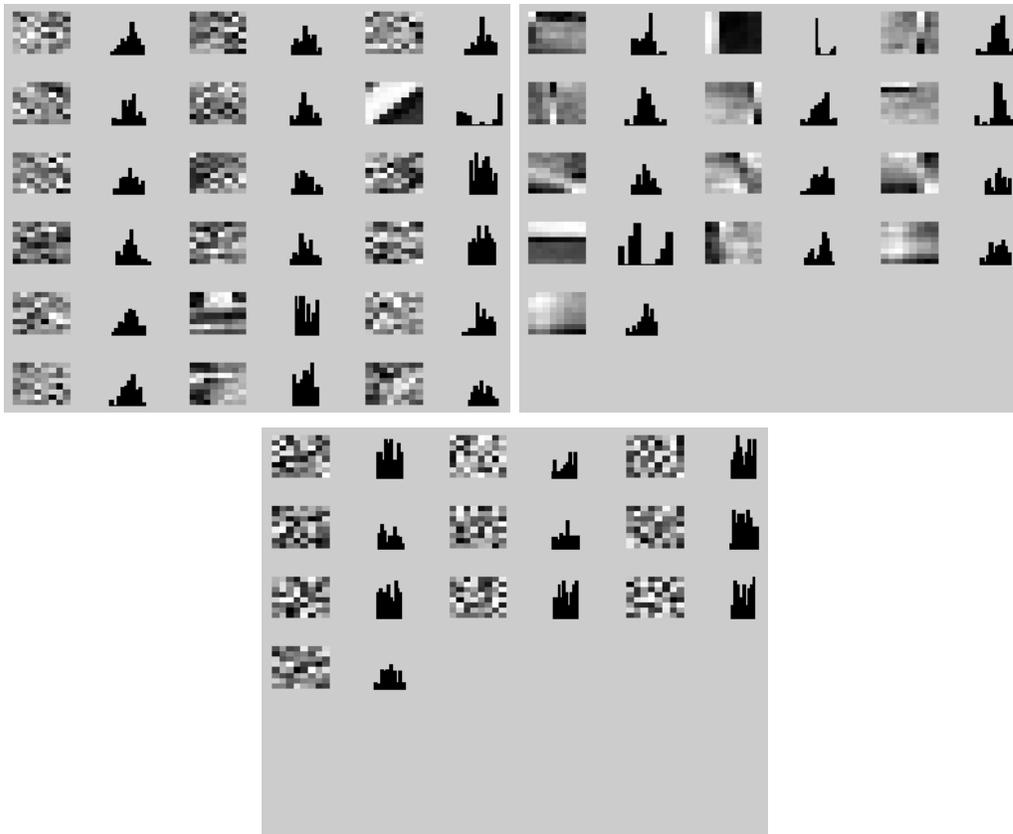


Figura 4.28: Algoritmo fastICA. Resultado de comparar la matriz de mezclas originales sin perturbaciones (parte superior) con la matriz de mezclas con ruido multiplicativo (parte inferior).

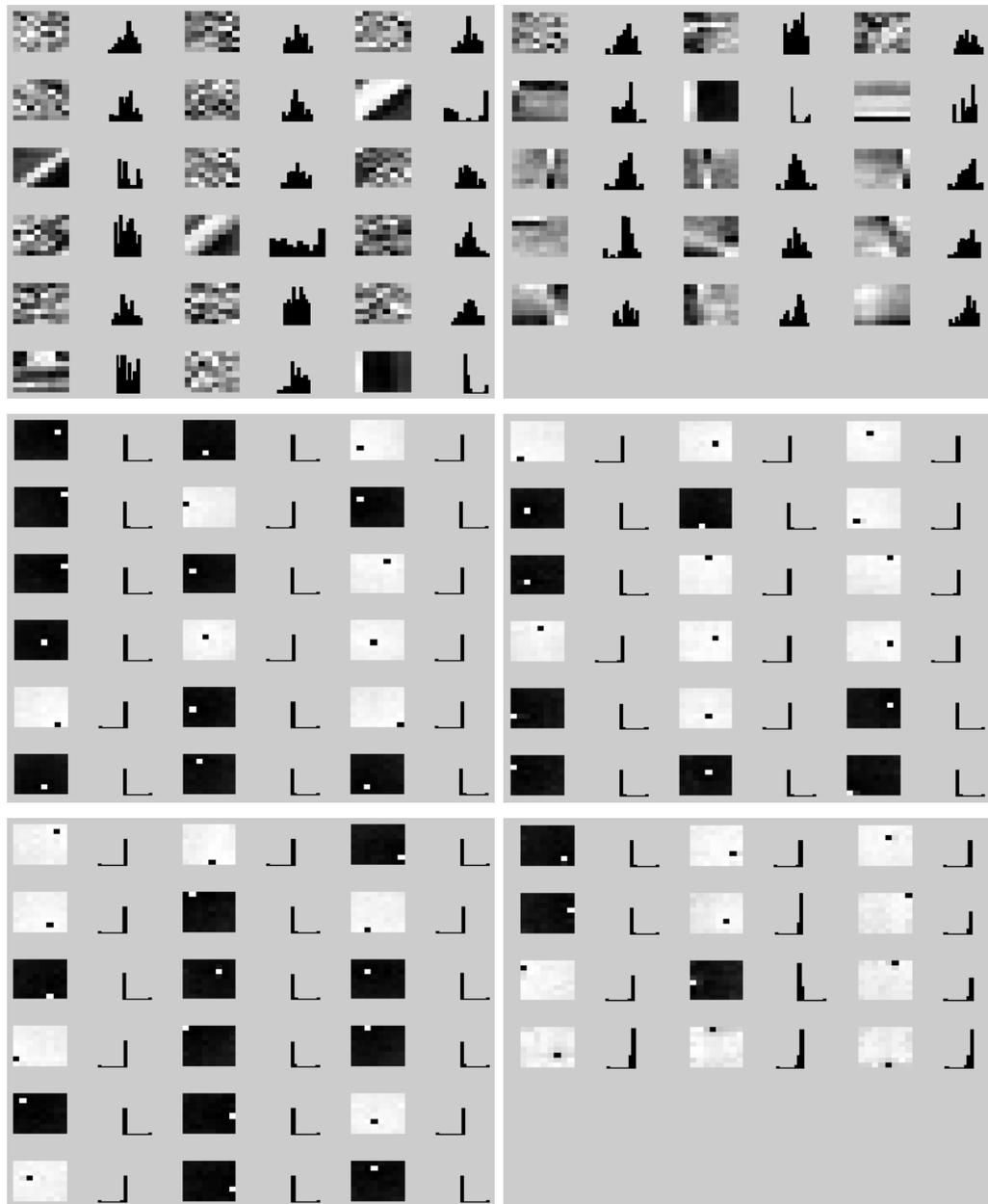


Figura 4.29: Algoritmo fastICA. Resultado de comparar la matriz de mezclas originales sin perturbaciones (parte superior) con la matriz de mezclas con ruido de sal y pimienta (parte media e inferior).

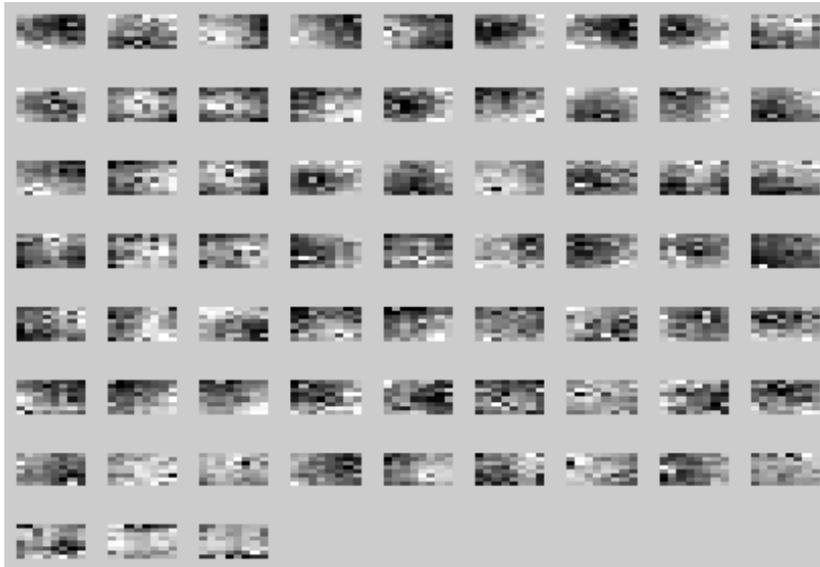


Figura 4.30: Algoritmo fastICA. Resultado de poner 0 en la matriz de filtros \mathbf{A} con ruido de sal y pimienta los puntos con mayor valor.

El desarrollo de este punto y presentación de los resultados plantea un nuevo interrogante referente a los resultados obtenidos en la sección 4.2, con los primeros resultados mostrados en la figura 4.8. Que pasa si en estos resultados de la figura 4.8 le aplicamos el mismo proceso? Como respuesta interesante se obtiene la figura 4.31, en la cual puede observarse similar comportamiento que en el caso anterior, con información que en ningún momento parece representar algún dato de interés, a excepción de las columnas 22, 68 y 69, las cuales son eliminadas en el proceso previo.

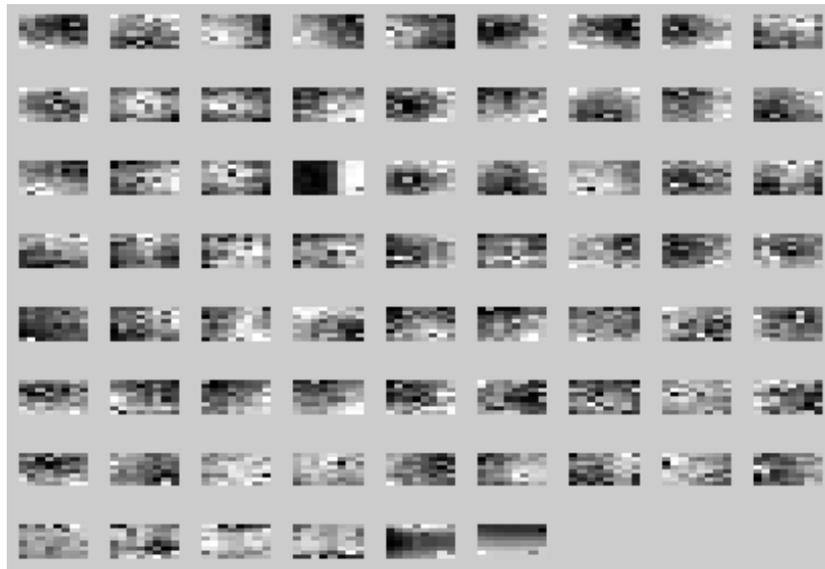


Figura 4.31: Algoritmo fastICA. Resultado de poner 0 en la matriz de filtros \mathbf{A} con ruido de sal y pimienta presentada en la figura 4.8, los puntos con mayor valor.

4.3.2. Evaluación de resultados presentados con JADE

La evaluación de los resultados presentados por el algoritmo JADE se presentan en las figuras 4.32, 4.33, 4.34 y 4.35.

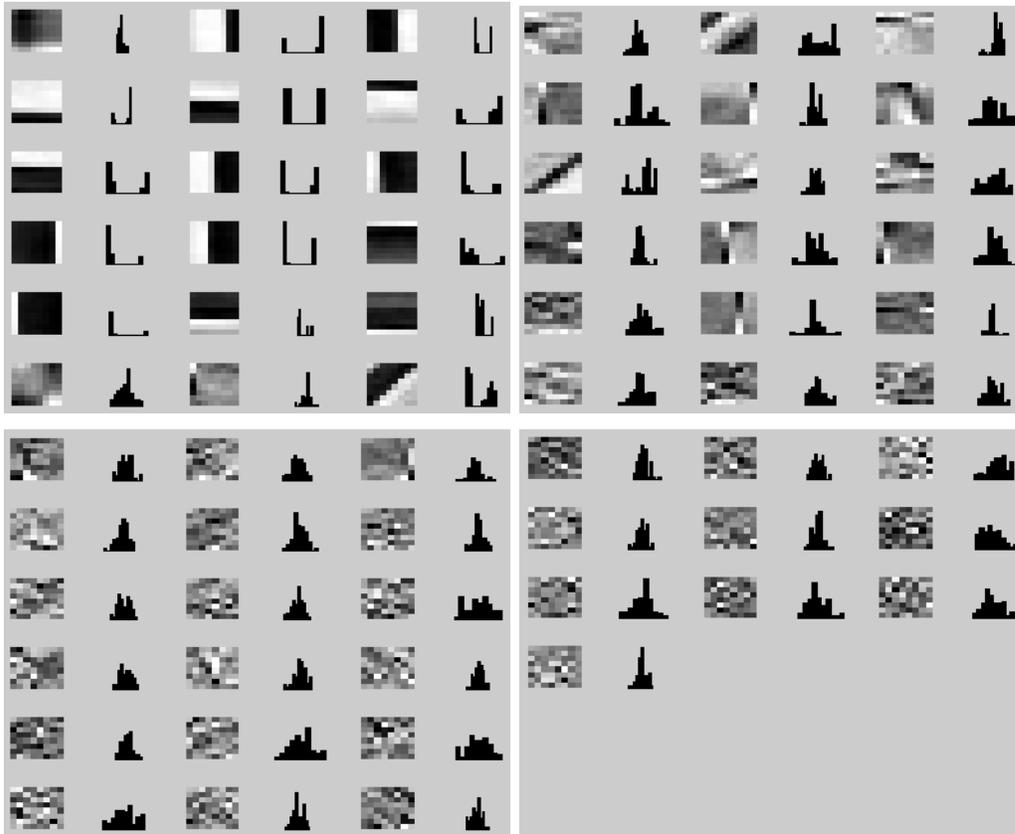


Figura 4.32: Algoritmo JADE. Evaluación de cada columna de \mathbf{A} para datos originales.

Al igual que para el caso anterior, con el algoritmo JADE se realiza la comparación entre las diferentes señales. Para este caso puede verse que se presentan de manera más escasa filtros estables entre la matriz de mezcla \mathbf{A}

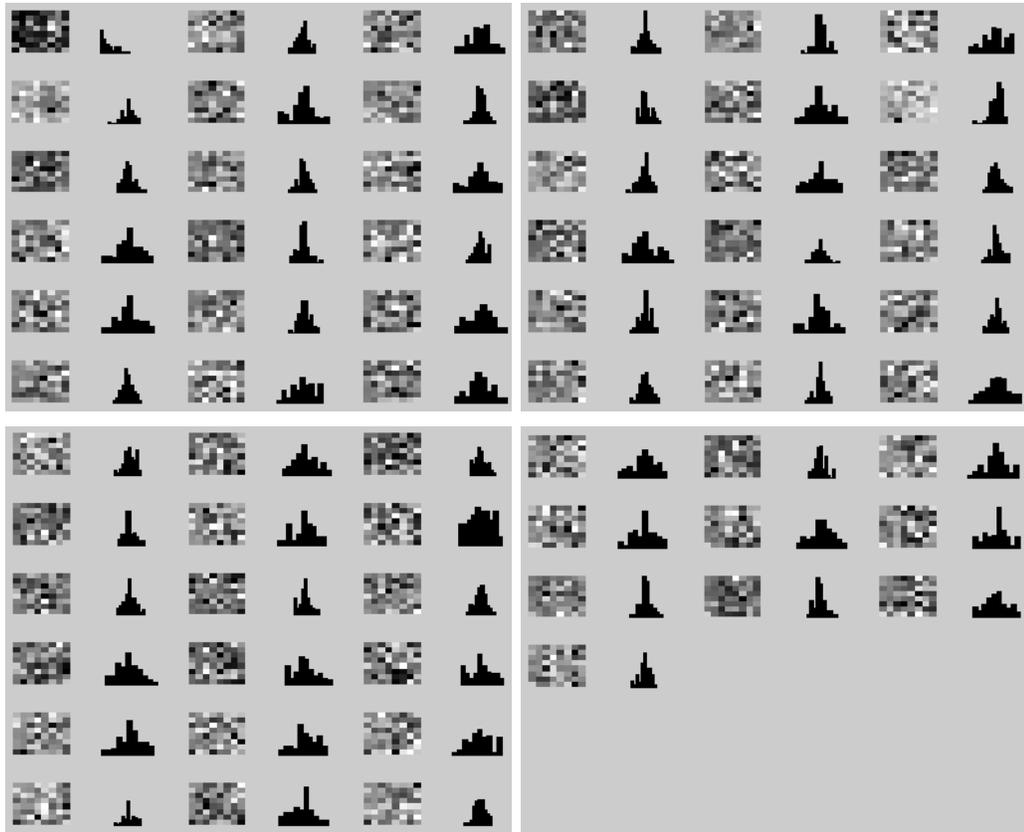


Figura 4.33: Algoritmo JADE. Evaluación de cada columna de \mathbf{A} para datos con ruido gaussiano.

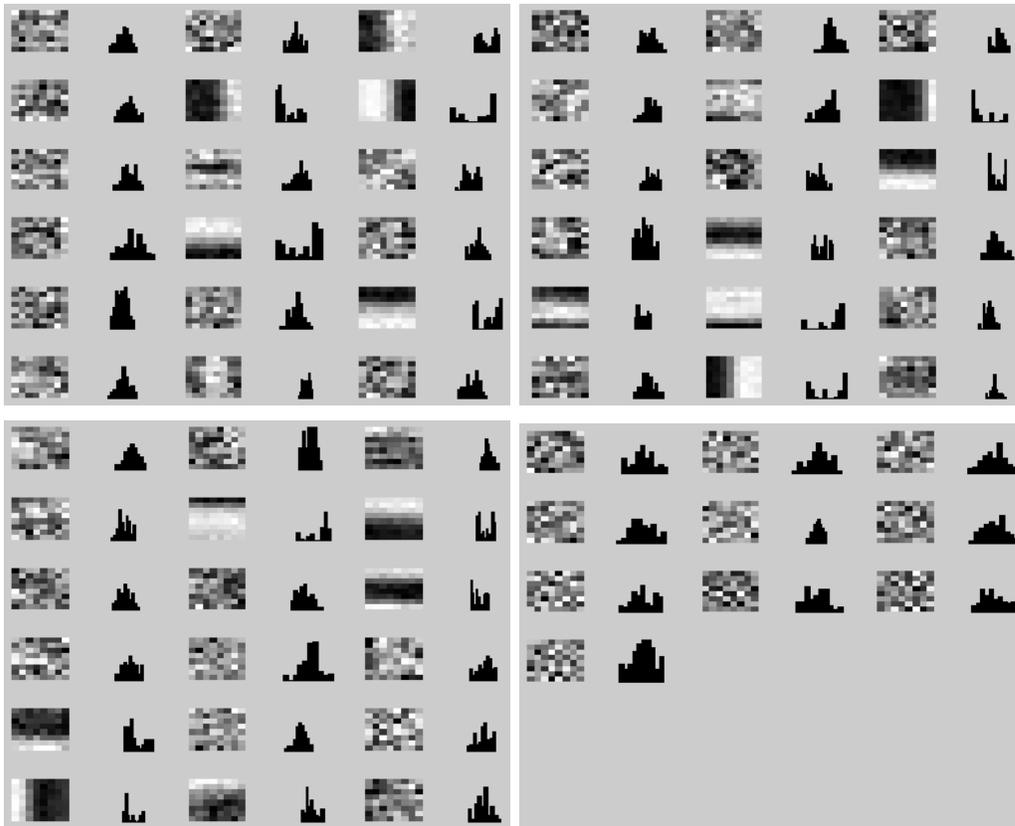


Figura 4.34: Algoritmo JADE. Evaluación de cada columna de \mathbf{A} para datos con ruido multiplicativo.

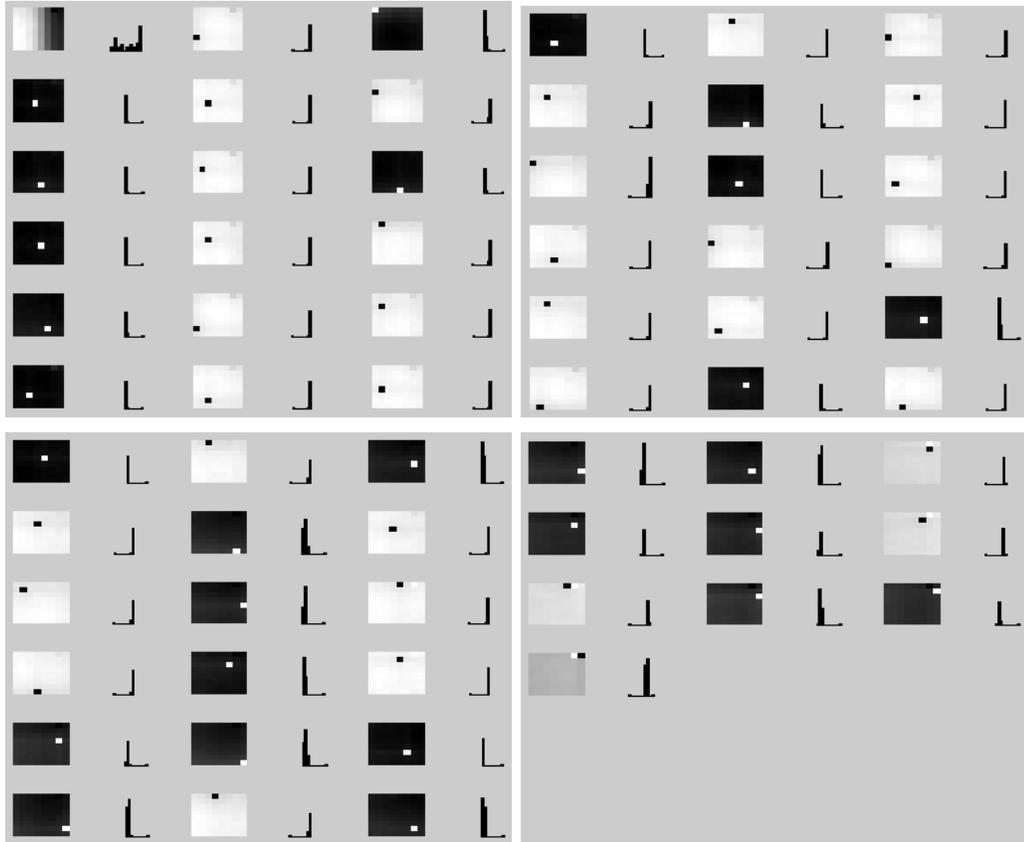


Figura 4.35: Algoritmo JADE. Evaluación de cada columna de \mathbf{A} para datos con ruido de sal y pimienta.

de las señales originales y las matrices de mezcla que contienen los diferentes ruidos. Los valores mínimos de correlación elegidos para este caso son por encima de 0.5. Propiamente para el caso de ruido gaussiano, entre las señales originales y el mismo no existe ninguna componente que sea estable, mientras que para las otras clases de ruido se presentan componentes; pero no en cantidad. Los ocho primeros resultados de la comparación con señales de ruido multiplicativo se presentan como ejemplo en la figura 4.36 y los resultados de la comparación con señales de ruido de sal y pimienta se muestran en la figura 4.37.

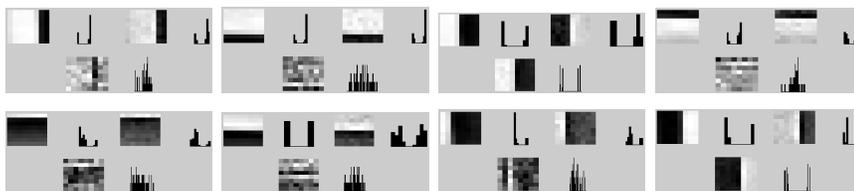


Figura 4.36: Algoritmo JADE. Ocho primeros filtros de componentes entre la señal original y la señal con ruido multiplicativo de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación.

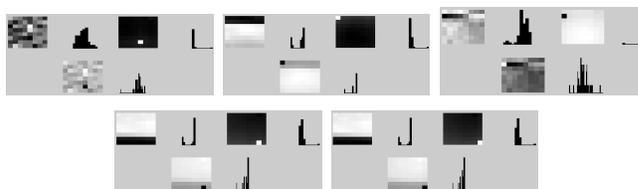


Figura 4.37: Algoritmo JADE. Filtros de componentes entre la señal original y la señal con ruido de sal y pimienta de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación.

Normalmente, elegir valores de correlación altos implican alta dependencia entre las variables. Es interesante observar que incluso con un valor no muy alto de correlación como es 0,5 (incluso), no se presentan componentes estables entre las muestras originales y las muestras con ruido gaussiano.

A continuación, la figura 4.38 se presentan los resultados obtenidos después de realizar la comparación con las muestras de ruido multiplicativo. La parte superior de la figura muestra los resultados de la matriz de mezcla original, mientras que en la parte inferior se muestran los resultados de la matriz con ruido multiplicativo.

En los filtros con ruido multiplicativo reflejados en la figura 4.38, puede verse que estos no contienen ninguna información aparte del ruido y por su característica de distribución representada al lado derecho de cada filtro, por medio del histograma, se puede observar que la clase de ruido dominante en cada uno de los filtros es ruido multiplicativo; aunque a primera vista pareciese que algunas componentes parecen describir distribución gaussiana. Otro punto que resulta de interés en la observación de estos resultados es que las señales originales (parte superior), no pierden la totalidad de los filtros interesantes, esto es, en particular y sobre todo en la primera gráfica, muchos de los resultados describen bordes o esquinas que en el análisis anterior no fueron eliminadas.

Para los resultados obtenidos después de comparar las muestras originales con las muestras de ruido de sal pimienta ocurre algo muy similar que en el caso anterior. Las figuras 4.39 y 4.40, muestra en la parte A los resultados con

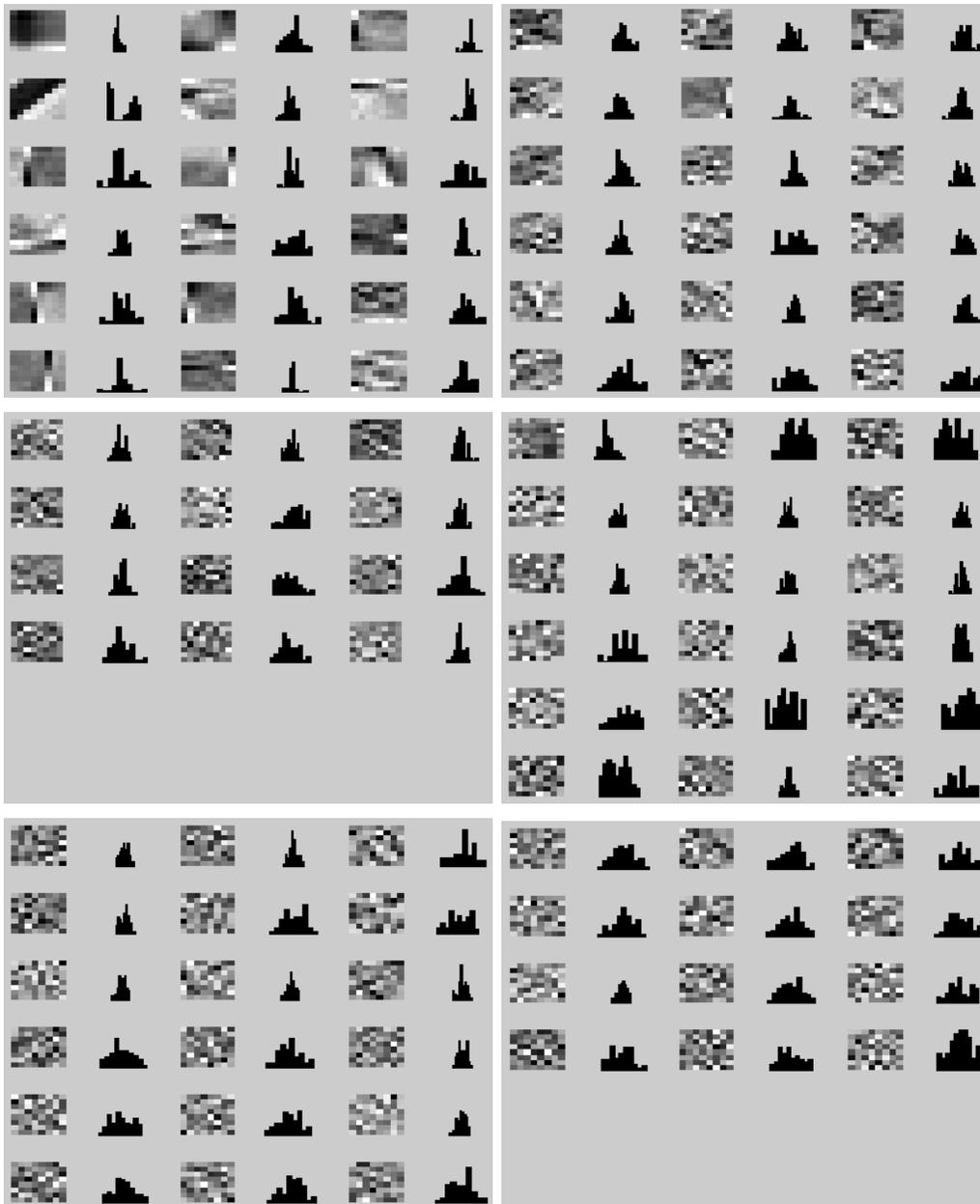


Figura 4.38: Algoritmo JADE. Filtros de componentes entre la señal original (parte superior y centro parte izquierda) y la señal con ruido multiplicativo (centro derecha y parte inferior) de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación.

las muestras originales que al igual que antes, parecen en su primera parte describir datos importantes; mientras que la parte B describe las muestras con ruido de sal y pimienta. Al igual que con el algoritmo de fastICA, el algoritmo JADE también parece tener problemas con el tipo de ruido impulsivo.

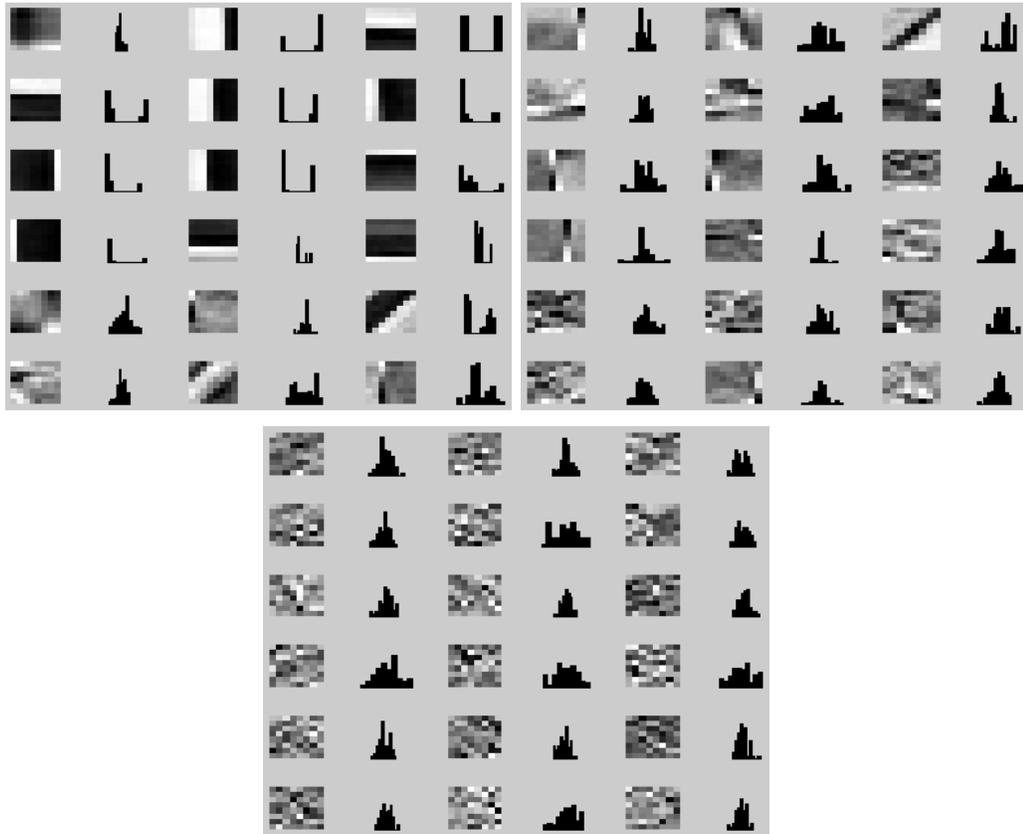


Figura 4.39: Algoritmo JADE. Parte A. Filtros de componentes entre la señal original (parte A) y la señal con ruido de sal y pimienta (parte B) de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación.

Debido a estos resultados de la figura 4.40, se hace interesante nuevamente saber que información es la que hay realmente en estos últimos resultados

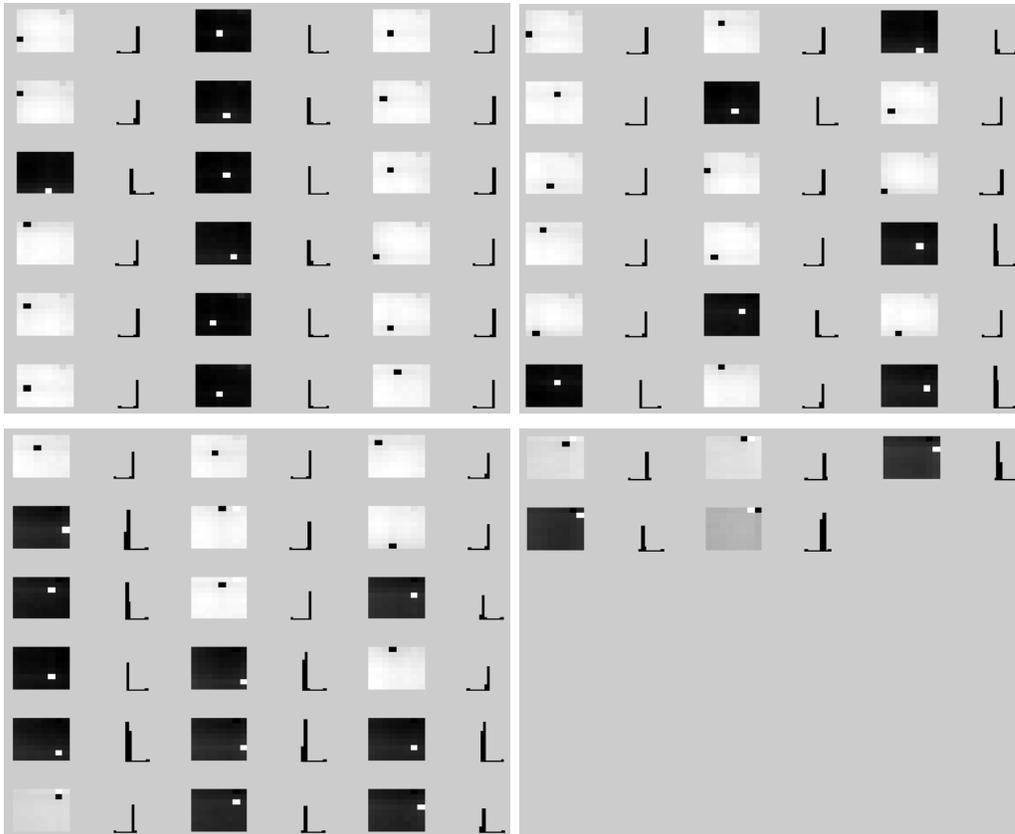


Figura 4.40: Algoritmo JADE. Parte B. Filtros de componentes entre la señal original (parte A) y la señal con ruido de sal y pimienta (parte B) de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación.

y como es de diferente con los primeros presentados 4.9. En la siguiente figura se muestran los resultados obtenidos una vez llevado a cabo el mismo proceso realizado que con el algoritmo de fastICA. En estos resultados puede apreciarse la falta de información lógica que los filtros parecen presentar, de hecho aunque a primera vista parece que no hubiese ninguna relación entre cualquiera de estos resultados, al realizar los respectivos cálculos se ha podido eliminar algunas componentes que tienen correlación.

4.3.3. Evaluación de resultados presentados con AMUSE

A continuación se realiza la evaluación de los filtros de la matriz de mezcla \mathbf{A} , de los resultados entregados por AMUSE. Éstos presentan diferencias bastante apreciables en comparación con los otros dos algoritmos anteriormente mostrados y en especial cuando se trabaja con ruido de sal y pimienta.

En primera instancia y siguiendo el esquema de antes, son presentadas las imágenes de las figuras 4.42, 4.43, 4.44 y 4.45, cada una de ellas representa respectivamente la información de la evaluación del histograma para los filtros de las matrices de mezcla \mathbf{A} de los datos originales, con ruido gaussiano, multiplicativo y de sal y pimienta respectivamente. En esta evaluación resulta interesante resaltar que para cada clase de ruido no es tan evidente poder especificar de manera inmediata (como ocurre con los otros algoritmos) que tipo de ruido se encuentra en cada filtro, esto es porque pueden verse

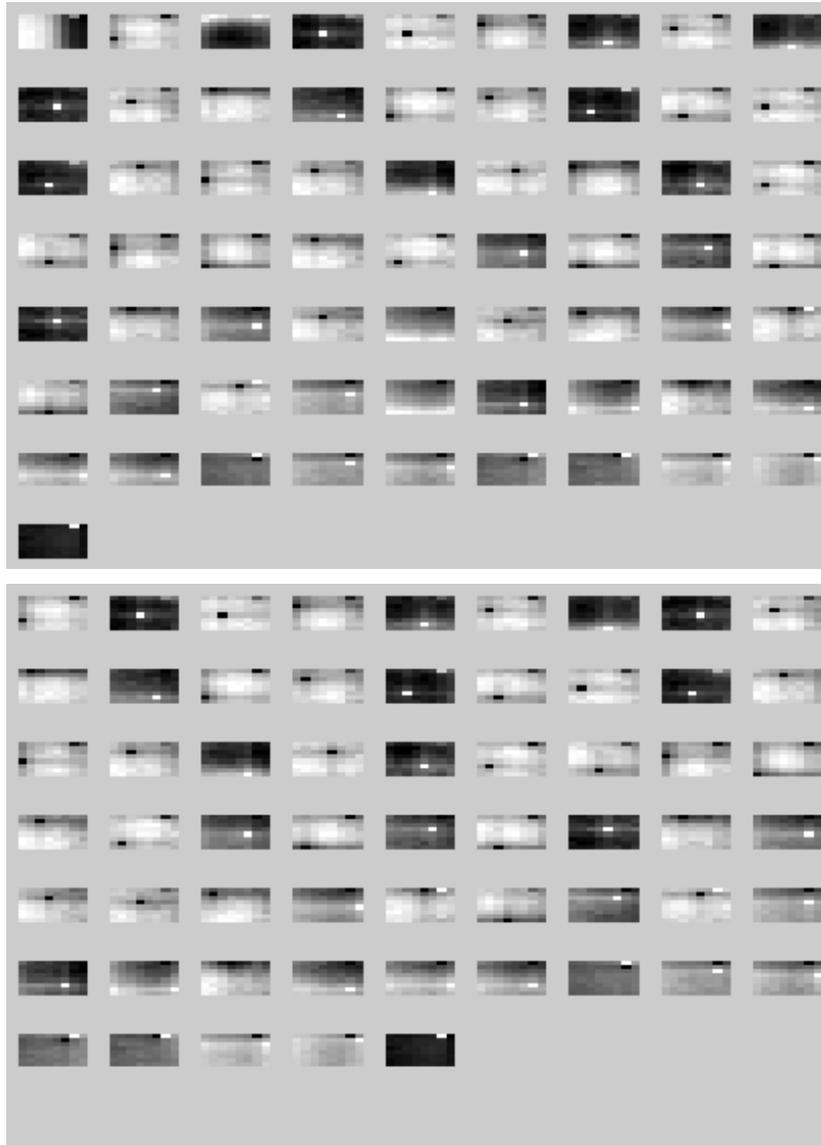


Figura 4.41: Algoritmo JADE. Comparación de los diferentes filtros resultantes llevando a 0 los puntos con valor más alto con el fin de buscar la información contenida. En la parte superior se presenta el resultado con todos los primeros filtros encontrados una vez llevado a cabo JADE. En la parte inferior se presentan todos los filtros resultantes después de llevada a cabo la correlación.

características de la distribución con ruido gaussiano en los resultados representados con ruido multiplicativo y viceversa. Igualmente la distribución con ruido de sal y pimienta pierde por completo la propiedad que desde el principio venía conservando con los otros algoritmos, presentando distribuciones similares a las otras clases de ruido.

Otro punto interesante que muestran los resultados de este algoritmo es la continuidad en los filtros que presentan alta correlación, esto es, a primera vista, una vez obtenida la información que representa cada uno de los resultados de los filtros (originales, ruido gaussiano, ruido multiplicativo y ruido de sal y pimienta), puede verse que son los mismos 8 primeros filtros y el último para cada caso, los que muestran continuidad y describen información importante como bordes.

La comparación de los filtros estables de las muestras sin ruido con las muestras contaminadas obtenidas con el algoritmo AMUSE se presentan en las figuras 4.46 para señal original comparada con ruido gaussiano, 4.47 para señal original comparada con ruido multiplicativo y 4.48 para señal original comparada con ruido de sal y pimienta. De la observación que se hace de los filtros con alta correlación es interesante mencionar la alta similitud que poseen estos datos al ser analizados una vez hecha las comparaciones. De estas imágenes anteriormente mencionadas, puede apreciarse como por ejemplo, la sub-imágenes encerradas en un rectángulo rojo. Aquí puede verse que sin importar la clase de ruido predominante, la información entregada para los tres casos de comparación es salvo algunas pequeñas diferencias, prácticamente



Figura 4.42: Algoritmo AMUSE. Evaluación de cada columna de \mathbf{A} para datos originales.

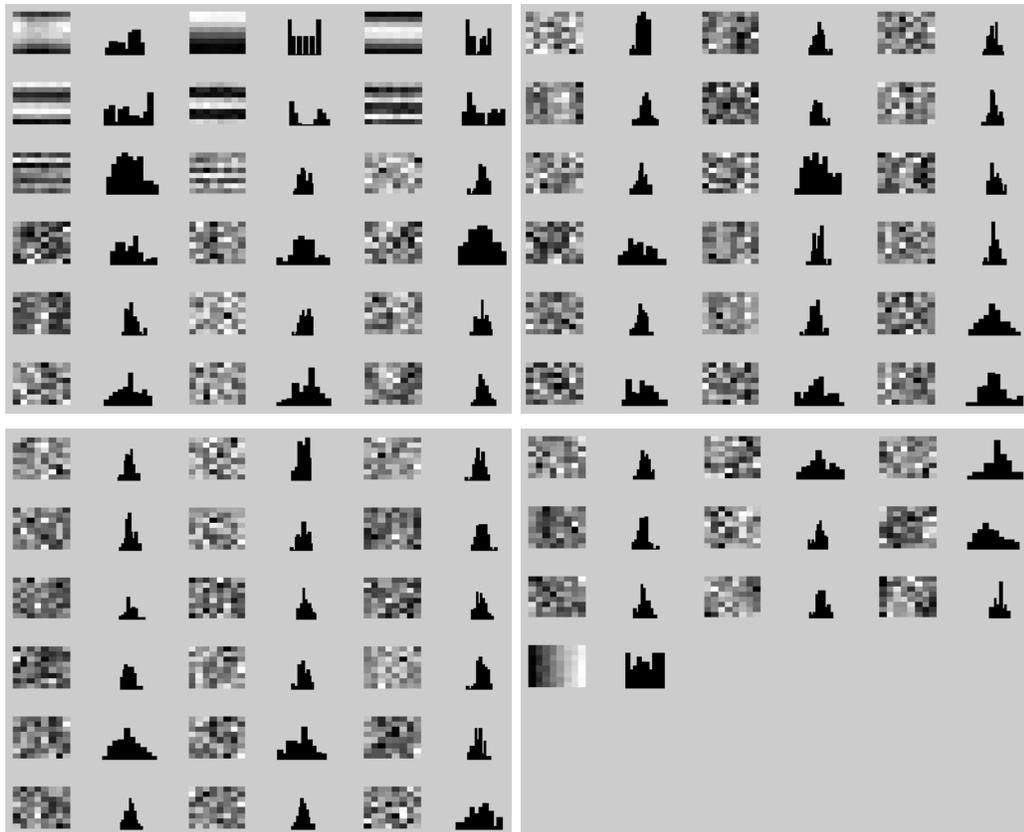


Figura 4.43: Algoritmo AMUSE. Evaluación de cada columna de A para datos con ruido gaussiano. Puede observarse como muchos de los filtros pierden las características típicas de una distribución gaussiana, dificultando más, la rápida evaluación.

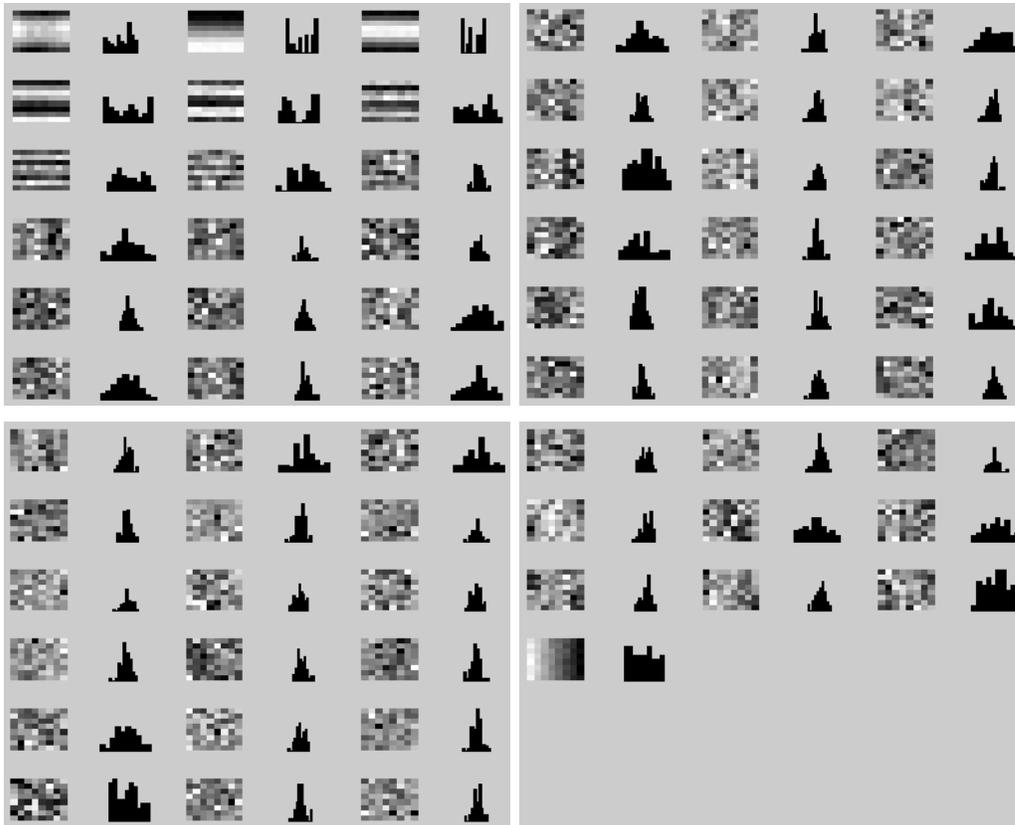


Figura 4.44: Algoritmo AMUSE. Evaluación de cada columna de \mathbf{A} para datos con ruido multiplicativo. Se observa la prevalencia de la mayoría de los filtros con tendencia a comportarse como una distribución gaussiana, perdiendo las propiedades multiplicativas.

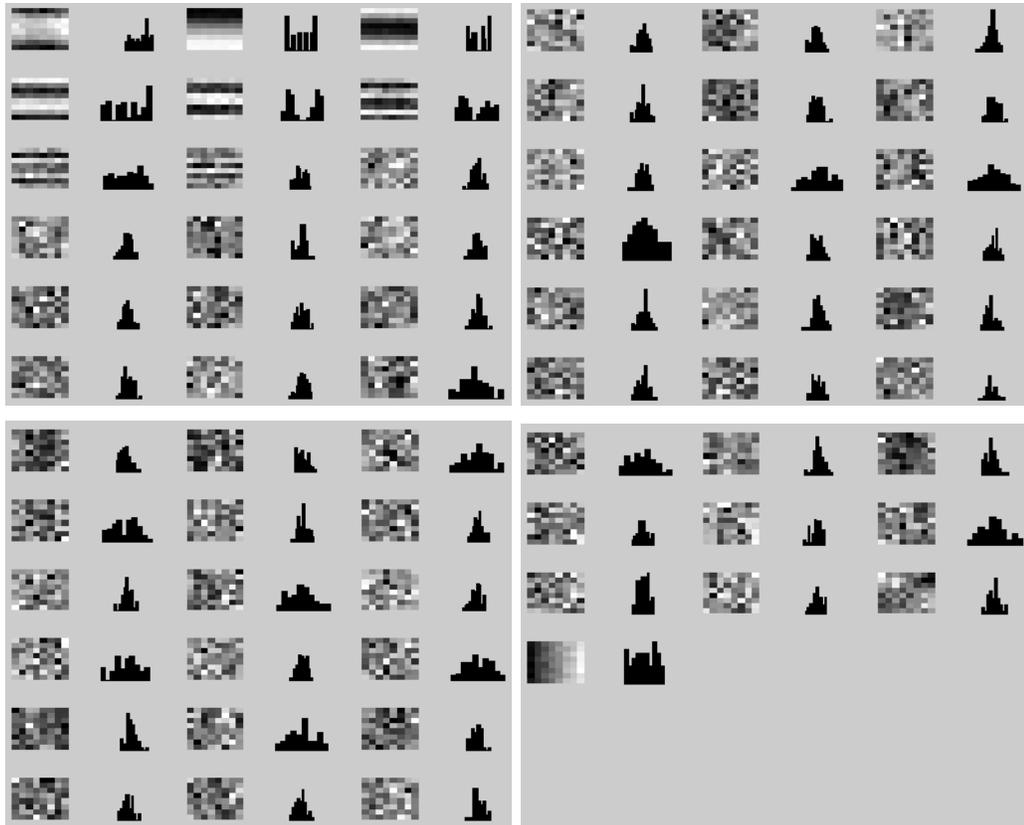


Figura 4.45: Algoritmo AMUSE. Evaluación de cada columna de \mathbf{A} para datos con ruido de sal y pimienta. Este resultado es interesante ya que los filtros para este caso, no se comportan como en casos anteriores, sino que parecieran describir otro tipo de característica de ruido.

la misma. Como puede apreciarse lo mismo ocurre con otros filtros que puede que no guarden la mismas posición; pero si las características.

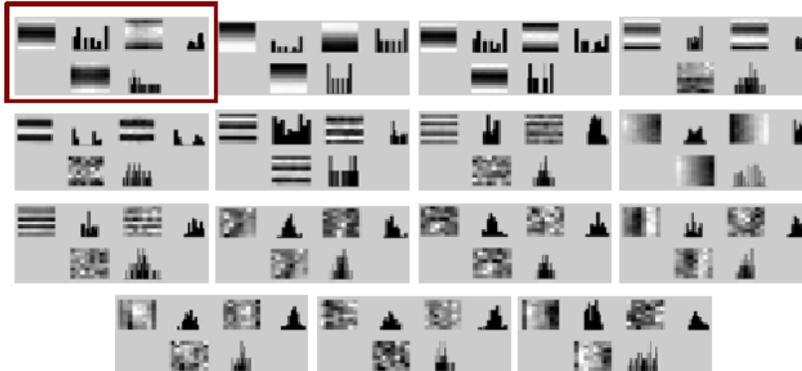


Figura 4.46: Algoritmo AMUSE. Comparación entre los filtros de componentes entre la señal original y la señal con ruido gaussiano de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación.

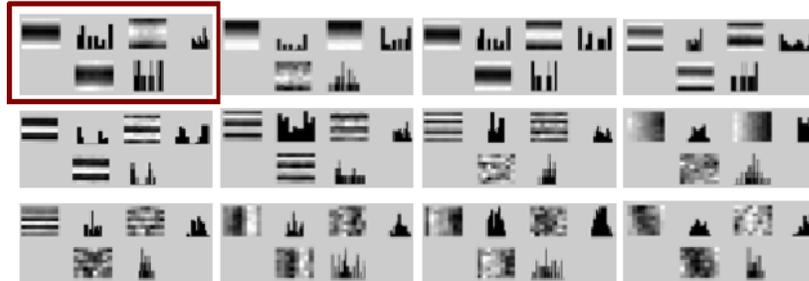


Figura 4.47: Algoritmo AMUSE. Comparación entre los filtros de componentes entre la señal original y la señal con ruido multiplicativo de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación.

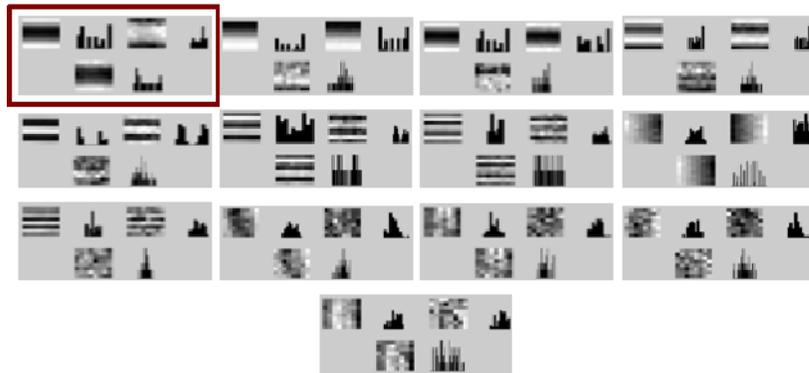


Figura 4.48: Algoritmo AMUSE. Comparación entre los filtros de componentes entre la señal original y la señal con ruido de sal y pimienta de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación.

Por último se realiza la eliminación de los filtros estables para obtener los filtros de información contaminados con sólo ruido. Las figuras 4.49 y 4.50, muestran la información de los datos obtenidos de la comparación entre la señal original y la señal con ruido gaussiano; las figuras 4.51 y 4.52, muestra la información sólo ruido de los filtros resultantes después de comparar las muestras con la señal de ruido multiplicativo y 4.53 y 4.54, representan los datos de la comparación de la señal original con la señal contaminada con ruido de sal y pimienta. Debido a la forma en que se han presentado estos datos puede verse como para los tres casos, los filtros originales representados en la parte A de cada una de las figuras, representan características especiales que no han sido eliminadas en la comparación; mientras que para la señal que contiene únicamente ruido (parte B), se muestran sólo datos con ruido donde las características de la forma de distribución de cada filtro como se dijo con anterioridad, no permiten una clara identificación del tipo de ruido contenido. También cabe resaltar como en la figura 4.52, se sigue presentando en la posición del primer filtro representado, una imagen que representa algún tipo de borde.

Otro dato de interés observado en la evaluación del algoritmo AMUSE es que la información eliminada, sin importar las características de la clase de ruido, son en su mayoría los mismos en cada uno de los casos; por ejemplo, en los filtros representados por las columnas 1, 2, 4, 5, 6, 7, 8, presentan el mismo comportamiento para cada tipo de característica de ruido y con ello son en cada uno de los procesos eliminadas de manera independiente.



Figura 4.49: Algoritmo AMUSE. Parte A. Resultado de ruido obtenido después de eliminar las componentes estables entre los datos originales (parte A) y la señal con ruido gaussiano (parte B).

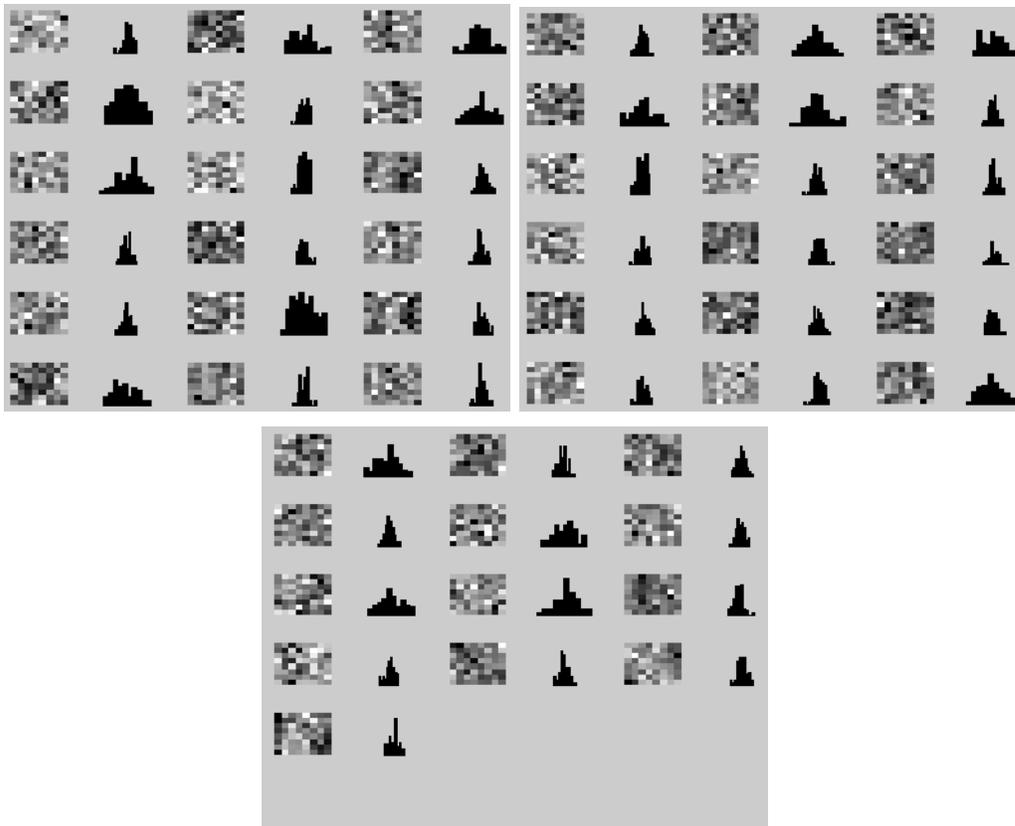


Figura 4.50: Algoritmo AMUSE. Parte B. Resultado de ruido obtenido después de eliminar las componentes estables entre los datos originales (parte A) y la señal con ruido gaussiano (parte B).



Figura 4.51: Algoritmo AMUSE. Parte A. Resultado de ruido obtenido después de eliminar las componentes estables entre los datos originales (parte A) y la señal con ruido multiplicativo (parte B).

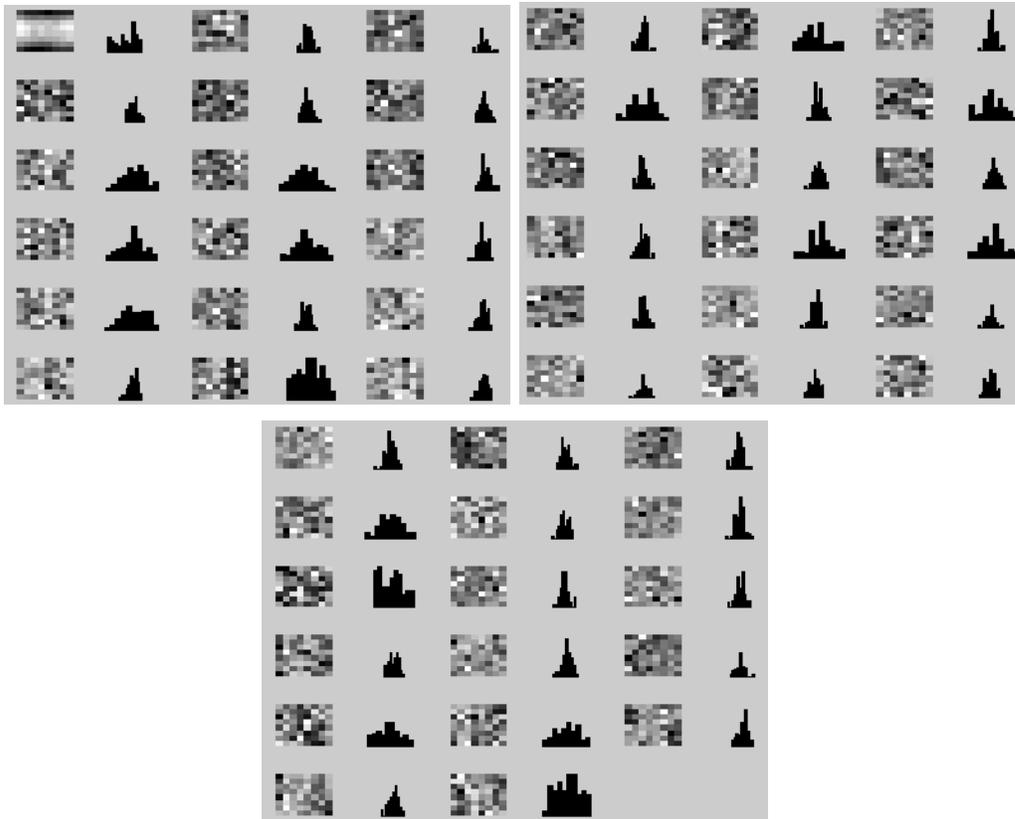


Figura 4.52: Algoritmo AMUSE. Parte B. Resultado de ruido obtenido después de eliminar las componentes estables entre los datos originales (parte A) y la señal con ruido multiplicativo (parte B).

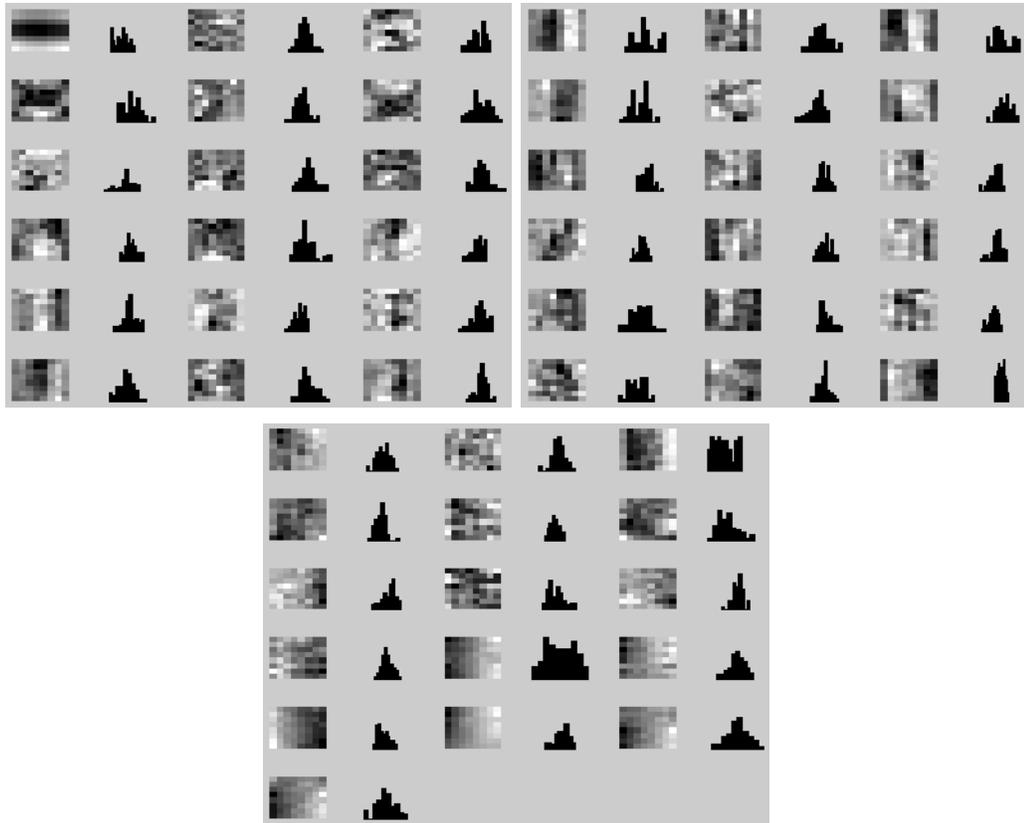


Figura 4.53: Algoritmo AMUSE. Parte A. Resultado de ruido obtenido después de eliminar las componentes estables entre los datos originales (parte A) y la señal con ruido de sal y pimienta (parte B).

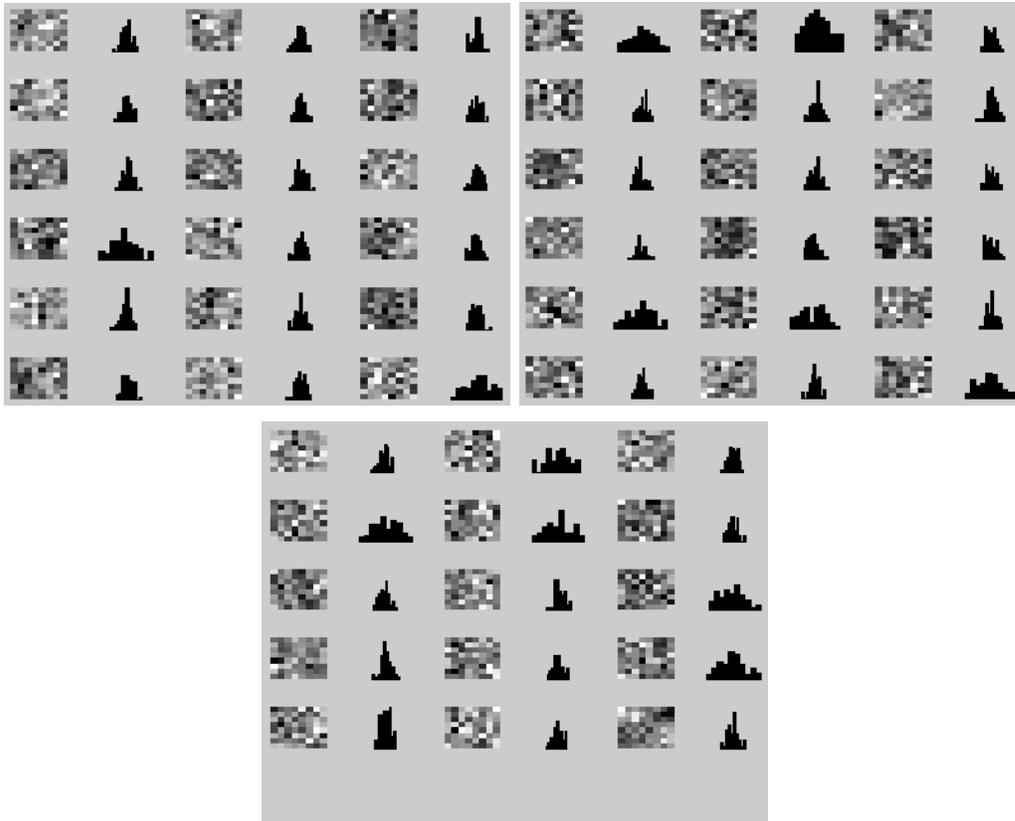


Figura 4.54: Algoritmo AMUSE. Parte B. Resultado de ruido obtenido después de eliminar las componentes estables entre los datos originales (parte A) y la señal con ruido de sal y pimienta (parte B).

4.3.4. Evaluación de resultados presentados con fastGEO

La evaluación de los resultados entregados por el algoritmo fastGEO se presentan en las figuras 4.55 para la muestra sin ruido, 4.56 para muestra con ruido gaussiano, 4.57 para muestra con ruido multiplicativo y 4.58 para muestra con ruido de sal y pimienta.

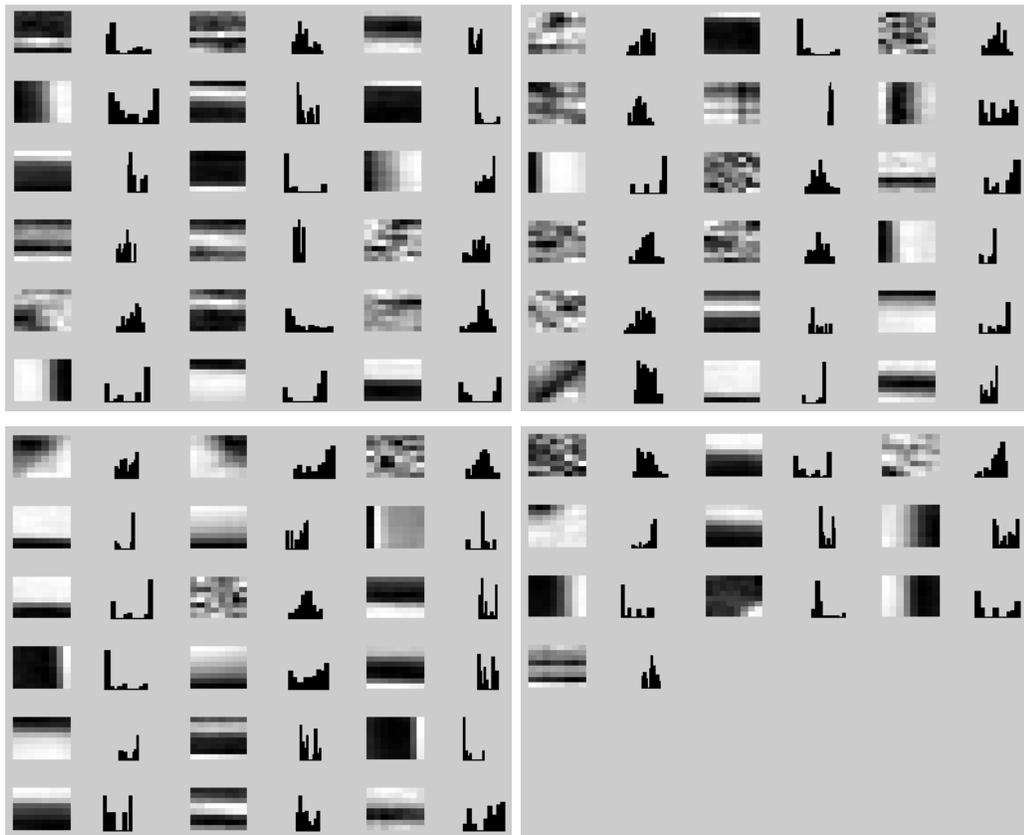


Figura 4.55: Algoritmo fastGEO. Evaluación de cada columna de \mathbf{A} para datos originales. Puede apreciarse una distribución clara en los filtros de información que contienen bordes o esquinas de interés.



Figura 4.56: Algoritmo fastGEO. Evaluación de cada columna de \mathbf{A} para datos con ruido gaussiano. Se observan filtros en su mayoría contaminados, donde la evaluación de su distribución se comporta de forma gaussiana.

La distribución de los datos presentados en la figura 4.55 representa información de filtros con características que parecen contener información importante de las muestras originales, esto es un resultado esperado ya que en sí, la información original sólo es una mezcla de sí misma. Para la figura 4.56 en general las componentes de ruido describen las representaciones típicas de ruido gaussiano, que permite una identificación rápida del caso en con-

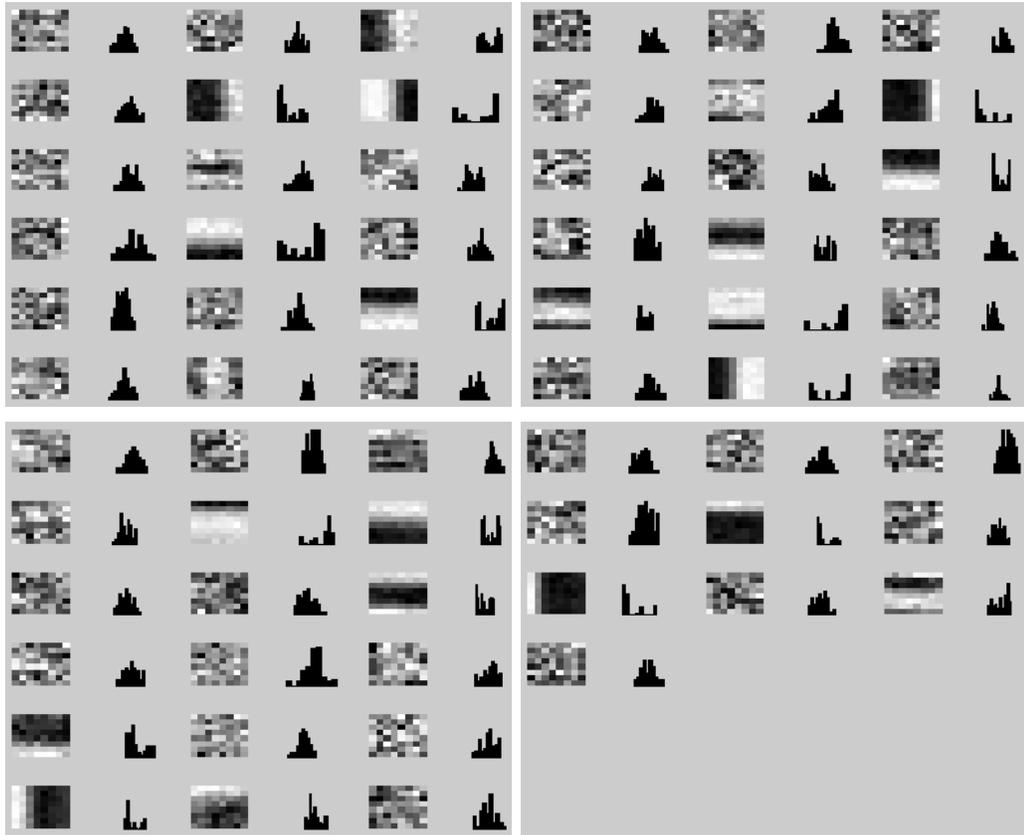


Figura 4.57: Algoritmo fastGEO. Evaluación de cada columna de \mathbf{A} para datos con ruido multiplicativo. La mayoría de los filtros contaminados no presentan la distribución típica del ruido multiplicativo hasta ahora trabajado. En muchos casos, estos resultados podrían ser confundidos con la forma de distribución del ruido gaussiano.

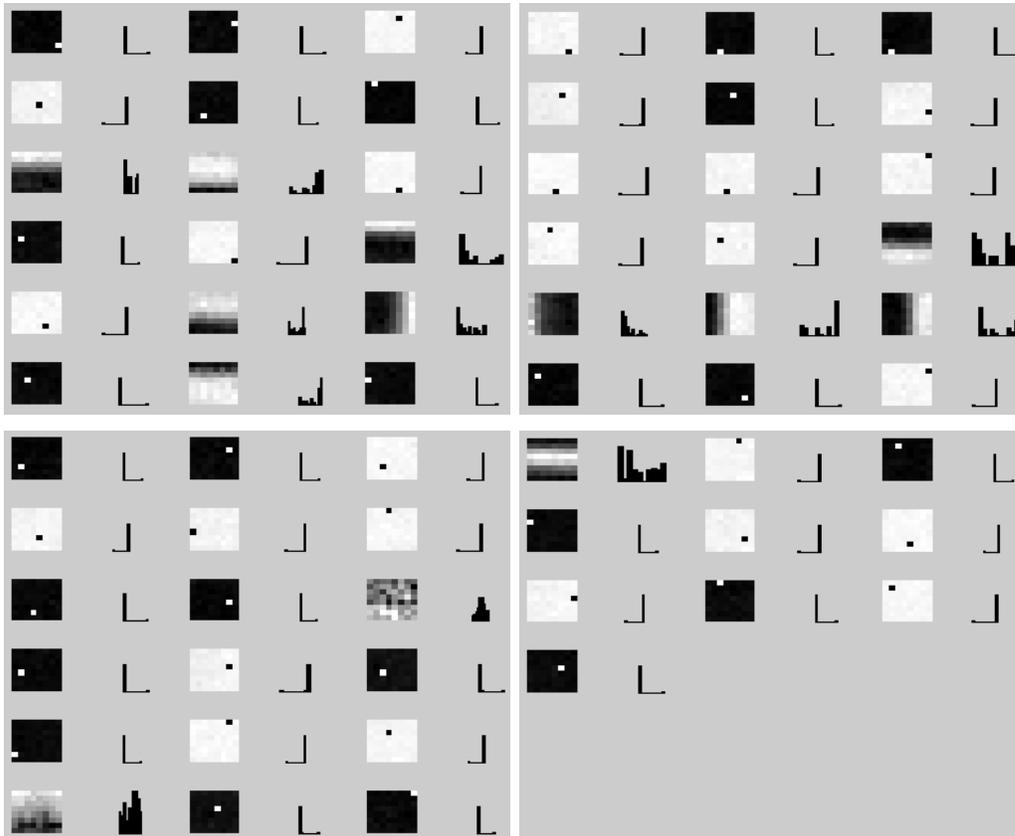


Figura 4.58: Algoritmo fastGEO. Evaluación de cada columna de \mathbf{A} para datos con ruido de sal y pimienta. Se retoman los resultados presentados por los algoritmos de fastICA y JADE; pero puede observarse también la presencia más numerosa de algunos filtros representando bordes.

creto, ocurriendo lo mismo para el ruido de sal y pimienta 4.58; el algoritmo fastGEO presenta en esta parte las mismas características obtenidas con los algoritmos de fastICA y JADE; pero a la hora de evaluar los datos obtenidos con el ruido multiplicativo se puede apreciar en la figura 4.57, que muchas de las componentes describen características de distribución muy similares a la del ruido gaussiano, lo que podría complicar una inmediata catalogación del problema.

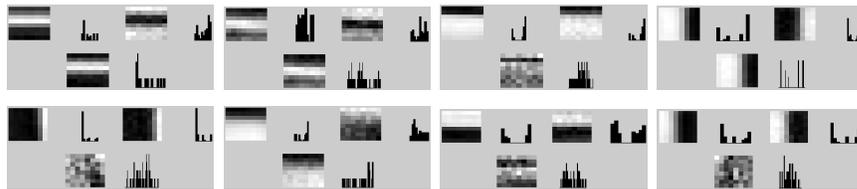


Figura 4.59: Algoritmo fastGEO. Primeros ocho filtros de componentes entre la señal original y la señal de ruido gaussiano de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación.

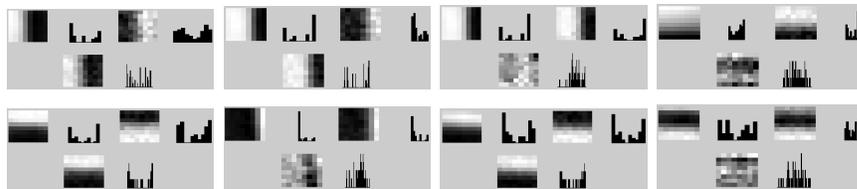


Figura 4.60: Algoritmo fastGEO. Primeros ocho filtros de componentes entre la señal original y la señal de ruido multiplicativo de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación.

La comparación de los ocho primeros filtros de la señal original con las señales de ruido gaussiano, multiplicativo y de sal y pimienta se presenta en

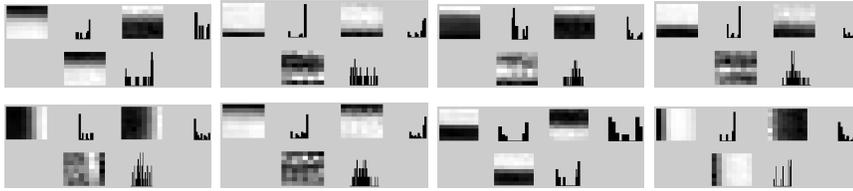


Figura 4.61: Algoritmo fastGEO. Primeros ocho filtros de componentes entre la señal original y la señal de ruido de sal y pimienta de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación.

la figura 4.59, 4.60 y 4.61 respectivamente. De la misma forma que los otros casos para la elección de estas componentes en un principio se selecciono valores de correlación por encima de 0,9 con lo que se vio que se conservaban componentes con características importantes las cuales era necesario eliminar de los resultados finales. Por ello en un principio se bajo el nivel de correlación a 0,5 esperando obtener en su mayoría componentes que representasen ruido.

Una vez extraídos los filtros de interés, se pasa a desarrollar las comparaciones entre la información original y la información con ruido. Es interesante notar que para este caso, los resultados obtenidos representados en las figuras 4.62, para la comparación entre señal original y señal con ruido gaussiano, las figuras 4.63 y 4.64 que comparan la señal original y la señal contaminada con ruido multiplicativo y las figuras 4.65 y 4.66, que comparan la señal original con ruido de sal y pimienta, la información que están representando cada imagen en la parte B, son filtros que contiene sólo ruido, en donde para la mayoría de los casos puede resultar rápida y fácil la catalogación del mismo; pero para los resultados representados en la parte A de las imágenes,

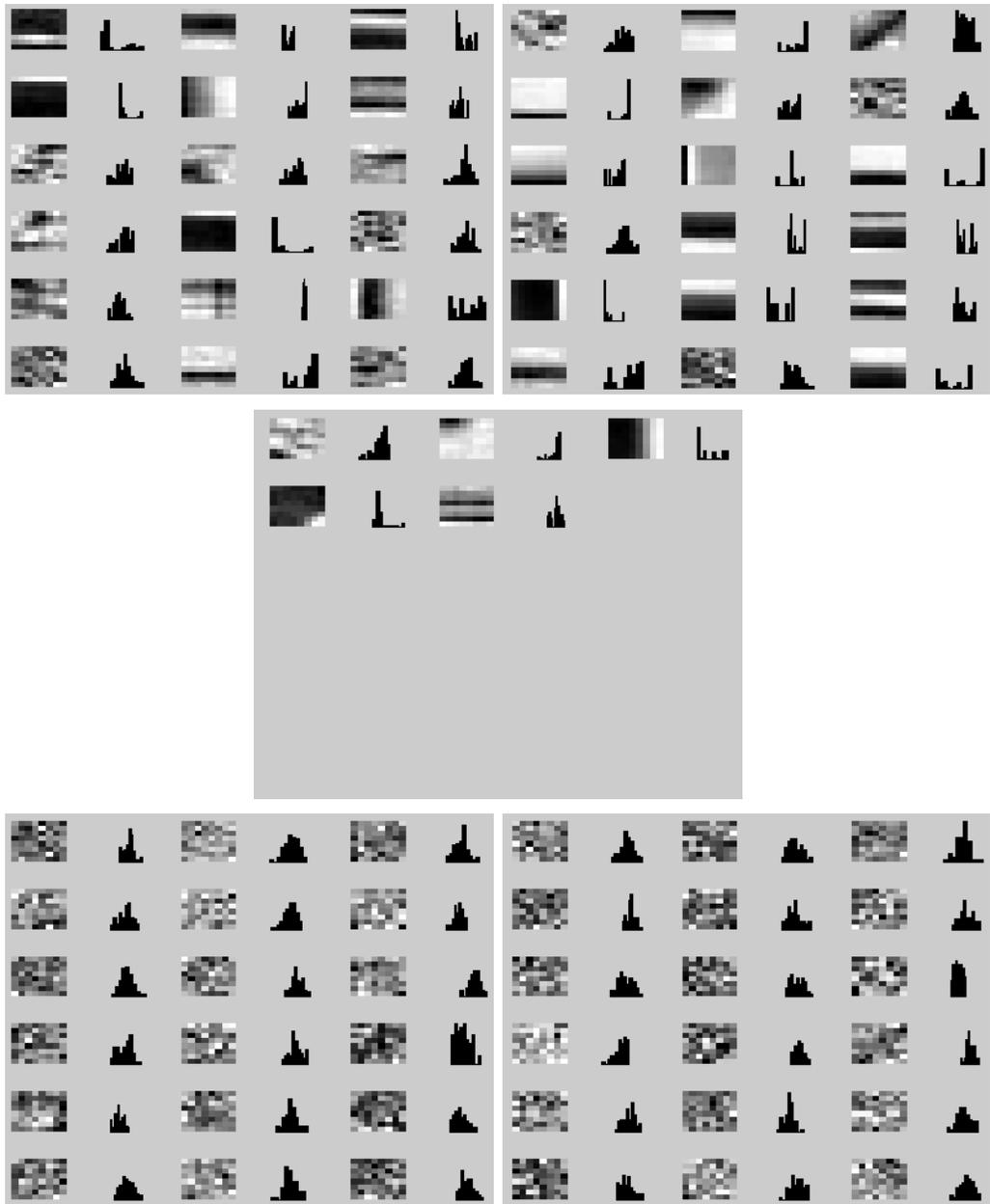


Figura 4.62: Algoritmo fastGEO. Resultado después de comparar las muestras originales (parte superior y centro), con las muestras de ruido gaussiano (parte inferior).

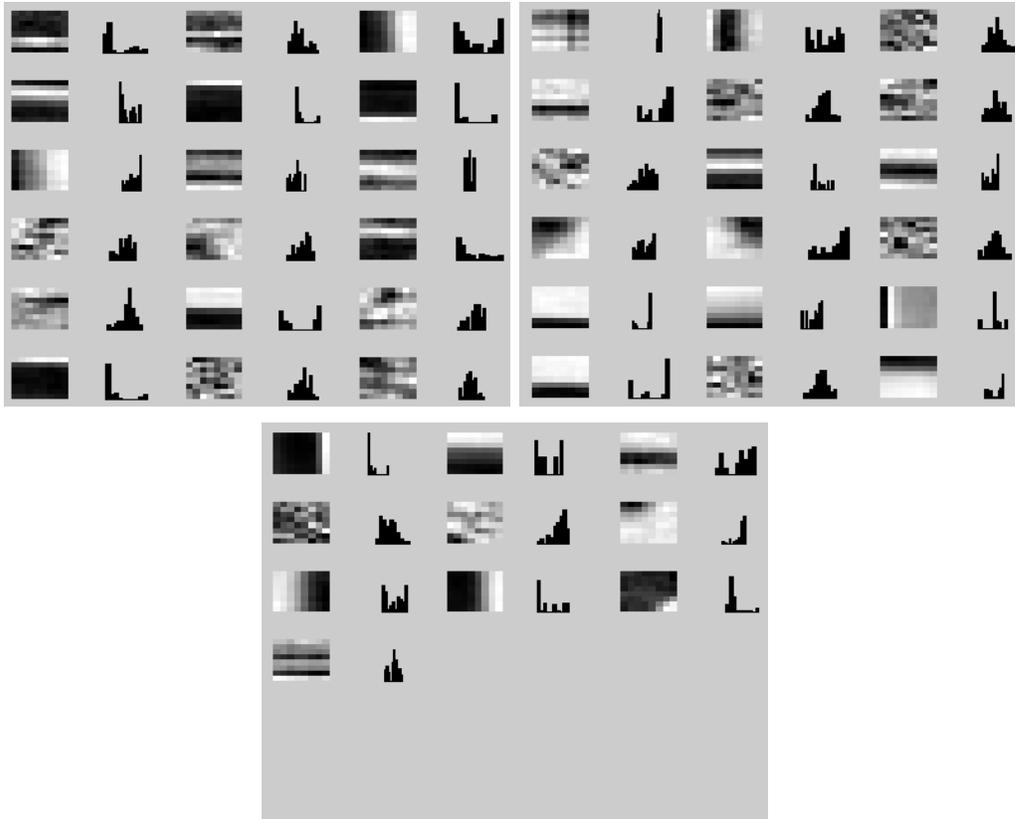


Figura 4.63: Algoritmo fastGEO. Parte A. Resultado después de comparar las muestras originales (parte A), con las muestras de ruido multiplicativo (parte B).

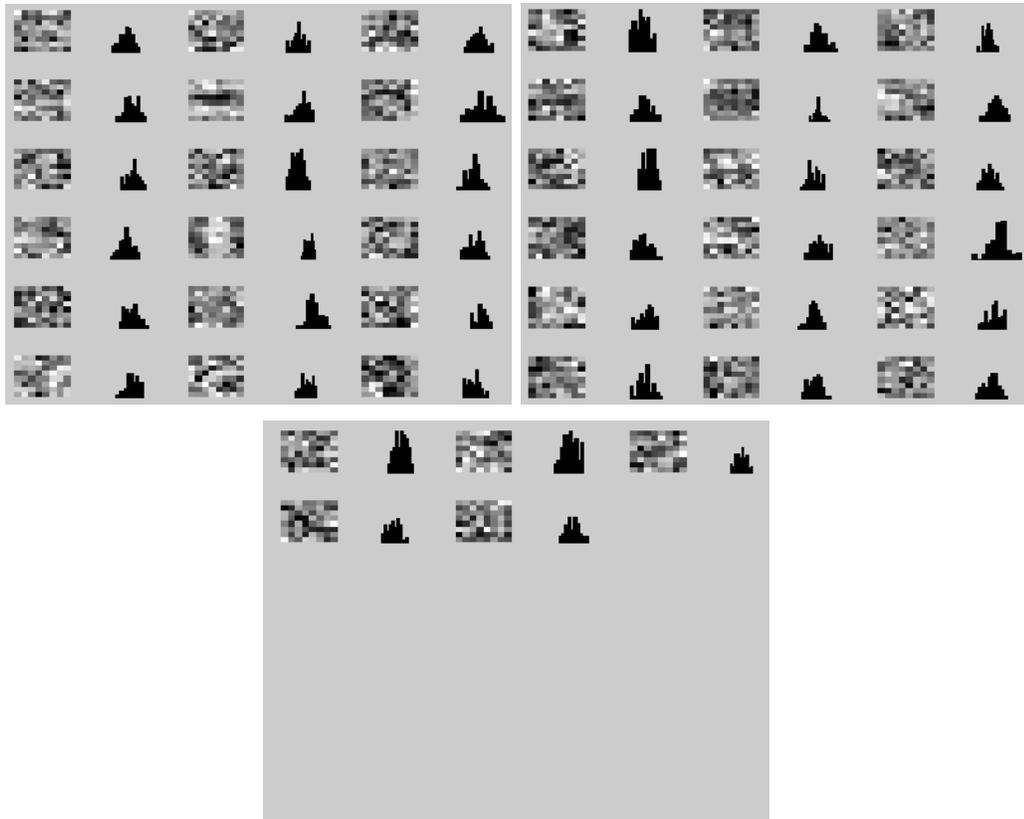


Figura 4.64: Algoritmo fastGEO. Parte B. Resultado después de comparar las muestras originales (parte A), con las muestras de ruido multiplicativo (parte B).

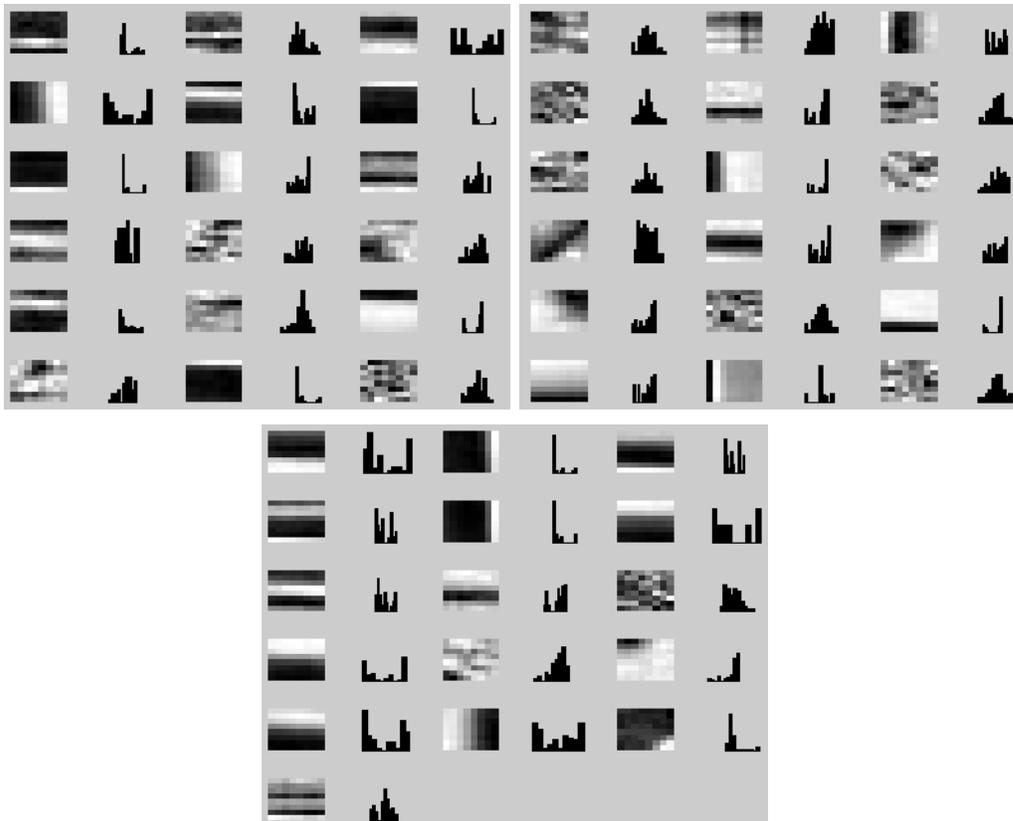


Figura 4.65: Algoritmo fastGEO. Parte A. Resultado después de comparar las muestras originales (parte A), con las muestras de ruido de sal y pimienta (parte B).

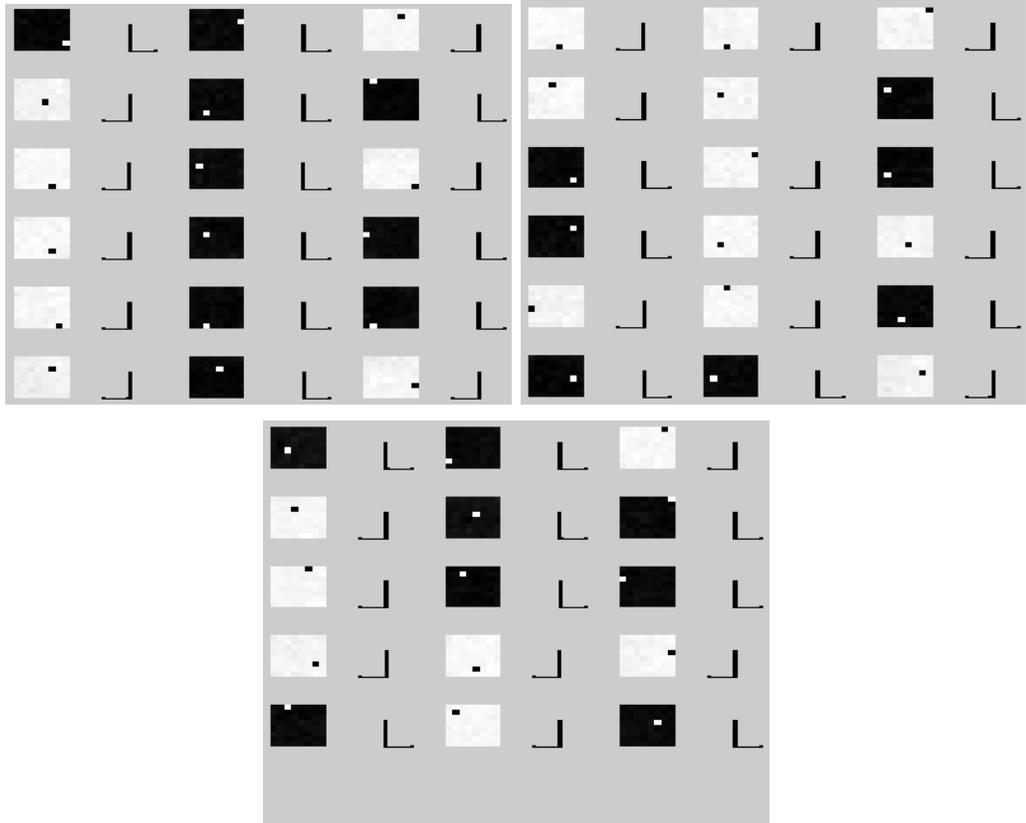


Figura 4.66: Algoritmo fastGEO. Parte B. Resultado después de comparar las muestras originales (parte A), con las muestras de ruido de sal y pimienta (parte B).

con las muestras sin ruido pasa lo contrario, teniendo en su mayoría, filtros que a primera vista se podría pensar que deberían haber sido eliminadas en el proceso anterior de búsqueda de alta correlación.

Ahora bien, como se menciona con anterior este algoritmo presenta las mismas características que los algoritmos de fastICA y JADE al trabajar con ruido impulsivo. Por lo que para este caso buscamos al igual que antes llevar los puntos máximos de cada columna de la matriz de mezcla \mathbf{A} a 0 para poder reconocer la información contenida detrás de la figura 4.66. La figura 4.67, muestra a la derecha los datos obtenidos de la figura 4.58, donde se ha llevado a 0 puntos altos de cada columna de la matriz de mezcla \mathbf{A} con el fin de observar la información hay debajo. A la figura de la parte derecha se le ha hecho el mismo proceso pero una vez eliminados los puntos altos de correlación en el cual sólo queda información que representa ruido.

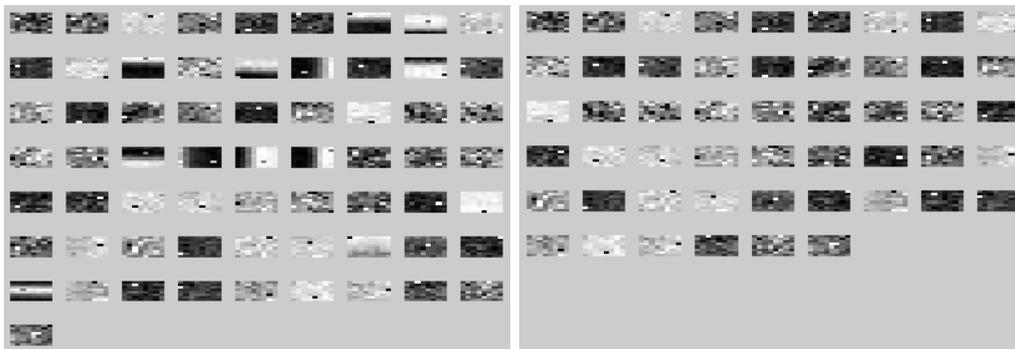


Figura 4.67: Algoritmo fastGEO. Comparación de las diferentes componentes resultantes llevando a 0 los puntos mas altos con el fin de buscar que hay debajo. A la derecha todas las componentes sin realizar la correlación. A la izquierda todas las componentes una vez llevada a cabo la correlación.

En esta figura puede verse que el resultado obtenido al final del proceso de eliminación de componentes esta conforme con los resultados esperados ya que ha eliminado las componentes importantes obteniendo al final filtros representados únicamente por ruido.

4.3.5. Evaluación de resultados presentados con `dss_fastica`

Los resultados obtenidos con `dss_fastica` se presentan a continuación en 4.68, para señales originales, 4.69 para señales con ruido gaussiano, 4.70 para señales con ruido multiplicativo, 4.71 para señales con ruido de sal y pimienta.

El algoritmo de `dss_fastica` es el último de los algoritmos seleccionados para trabajar el análisis comparativo que tiene como fin este trabajo. La forma en como se presentan estos resultados en este punto se debe a que DSS por medio de la interfaz de `dss_fastICA`, desarrolla una PCA más un filtrado. Es por esta razón que los filtros de la matriz de mezcla \mathbf{A} parecen filtros de PCA. Las figuras 4.68, 4.69, 4.70 y 4.71 muestran el primer análisis realizado del comportamiento de cada filtro resultante de las columnas de la matriz de mezcla \mathbf{A} llevado a cabo por medio del histograma. Propiamente en la figura 4.69, que representa los resultados obtenidos con ruido gaussiano se observa la falta de filtros con ruido ya que todas las columnas parecen estar representando información importante de la muestra original; este caso también se presenta para el caso del análisis de ruido multiplicativo en el cual se puede observar solamente el 3 filtros de ruido. Igualmente las distribuciones de los histogramas de cada filtro de la matriz tienden a comportarse como

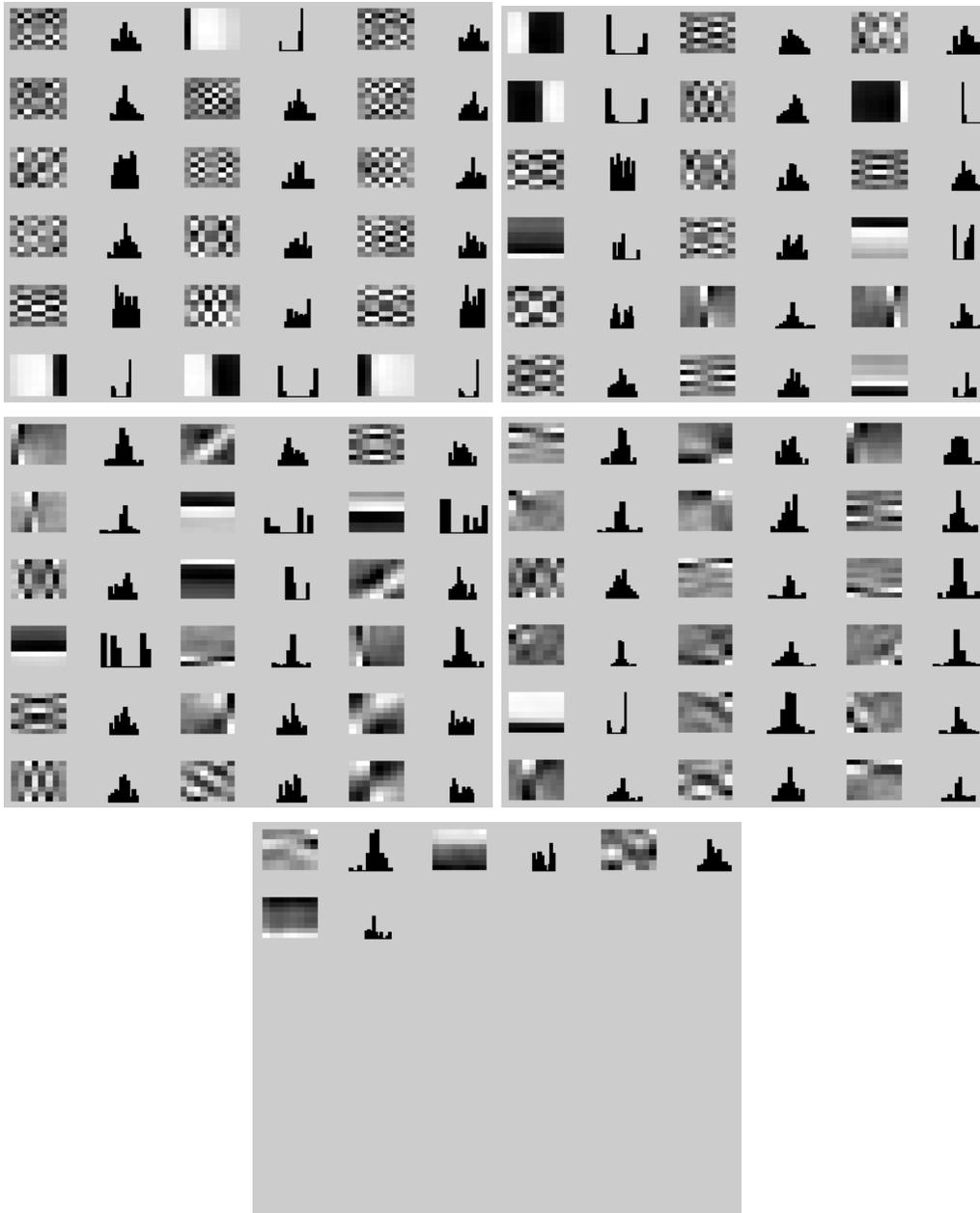


Figura 4.68: `dss_fastICA`. Evaluación de cada columna de A para datos originales.

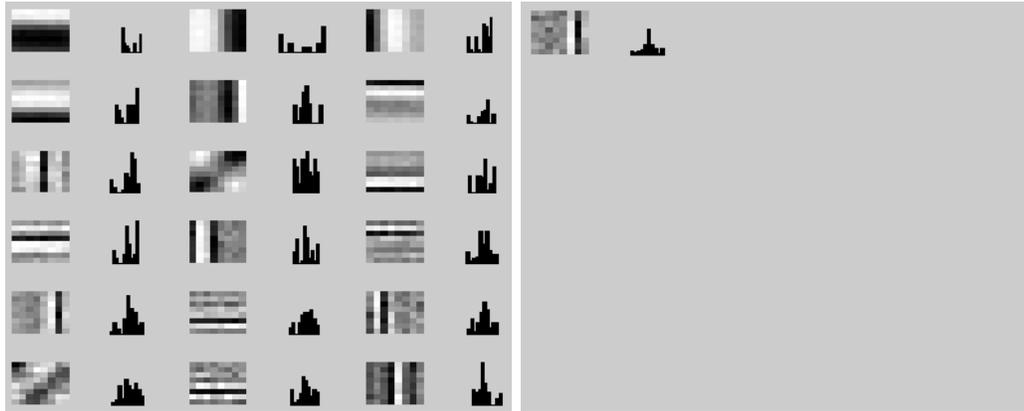


Figura 4.69: `dss_fastICA`. Evaluación de cada columna de \mathbf{A} para datos con ruido gaussiano.

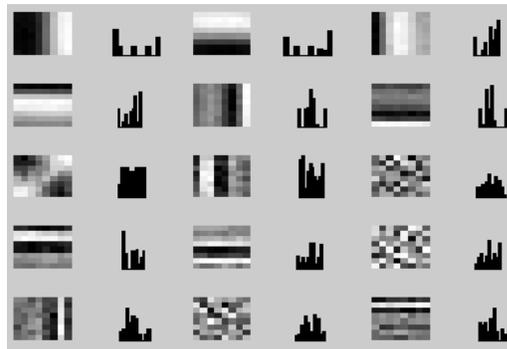


Figura 4.70: `dss_fastICA`. Evaluación de cada columna de \mathbf{A} para datos con ruido multiplicativo.

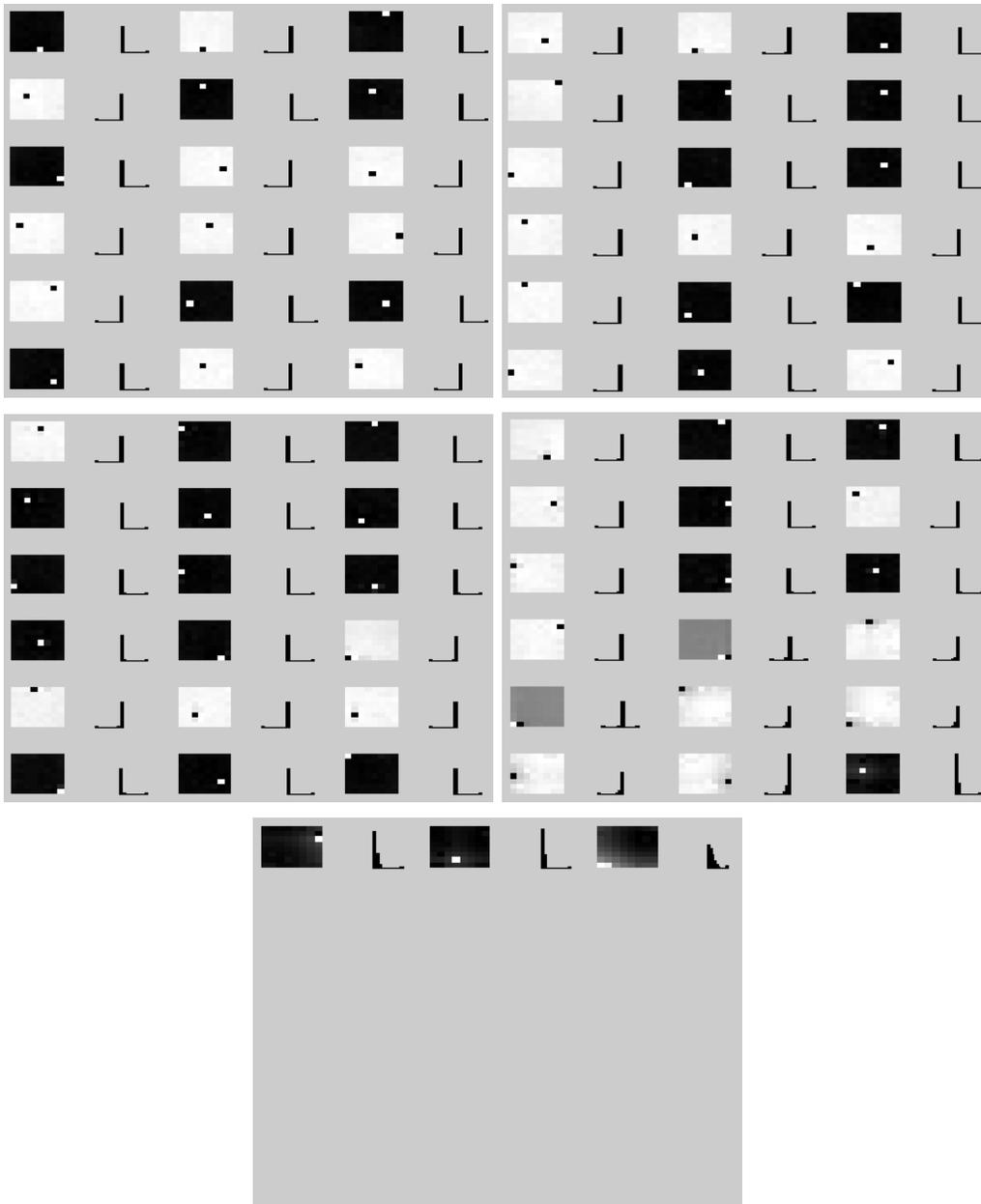


Figura 4.71: dss_fastICA. Evaluación de cada columna de \mathbf{A} para datos con ruido de sal y pimienta.

si la clase de ruido fuese multiplicativo y no gaussiano; mientras que para la figura 4.70, la cual esta representando el análisis con ruido multiplicativo las gráficas del histograma si se comportan con la clase de ruido tratado en este caso, cosa que también ocurre para el ruido de sal y pimienta representado en 4.71 y al igual que la mayoría de los algoritmos estudiados en este trabajo, este algoritmo también parece presentar problemas con esta clase de ruido impulsivo ya que la información se vuelve a presentar en forma de cuadros blancos o negros con un punto diferente.

Continuando con el paso siguiente de la evaluación, en las figuras 4.72 y 4.73 se presentan las comparaciones realizadas con las matriz de muestras original para ruido gaussiano y ruido multiplicativo respectivamente. La comparación realizada entre las componentes originales y las componentes con ruido de sal y pimienta no se presenta ninguna componente estable para una alta correlación por valores por encima de 0,9, para los cual como en casos anteriores se busco correlaciones hasta 0,5 incluido, presentando los resultados mostrados en la figura 4.74.

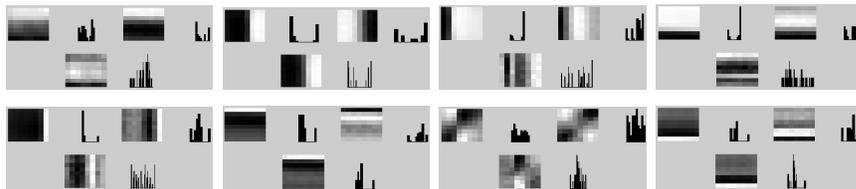


Figura 4.72: Algoritmo `dss_fastICA`. Primeros ocho filtros de componentes entre la señal original y la señal de ruido gaussiano de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación.

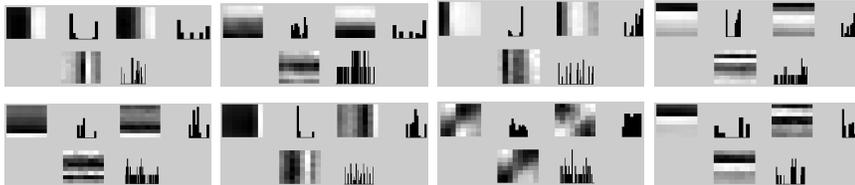


Figura 4.73: Algoritmo `dss_fastICA`. Primeros ocho filtros de componentes entre la señal original y la señal de ruido multiplicativo de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación.

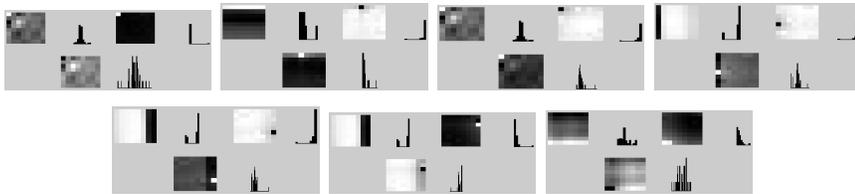


Figura 4.74: Algoritmo `dss_fastICA`. Filtros de componentes entre la señal original y la señal de ruido de sal y pimienta de las respectivas matrices de mezcla \mathbf{A} , que conservan una alta correlación.

Realizada la comparación de la información se obtienen las figuras 4.75 en la que se representa los resultados de la información entre los datos originales y los datos con ruido gaussiano, la figura 4.76 muestra los resultados de la información entre los datos originales y los datos con ruido multiplicativo y 4.77 que representa los resultados de la información entre los datos originales y los datos con ruido de sal y pimienta. De aquí se observa que muchas componentes presentan información que pueden representar bordes o partes importantes de la información original que no han sido eliminadas; esto se presenta para las tres comparaciones realizadas tanto en la información de las señales con datos originales así como en la información con ruido. De hecho hay muy pocas componentes que parecen representar sólo ruido a pesar del análisis previo de correlación.

El análisis llevando en los datos de ruido de sal y pimienta, los puntos altos a 0 es mostrado en la figura 4.78, antes y después de eliminar las componentes que presentan estabilidad en el desarrollo del análisis. En esta figura puede verse que si no es llevado a cabo el análisis de correlación no podría decidirse a simple vista que filtros representan información importante, con el fin de ser eliminados; de hecho en la información en su mayoría mostrada es ruido.

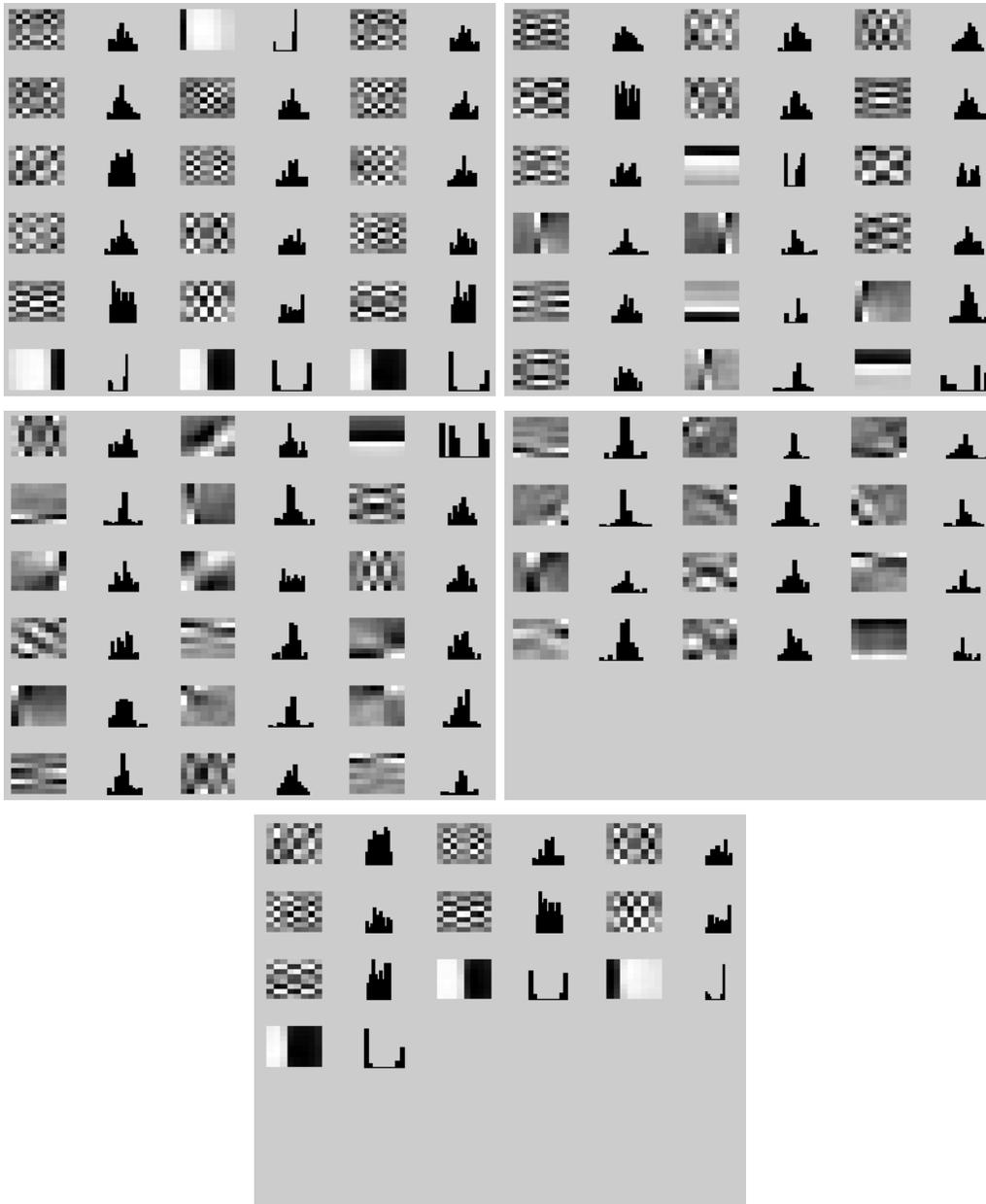


Figura 4.75: Algoritmo `dss_fastICA`. Comparación entre los filtros originales (parte superior y media) y los filtros de componentes con ruido gaussiano (parte inferior) de las respectivas matrices de mezcla \mathbf{A}

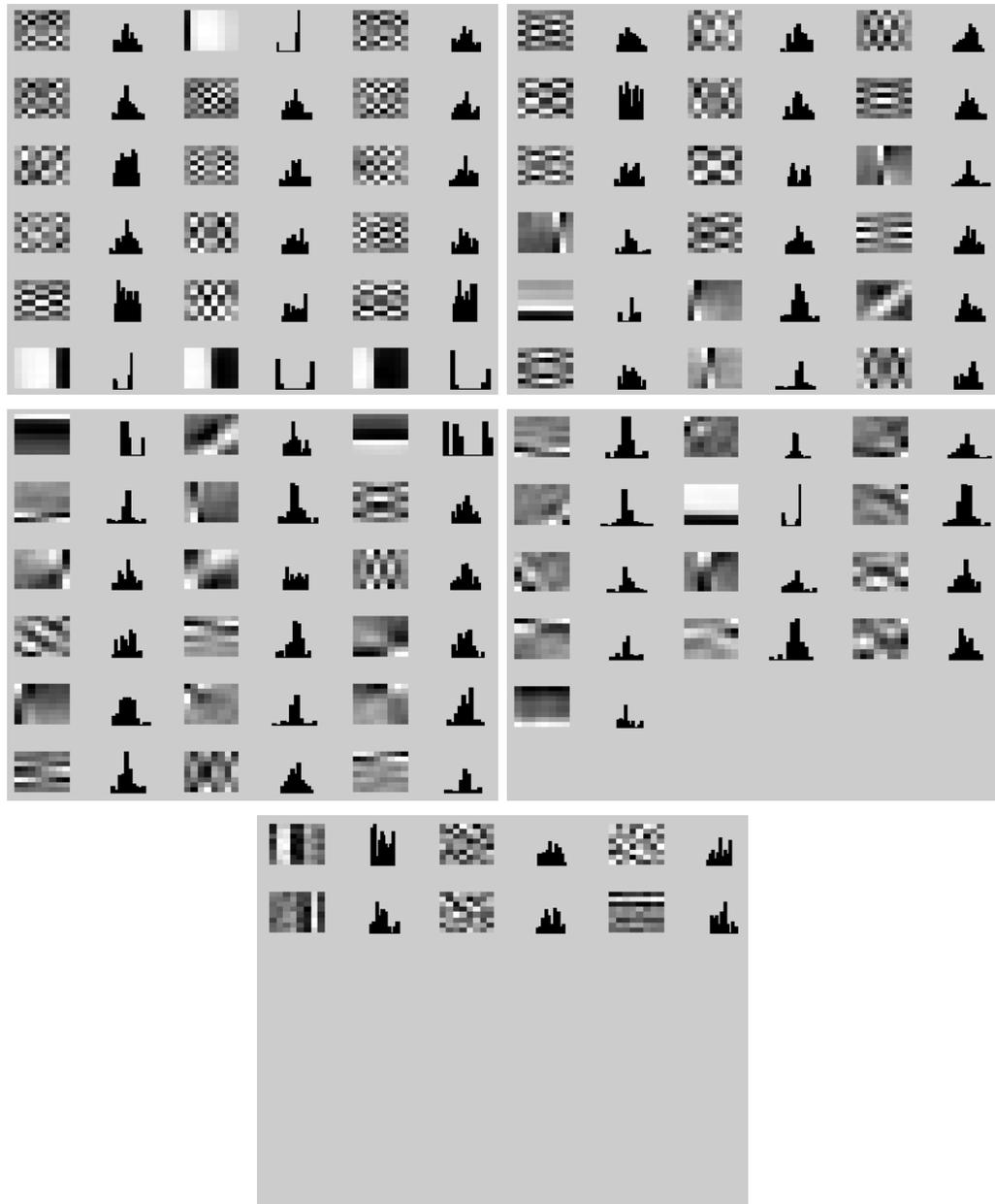


Figura 4.76: Algoritmo `dss_fastICA`. Comparación entre los filtros originales (parte superior y media) y los filtros de componentes con ruido multiplicativo (parte inferior) de las respectivas matrices de mezcla \mathbf{A}

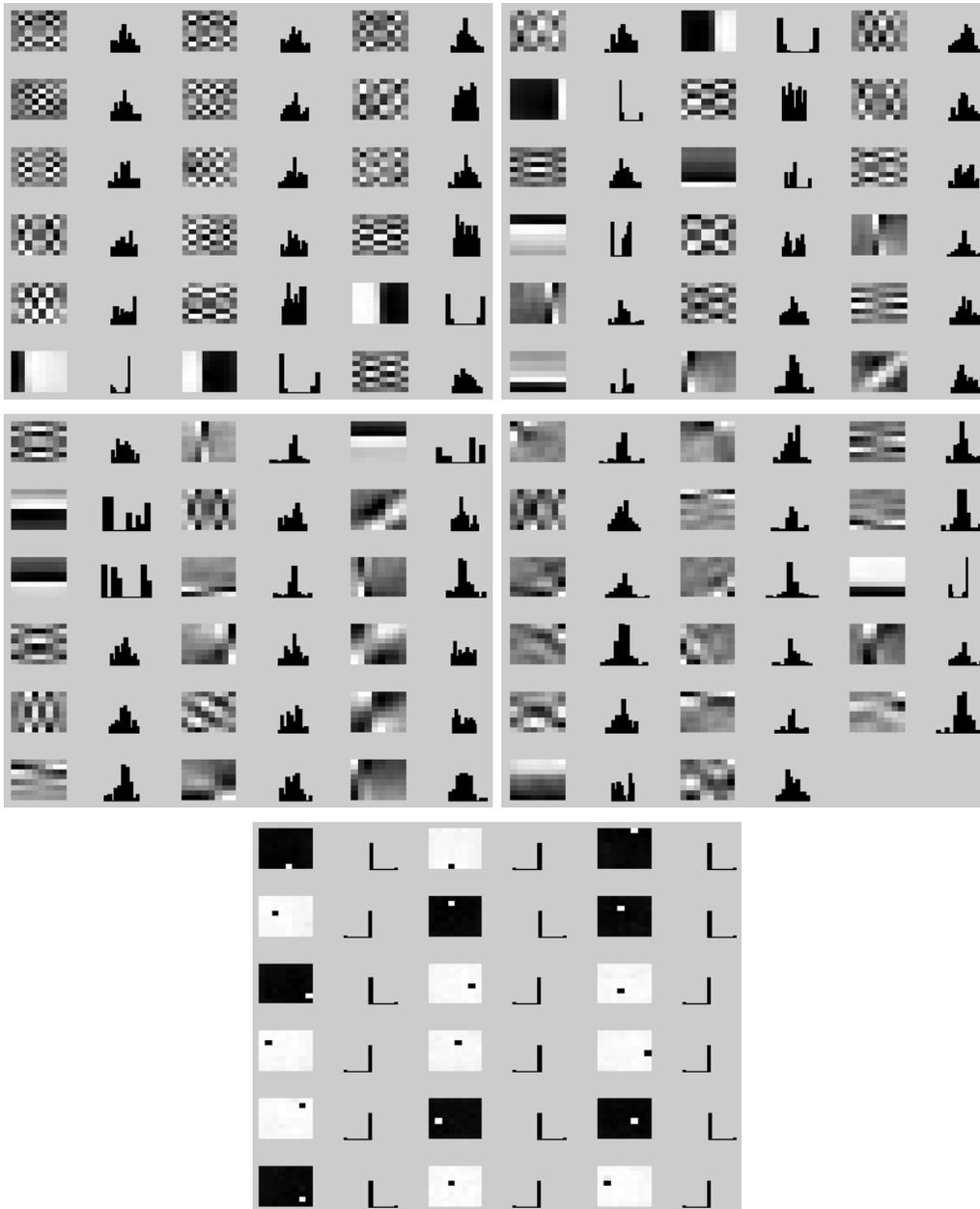


Figura 4.77: Algoritmo `dss_fastICA`. Comparación entre los filtros originales (parte superior y media) y los filtros de componentes con ruido de sal y pimienta (parte inferior) de las respectivas matrices de mezcla \mathbf{A}



Figura 4.78: Algoritmo `dss_fastICA`. Comparación de las diferentes componentes resultantes llevando a 0 los puntos mas altos con el fin de buscar que hay debajo. En la parte superior todas los filtros aún sin realizar la correlación. En la parte inferior todo los filtros una vez llevada a cabo la correlación.

4.4. Comportamiento de AMUSE

En este punto del trabajo se ha seleccionado el algoritmo AMUSE debido a que presenta una serie de particularidades las cuales ya se han mencionado en 4.3.3. Retomando nuevamente sobre estos puntos, se destaca el comportamiento del algoritmo frente a información con la clase de ruido impulsivo (propriadamente ruido de sal y pimienta) y el desarrollo estable en el que nuevamente, sin importar la clase de ruido presente en la información los filtros para comportarse siempre de forma constante.

En primer lugar se analiza el comportamiento de las bases de la matriz de mezcla para cada uno de los casos (señales originales y señales con ruido), realizando previamente el una reducción de dimensiones por medio de PCA, para este paso los algoritmos utilizados son `remmean.m`, `whitenv.m` y `pcamat.m`, con los cuales se extrae la media, se realiza la reducción de la dimensionalidad y el blanqueado respectivamente. Estos programas han sido extraídos del paquete de `fastICA` para matlab y más información puede encontrarse en [1].

La reducción de las dimensiones se ha llevado a cabo para cada uno de los casos y reduciendo sucesivamente a 49, 36, 25, 16 y 9. Los resultados obtenidos pueden observarse en las figuras 4.79 para la reducción de las señales originales, 4.80 para la reducción de las señales con ruido gaussiano, 4.81 para la reducción de las señales con ruido multiplicativo y 4.82 para la reducción de las señales con ruido de sal y pimienta.

Tomando una de las imágenes anteriores por ejemplo la 4.79, ésta representa la comparación entre cada uno de los filtros al reducir las dimensiones. En primer lugar se presenta la reducción a 49, para pasar después a 36, 25, 16 y terminar con 9 dimensiones. Con el fin de conocer que filtros se comportan de manera estable y si pudiese existir alguna conexión entre las diferentes clases de datos, se realiza una correlación entre los datos entregados; pero de la misma clase de información, el factor de correlación para este caso es evaluado por encima de 0,7, debido que se pueden observar gran cantidad de filtros que parecen describir información de ruido. El proceso llevado a cabo en este punto se describe a continuación: los primeros resultados de la reducción a 49 dimensiones son organizados para ser comparados con los resultados representados en la reducción a 36 dimensiones. Así sucesivamente es comparada cada información hasta llegar a los resultados entregados por la reducción a 16 dimensiones con la reducción a 9 dimensiones.

De esta información puede verse de nuevo la estabilidad en los resultados presentados. Sin importar la clase de información, AMUSE para describir resultados estables en los filtros entregados, en donde cada iteración de comparación entre las diferentes reducciones dimensionales presenta constancia y concentración de filtros estables en las primeras 8 posiciones e igualmente en su mayoría los últimos 3 o 4 filtros. Esto puede verse representado sobre las figuras que actualmente se discuten (4.79, 4.80, 4.81 y 4.82), en donde los cuadros en rojo resaltan la constancia de los filtros estables (primeros y últimos filtros) que se mantienen a lo largo de cada reducción de dimensionalidad,

llegando a obtener finalmente una reducción a 9 dimensiones en la que se presentan filtros que demuestran claramente bordes y esquinas que parecen describir información importante; incluso, esta reducción puede también centrarse para el caso de las imágenes de muestra o imágenes originales en una PCA que represente reducción hasta 25 dimensiones, con el fin de conservar otras componentes que para la figura 4.79 parecen contener información de relevancia. Caso contrario ocurre con la otra información representada en las imágenes de ruido gaussiano, multiplicativo y de sal y pimienta (4.80, 4.81, 4.82 respectivamente), en donde puede apreciarse claramente que se hace necesaria la aplicación de PCA hasta llegar a 16 dimensiones, ya que para 25 todavía se aprecian filtros con información que describe ruido.

4.5. Conclusiones

En este capítulo se han analizado 5 diferentes clases de algoritmos que se enfocan en la técnica ICA y el comportamiento de los mismos frente a imágenes que no contienen ruido e imágenes con diferentes clases de ruido. De este estudio pueden extraerse las siguientes conclusiones que a nivel general son:

- El cerebro cumple una función activa importante en el momento de optimizar la información que recibe. Es muy importante la acción que realizan las células en la retina, en especial los conos y bastones en respuesta a la cantidad de intensidad luminosa.
- De acuerdo a la forma en las características y forma de trabajo de cada

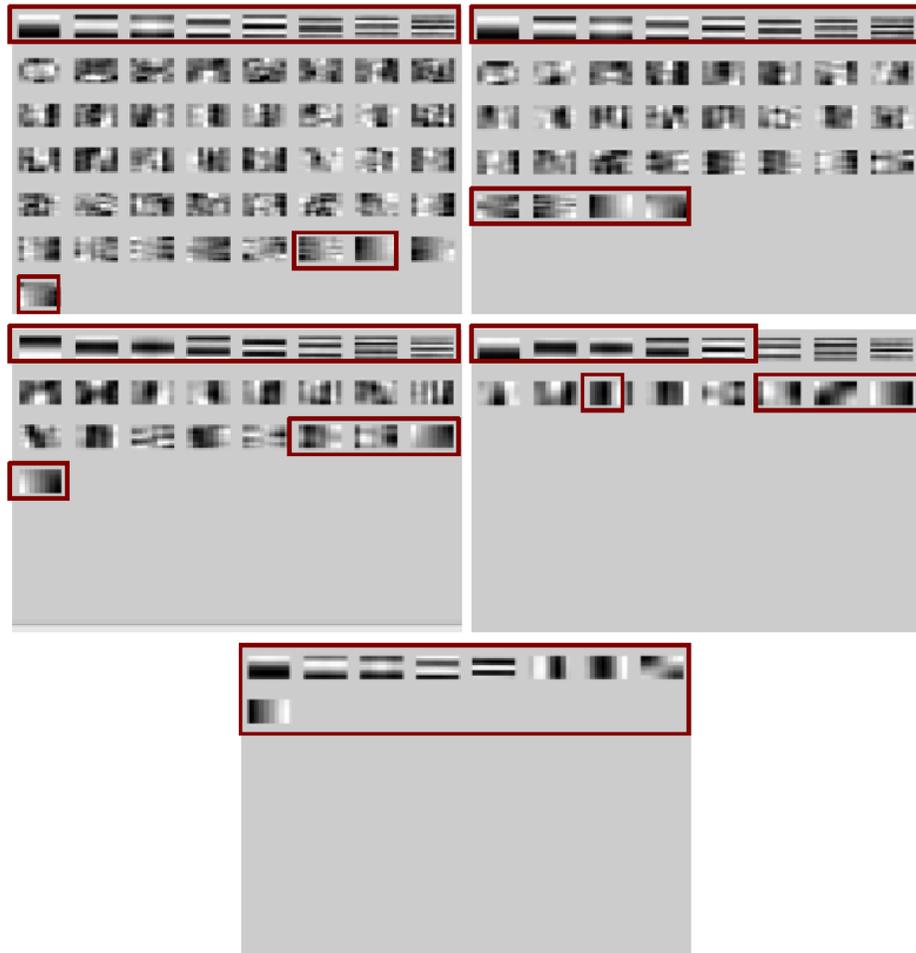


Figura 4.79: Algoritmo AMUSE. Resultados de aplicar el algoritmo AMUSE realizando previamente PCA a las señales originales. Parte superior izquierda, reducción de dimensiones a 49; parte superior derecha, reducción de dimensiones a 36; parte central izquierda, reducción de dimensiones a 25; parte central derecha, reducción de dimensiones a 16 y parte inferior, reducción de dimensiones a 9. Con este número reducido de dimensiones puede apreciarse filtros con información libre de mezclas.

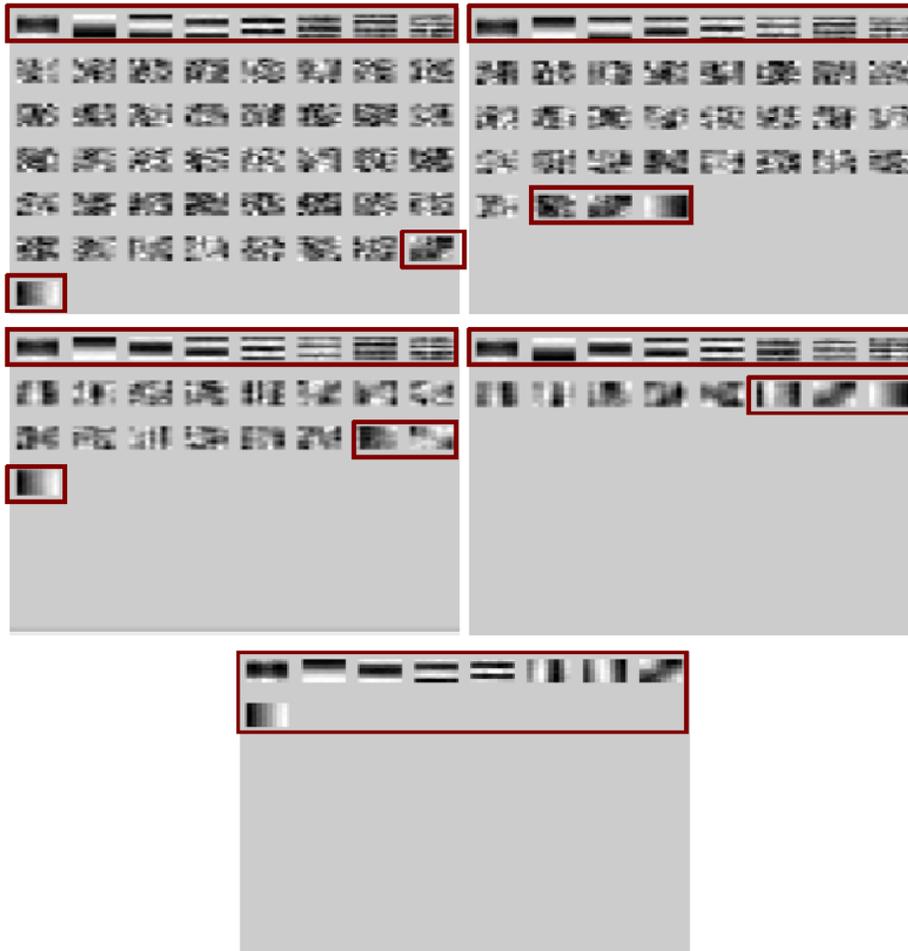


Figura 4.80: Algoritmo AMUSE. Resultados de aplicar el algoritmo AMUSE realizando previamente PCA a las señales con ruido gaussiano. Parte superior izquierda, reducción de dimensiones a 49; parte superior derecha, reducción de dimensiones a 36; parte central izquierda, reducción de dimensiones a 25; parte central derecha, reducción de dimensiones a 16 y parte inferior, reducción de dimensiones a 9. Puede observarse al final la obtención de una determinada reducción de filtros libres de cualquier influencia de ruido gaussiano.

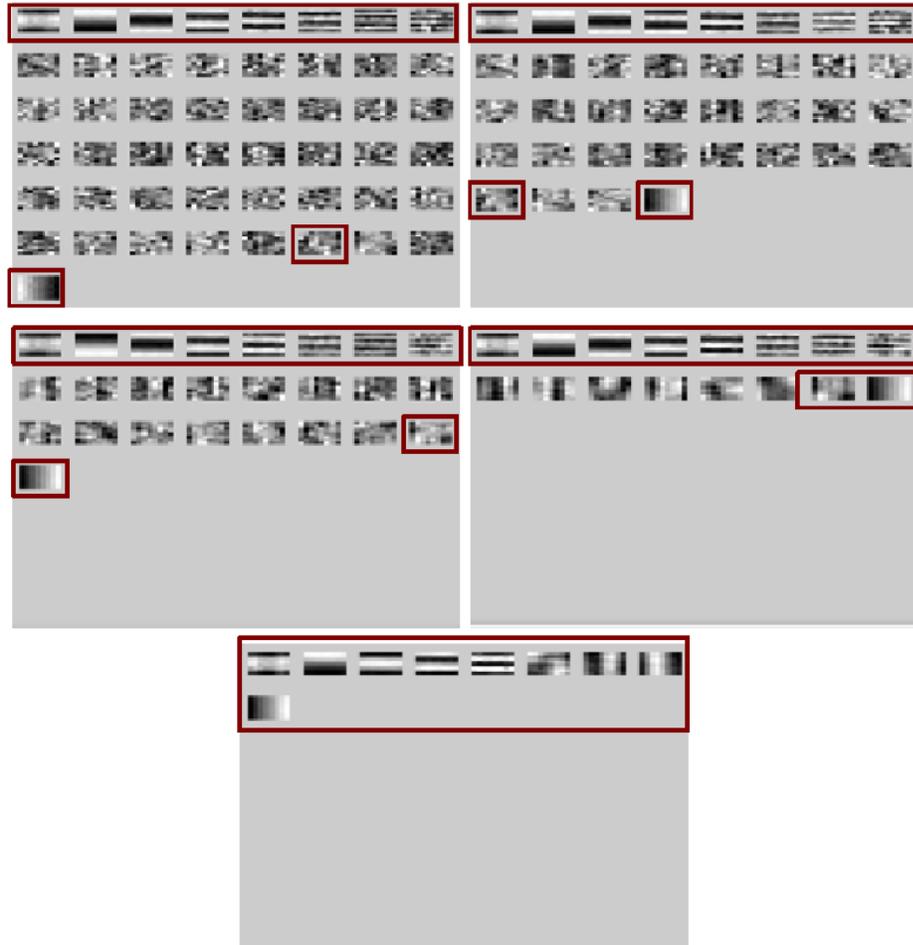


Figura 4.81: Algoritmo AMUSE. Resultados de aplicar el algoritmo AMUSE realizando previamente PCA a las señales con ruido multiplicativo. Parte superior izquierda, reducción de dimensiones a 49; parte superior derecha, reducción de dimensiones a 36; parte central izquierda, reducción de dimensiones a 25; parte central derecha, reducción de dimensiones a 16 y parte inferior, reducción de dimensiones a 9. Puede observarse una reducción de filtros que no poseen influencia de ruido multiplicativo.

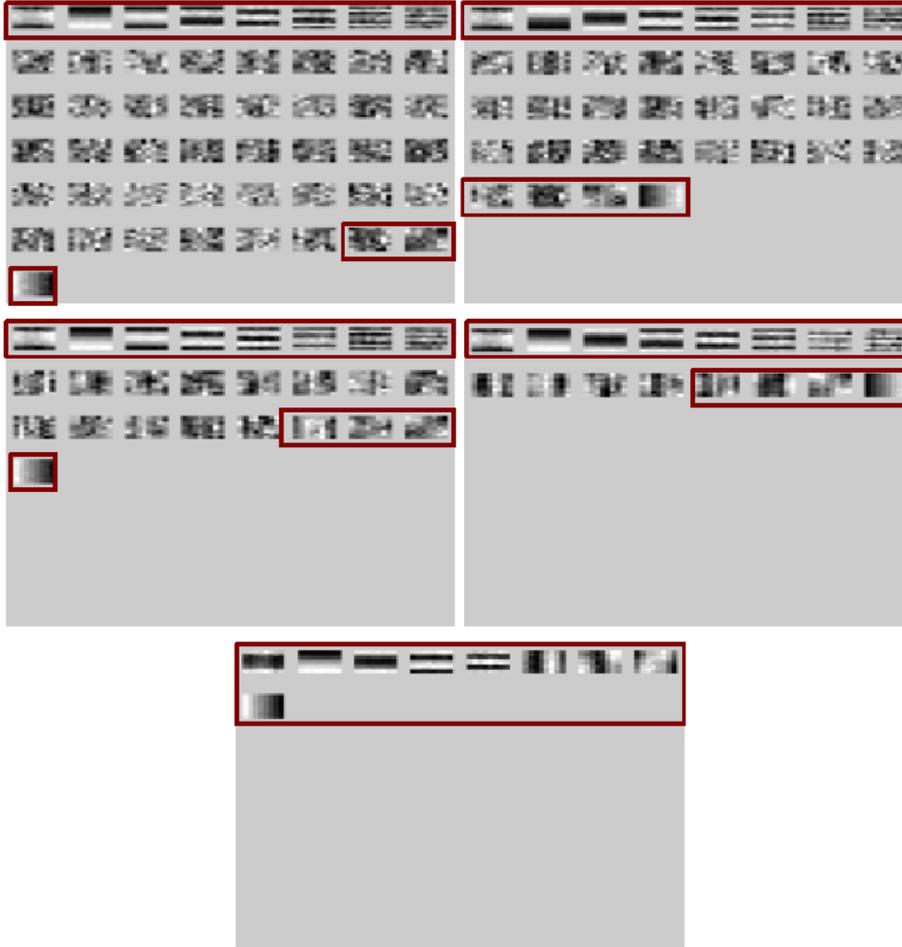


Figura 4.82: Algoritmo AMUSE. Resultados de aplicar el algoritmo AMUSE realizando previamente PCA a las señales con ruido de sal y pimienta. Parte superior izquierda, reducción de dimensiones a 49; parte superior derecha, reducción de dimensiones a 36; parte central izquierda, reducción de dimensiones a 25; parte central derecha, reducción de dimensiones a 16 y parte inferior, reducción de dimensiones a 9. Puede observarse al final que la información contenida en los filtros resultantes, son datos libres de ruido de sal y pimienta.

algoritmo; se han obtenido diferentes respuestas para cada clase de ruido, presentado como desventaja a la mayoría de los algoritmos a la hora de trabajar con ruido impulsivo a excepción del algoritmo AMUSE en el que se destaca la representación de información presente en la matriz de mezcla \mathbf{A} .

A nivel particular para el análisis de las diferentes clases de algoritmos las conclusiones que se han podido obtener son:

fastICA: al trabajar este algoritmo de forma estadística, para el caso de trabajos enfocados en el análisis de información hace necesario el desarrollo previo de pruebas de estabilidad con el fin de obtener componentes constantes que permitan una correcta continuidad de la investigación. Para las diferentes clases de ruido tratado, fastICA ha demostrado comportarse de forma satisfactoria para los casos de ruido gaussiano y multiplicativo; pero en el caso de ruido de sal y pimienta no sucede lo mismo, ya que los primeros resultados obtenidos parece que se trabajase con información que no contienen nada más aparte de ruido. Igualmente, las pruebas del histograma realizadas demuestran que aquellos filtros en los cuales no se parece describir ninguna información como bordes o esquinas, sino que al contrario se encuentra información aleatoria que describe ruido, el comportamiento presente para la mayoría de los filtros es en su caso una distribución típica de ruido gaussiano (4.20), multiplicativo (4.22) e impulsivo (4.23).

JADE: como se ha mencionado en el capítulo 3.3 JADE es un algoritmo

que trabaja con estadísticos de alto orden y la diagonalización de la descomposición de los autovalores de las mezclas. A diferencia del algoritmo anterior, JADE no requiere de la realización de pruebas de estabilidad; pero como mayor punto débil se presenta el alto tiempo computacional. Propiamente en el desarrollo de este análisis el algoritmo necesito de más de un día con el fin de completar todo el proceso. Para los resultados de las muestras contaminadas con ruido gaussiano JADE ha mostrado la mayoría de los filtros con la forma de la distribución gaussiana (4.33); pero no ha mostrado resultado de filtros en los que se pueda apreciar información. Para el ruido multiplicativo (4.34) el algoritmo ha presentado descripción de bordes; pero muchas de las componentes presentadas como sólo ruido parece que se comportaran con distribución gaussiana. Al igual que fastICA, JADE también presenta resultados similares con el ruido de sal y pimienta.

AMUSE: los resultados entregados por el algoritmo AMUSE difieren un poco en relación a los dos algoritmos cuando se habla del ruido impulsivo. En estos resultados puede observarse para esta clase de ruido que así como existe presencia de información que describe bordes, igualmente existe la presencia de filtros de sólo ruido los cuales en ningún caso se comportan como la distribución típica descrita en 4.19; sino caso contrario, en la mayoría de los casos se esta describiendo distribuciones con ruido gaussiano y otras pocas con ruido multiplicativo. Este mismo caso se produce en las muestras contaminadas con ruido

gaussiano (4.43) y multiplicativo (4.44) en donde no hay constancia del tipo de ruido aplicado.

fastGEO: los resultados de las muestras contaminadas con ruido gaussiano (4.56) presentan para los filtros con ruido distribuciones gaussianas que incluso hacen fácil pensar a simple vista en una forma rápida de identificación de la clase de ruido. Caso contrario ocurre cuando nos referimos al ruido multiplicativo (4.57), en donde al igual que para otros algoritmos la identificación de algunos filtros haría pensar en contaminación con ruido gaussiano. El ruido impulsivo (4.58) aquí se comporta nuevamente como lo a hecho con los algoritmos de fastICA y JADE.

dss_fastICA: hay que recordar que el desarrollo de este algoritmo recurre en primera instancia a realizar una PCA. Al igual que para fastICA en dss_fastICA se ha recurrido a las mismas pruebas de estabilidad del algoritmo. Es muy interesante observar los resultados entregados por este algoritmo, ya que en la evaluación de los datos contaminados con ruido gaussiano (4.69), se ha obtenido una respuesta donde tan sólo se presentan 19 filtros y en los cuales puede apreciarse claramente la descripción de bordes o esquinas. Caso parecido ocurre con el ruido multiplicativo (4.70), en donde a excepción de 3 de los filtros presentados en los resultados, los demás describen información interesante. Pero al igual que para la mayoría de los algoritmos el comportamiento de dss_fastICA en el caso de ruido de sal y pimienta se presenta como la

presencia de sólo ruido en los datos de información.

Capítulo 5

Aplicación al De-noising

El ruido es una parte importante en el análisis de imágenes. Cuando se tiene un mejor conocimiento del éste y de sus causas, pueden desarrollarse algoritmos más adecuados para llevar a cabo una correcta eliminación en la cadena de pre-procesado de imágenes.

En este capítulo se va a llevar a cabo una aplicación basada en ICA para eliminar el ruido (gaussiano, multiplicativo y sal y pimienta), de las imágenes de muestra; e igualmente, también se desarrolla una comparación con el método de PCA para el mismo grupo de imágenes.

En primera instancia se explicará el método a seguir bajo el cual se aplican las ideas revisadas con anterioridad, con el fin de eliminar el ruido presente en las imágenes de muestra, para paso seguido observar los resultados entregados y terminar realizando una comparación entre las técnicas de ICA y PCA.

5.1. Método desarrollado

En el capítulo anterior se ha analizado el comportamiento de algoritmo AMUSE y se ha desarrollado una reducción de dimensiones para cada uno de los tipos de ruido, que ha permitido llegar hasta encontrar 9 filtros los cuales se presentan sin ningún tipo de contaminación independientemente de que las señales observadas se encuentren contaminadas. (ver figuras 4.80, 4.81 y 4.82).

Con esta reducción de dimensiones se obtiene un sistema que en muchos casos puede presentar un exceso o defecto de ecuaciones. Para casos en concreto como este tipo, no se busca calcular la matriz inversa, sino que busca una solución cuyo error cumpla una condición de la norma mínima o busque un subespacio de soluciones, generando un resultado aproximadamente igual a la matriz unitaria. Es por esto que para estos casos se dice que en vez de buscar la inversa, se calcula una pseudoinversa que resuelva la ecuación [83]. Por este motivo que al presentarse nuestra nueva matriz de mezcla \mathbf{A} como una matriz de dimensiones de 64 filas por 9 columnas se hace necesario recurrir en matlab a la función `pinv` para poder hallar la ecuación de separación \mathbf{W} .

Matemáticamente, denotando la matriz \mathbf{M} , se define como pseudoinversa \mathbf{M}^+ , a una matriz que multiplicada por \mathbf{M} , entrega aproximadamente la matriz diagonal con todos los elementos diagonales iguales a 1:

$$\mathbf{I} \approx \mathbf{M}\mathbf{M}^+ \tag{5.1}$$

Para calcular la matriz pseudoinversa de \mathbf{M} se recurre a la ecuación:

$$\mathbf{M}^+ = \mathbf{M}^T [\mathbf{M}^T \mathbf{M}]^{-1} \quad (5.2)$$

El cálculo de la expresión representada en 5.2, representa $[\mathbf{M}^T \mathbf{M}]$ como una matriz cuadrada, la cual puede ser invertida y su inversa es única. Para el caso de la función en matlab, ésta recurre igualmente al uso del valor de descomposición singular, por sus siglas en ingles Singular Value Descomposition (SVD), en el cual toda matriz \mathbf{M} de tamaño $m \times n$ permite descomposiciones de la forma $\mathbf{M} = \mathbf{U} \mathbf{N} \mathbf{V}$, donde para este caso, las matrices $\mathbf{U}_{m \times m}$ y $\mathbf{V}_{n \times n}$ son unitarias y si \mathbf{M} es real, ortogonales y $\mathbf{N}_{m \times n}$ representa una matriz diagonal cuyos elementos diagonales son los valores singulares de \mathbf{M} . Más información pueden encontrarse en [83], [95], o en general, libros de álgebra lineal.

Una vez calculada la matriz de separación \mathbf{W} para cada tipo de ruidos y la cual se haya por medio de la ecuación representada en 1.4 del capítulo 1, se pasa a calcular las fuentes representadas por medio de la matriz \mathbf{S} . Para este desarrollo se divide el camino en tres grupos los cuales representan las 3 clases de ruido (gaussiano, multiplicativo y sal y pimienta) con las que se ha venido trabajando a lo largo de la tesis. Estos caminos se hacen por la necesidad de tomar de manera independiente las observaciones que se representan en la matriz \mathbf{X} de los ruidos. Matemáticamente se va a trabajar con tres ecuaciones sencillas:

$$\mathbf{S} = \mathbf{W}\mathbf{X}_{gau} \quad (5.3)$$

$$\mathbf{S} = \mathbf{W}\mathbf{X}_{mul} \quad (5.4)$$

$$\mathbf{S} = \mathbf{W}\mathbf{X}_{syp} \quad (5.5)$$

donde para cada una de las ecuaciones (5.3, 5.4 y 5.5), se cambian la información de las observaciones.

Paso seguido, para cada una de las componentes (filas) de la matriz \mathbf{S} se hayan los valores absolutos máximos, con el fin de colocarlos en 0 y se crea una variable la cual se denomina “factor”, cuyo valor va a representar una disminución del rango de ruido que es, para ruido gaussiano y de sal y pimienta del 10% y para ruido multiplicativo de un 12%, cuando se aplica la técnica ICA. Las imágenes 5.1, 5.2 representan los resultados obtenidos para las muestras contaminadas con ruido gaussiano, las imágenes 5.3 y 5.4 muestras los datos contaminados con ruido multiplicativo y las imágenes 5.5 y 5.6 muestras los datos contaminados con ruido impulsivo de sal y pimienta.

Para realizar la comparación de este método de reducción de ruido con la técnica PCA, se lleva a cabo el mismo procedimiento anterior, en donde sólo se hace necesario variar el rango de eliminación de ruido multiplicativo a 10% ya que en la realización de las diferentes prueba presentó mejores resultados.

Los resultados de aplicar PCA a los diferentes datos son mostrados para

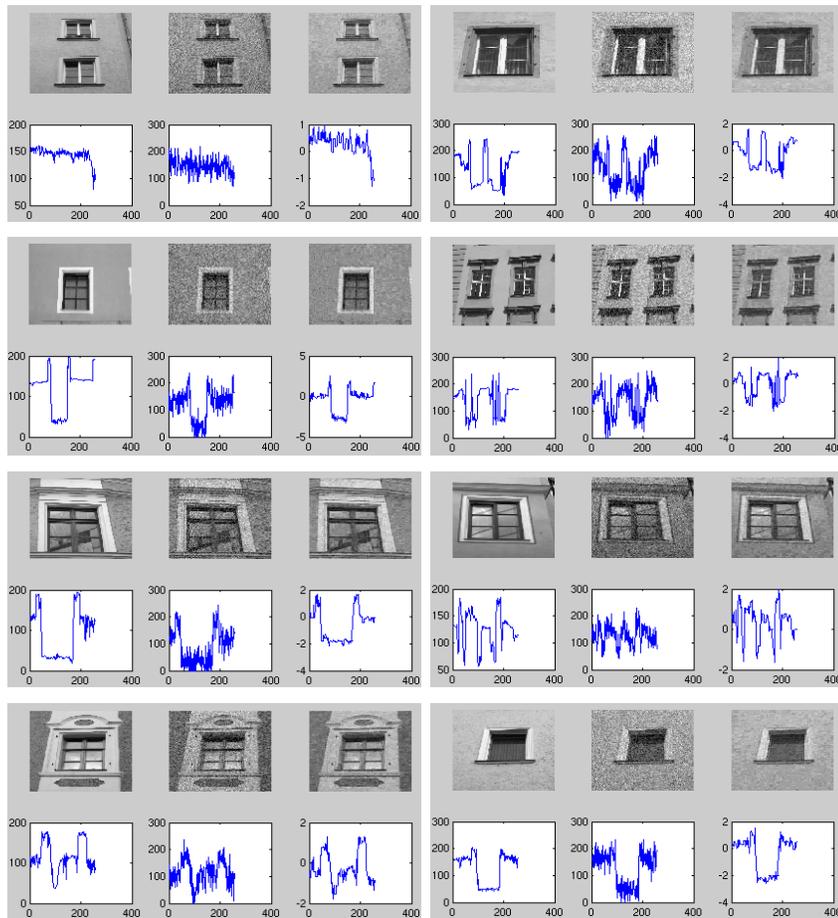


Figura 5.1: Algoritmo AMUSE. Parte A. Resultados obtenidos después de aplicar la técnica ICA al grupo de Imágenes de muestra contaminadas con ruido gaussiano. Imágenes mostradas de la 1 a la 8.

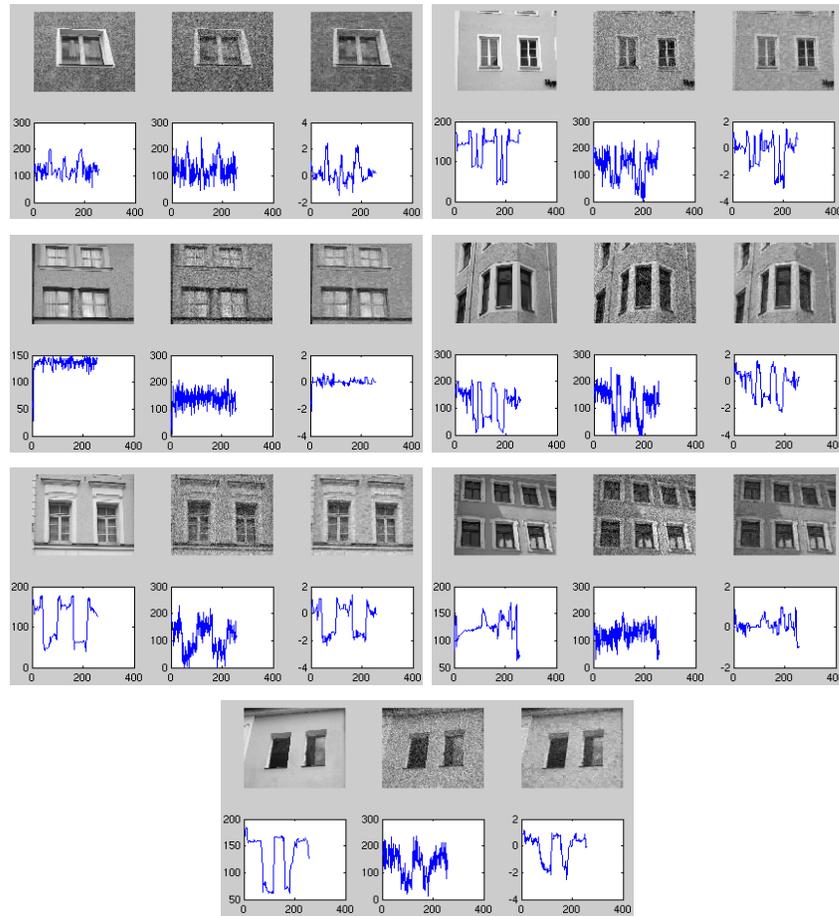


Figura 5.2: Algoritmo AMUSE. Parte B. Resultados obtenidos después de aplicar la técnica ICA al grupo de Imágenes de muestra contaminadas con ruido gaussiano. Imágenes mostradas de la 9 a la 15.

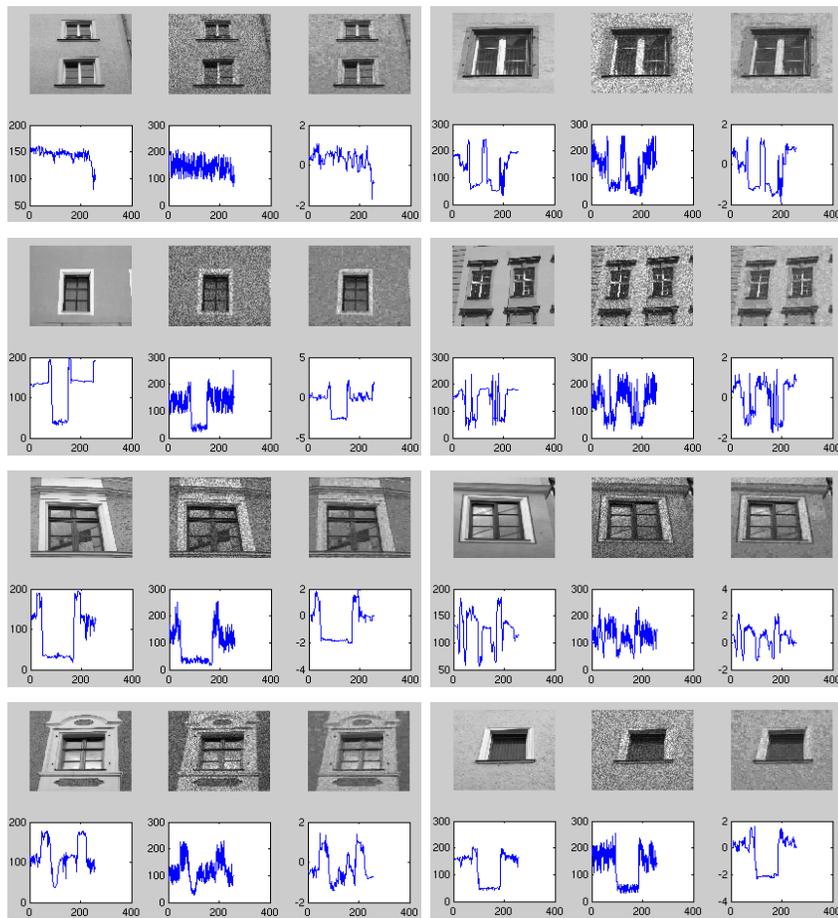


Figura 5.3: Algoritmo AMUSE. Parte A. Resultados obtenidos después de aplicar la técnica ICA al grupo de Imágenes de muestra contaminadas con ruido multiplicativo. Imágenes mostradas de la 1 a la 8.

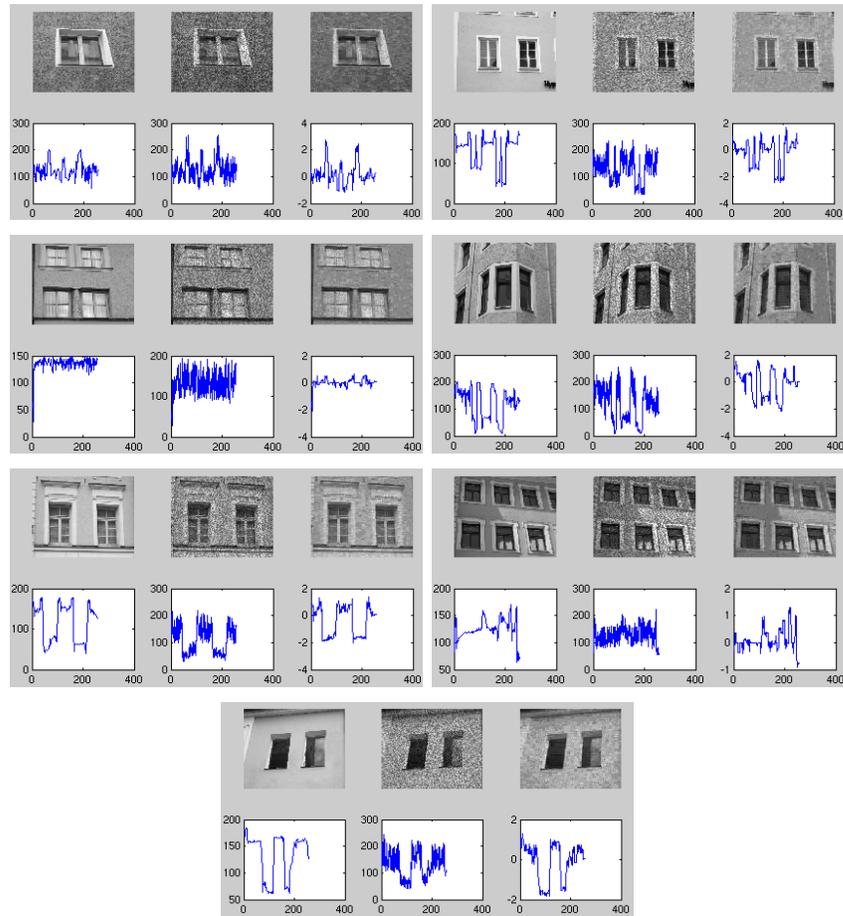


Figura 5.4: Algoritmo AMUSE. Parte B. Resultados obtenidos después de aplicar la técnica ICA al grupo de Imágenes de muestra contaminadas con ruido multiplicativo. Imágenes mostradas de la 9 a la 15.

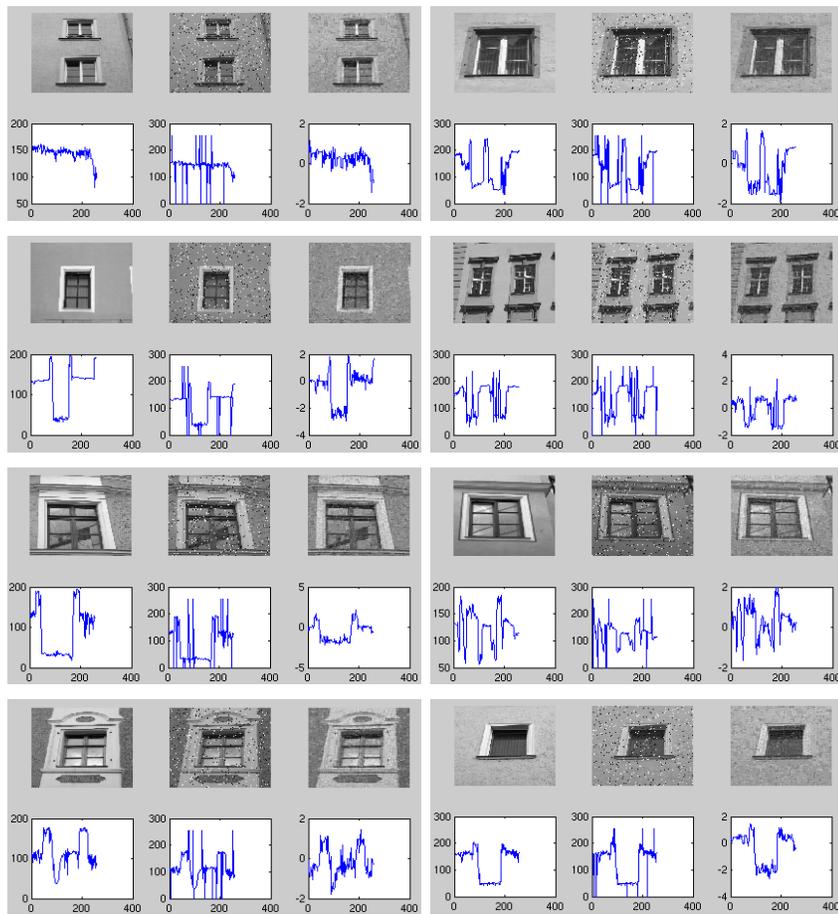


Figura 5.5: Algoritmo AMUSE. Parte A. Resultados obtenidos después de aplicar la técnica ICA al grupo de Imágenes de muestra contaminadas con ruido de sal y pimienta. Imágenes mostradas de la 1 a la 8.

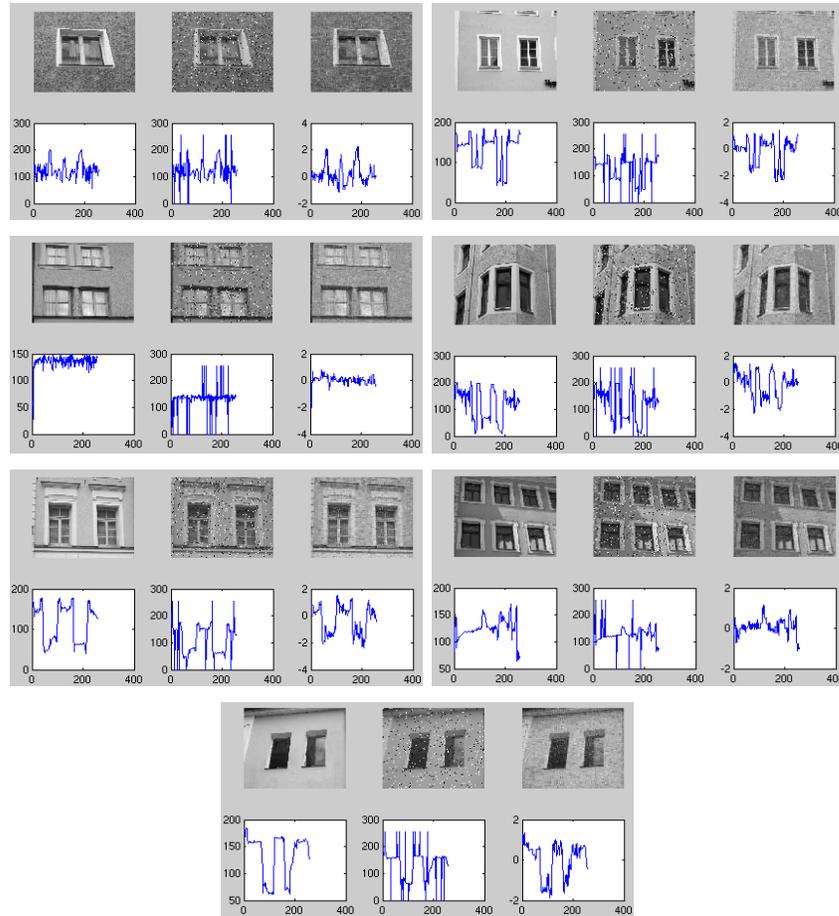


Figura 5.6: Algoritmo AMUSE. Parte B. Resultados obtenidos después de aplicar la técnica ICA al grupo de Imágenes de muestra contaminadas con ruido de sal y pimienta. Imágenes mostradas de la 9 a la 15.

ruido gaussiano en las imágenes 5.7 y 5.8, para la información con ruido multiplicativo en las imágenes 5.9 y 5.10 y para la información con ruido de sal y pimienta en las imágenes 5.11 y 5.12.

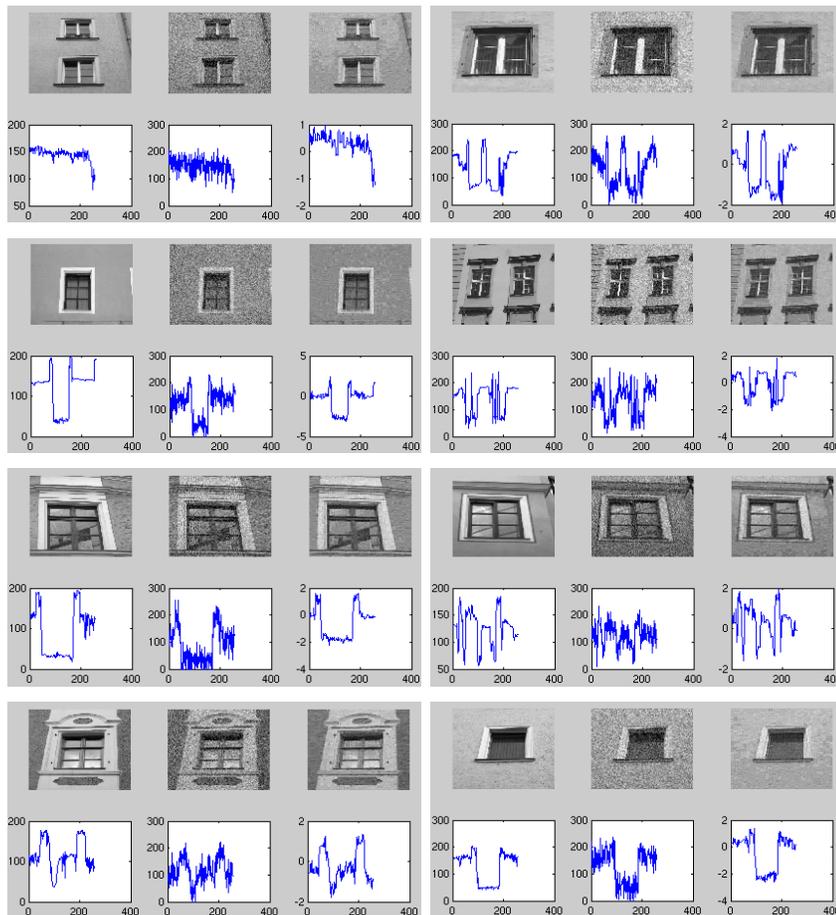


Figura 5.7: Algoritmo AMUSE. Parte A. Resultados obtenidos después de aplicar la técnica PCA al grupo de Imágenes de muestra contaminadas con ruido gaussiano. Imágenes mostradas de la 1 a la 8.

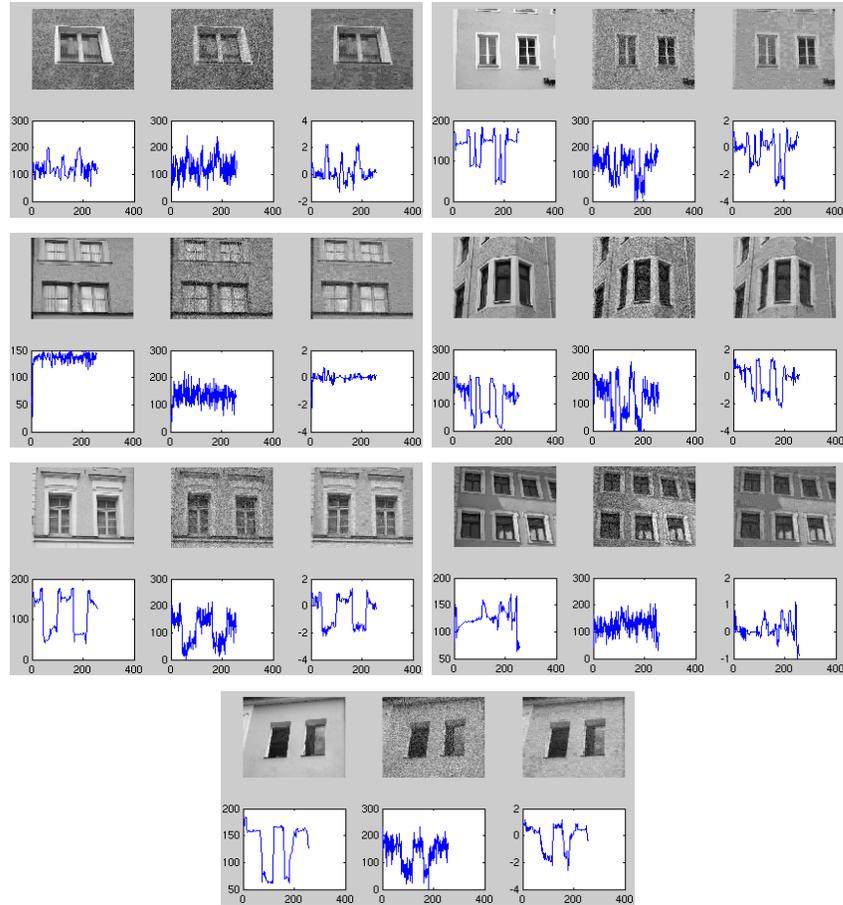


Figura 5.8: Algoritmo AMUSE. Parte B. Resultados obtenidos después de aplicar la técnica PCA al grupo de Imágenes de muestra contaminadas con ruido gaussiano. Imágenes mostradas de la 9 a la 15.

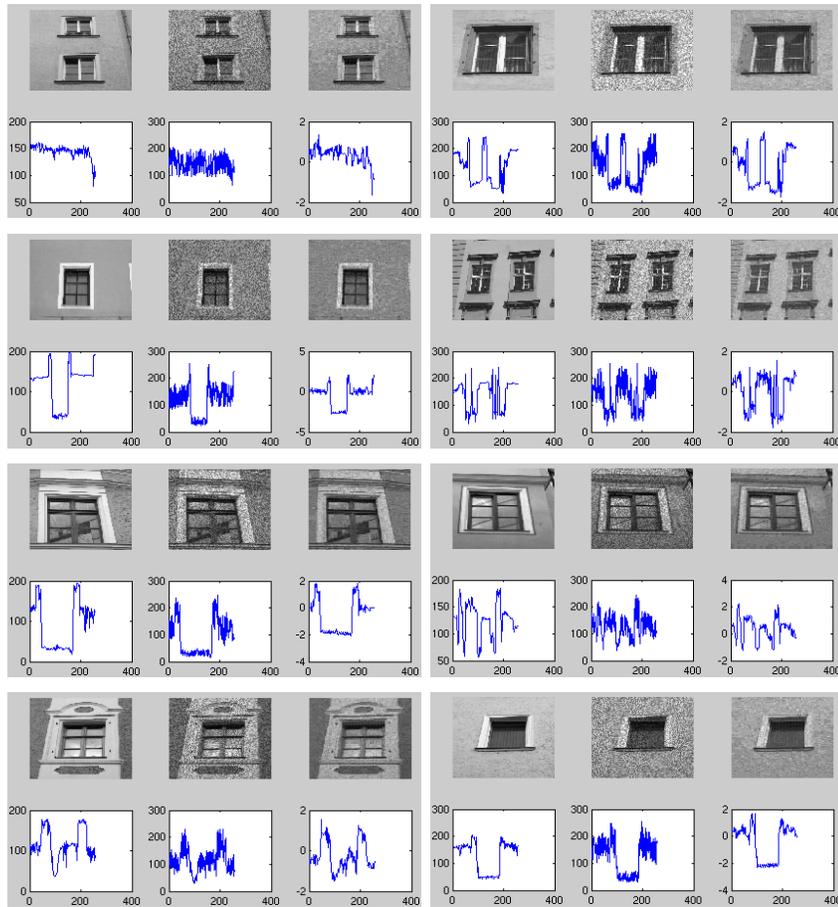


Figura 5.9: Algoritmo AMUSE. Parte A. Resultados obtenidos después de aplicar la técnica PCA al grupo de Imágenes de muestra contaminadas con ruido multiplicativo. Imágenes mostradas de la 1 a la 8.

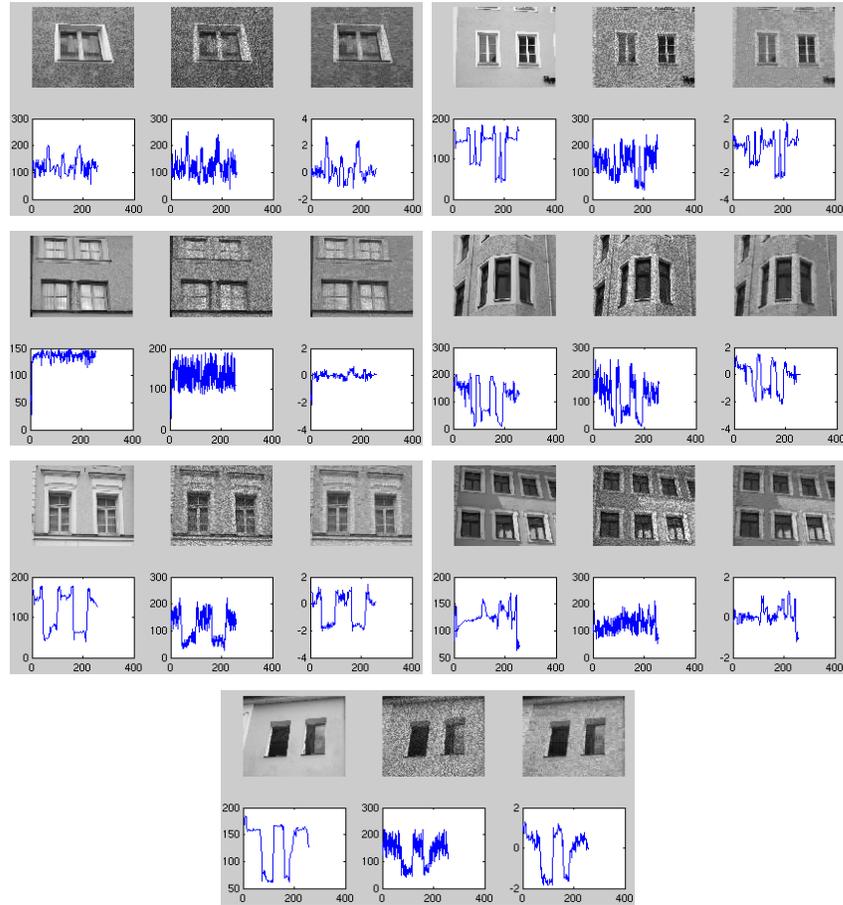


Figura 5.10: Algoritmo AMUSE. Parte B. Resultados obtenidos después de aplicar la técnica PCA al grupo de Imágenes de muestra contaminadas con ruido multiplicativo. Imágenes mostradas de la 9 a la 15.

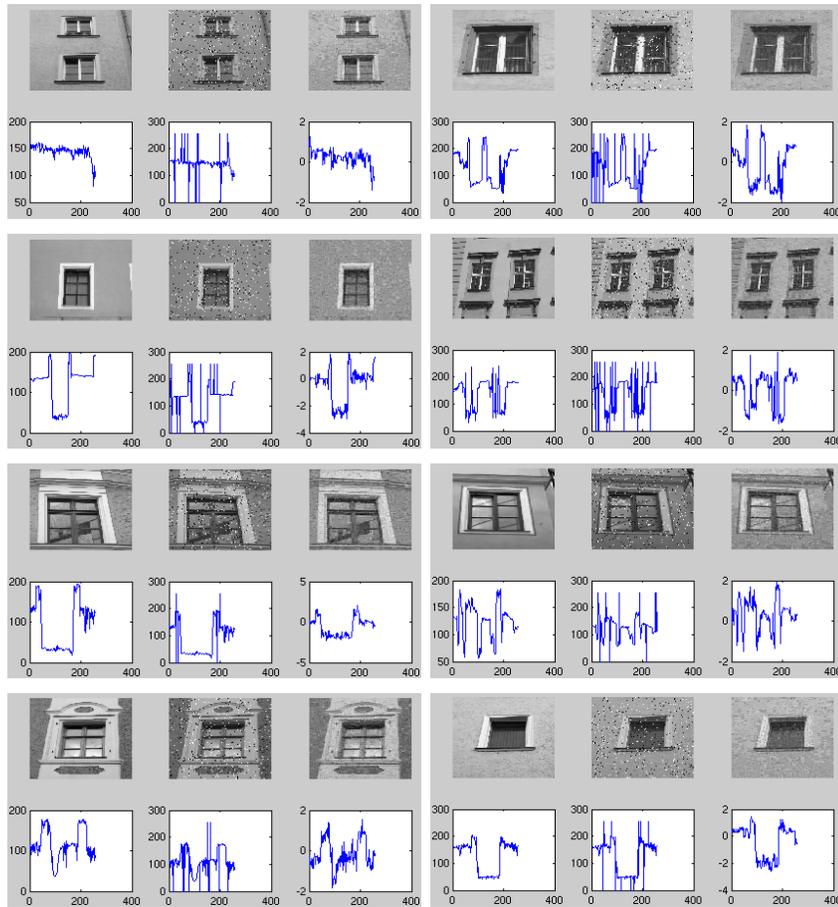


Figura 5.11: Algoritmo AMUSE. Parte A. Resultados obtenidos después de aplicar la técnica PCA al grupo de Imágenes de muestra contaminadas con ruido de sal y pimienta. Imágenes mostradas de la 1 a la 8.

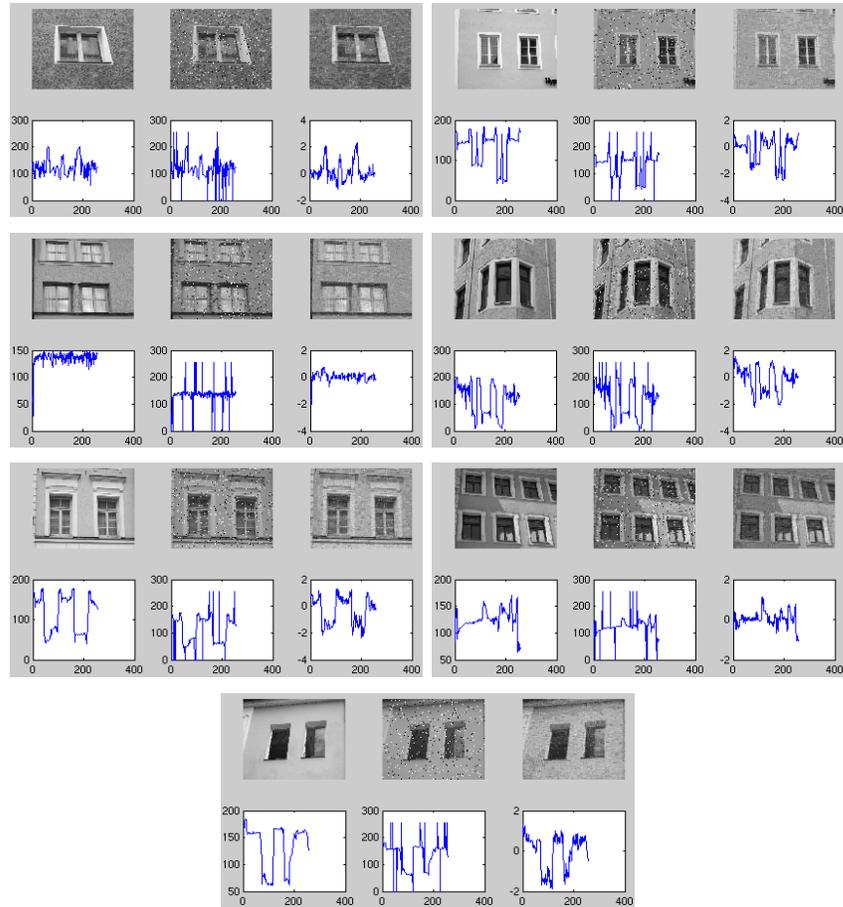


Figura 5.12: Algoritmo AMUSE. Parte B. Resultados obtenidos después de aplicar la técnica PCA al grupo de Imágenes de muestra contaminadas con ruido de sal y pimienta. Imágenes mostradas de la 9 a la 15.

5.1.1. Análisis de resultados

A nivel general, cada imagen representada en cada una de las figuras (5.1 a la 5.12), muestran los resultados obtenidos en cada una de las imágenes muestra (recordemos que en su totalidad, se ha trabajado con 15 imágenes muestra), después de llevar a cabo la eliminación de las diferentes clases de ruido trabajado. La forma como se ha representado la información se explica a continuación, cada imagen contiene 6 gráficos, 3 en la parte superior representan las estructuras y las otras 3 en la parte inferior representan la respectiva distribución de cada matriz de información. Esto es, en la parte superior izquierda se presenta la imagen de muestra original, esto es sin ruido y su respectiva gráfica de la forma de distribución como se presenta la información de la matriz que contiene la imagen en la parte inferior izquierda. En el centro se presenta la misma imagen después de añadir ruido, en cual puede ser cualquiera de los tratados en esta tesis, y al igual que antes se en la parte del centro abajo se puede observar que influencia tiene esta señal parásita, agregada a la señal original. La figura representada en la parte derecha superior contiene la información de la imagen después de llevar a cabo la eliminación de ruido y una mejor observación de los resultados pueden obtenerse en la gráfica de la derecha parte de abajo.

En las figuras representadas de la 5.1 hasta la 5.6 pueden observarse los resultados obtenidos después de evaluar cada una de las matrices de observaciones \mathbf{X} con la técnica ICA. En estos resultados puede observarse la mejora que se presenta en la reducción de ruido especialmente, ruido multiplicativo.

En las otras clases de ruido en especial el de sal y pimienta, puede verse que aunque la mejora es aceptable pueden encontrarse en la actualidad otras herramientas que superan en gran medida los resultados presentados aquí.

Las figuras 5.7 hasta 5.12, presenta los resultados obtenidos utilizando la técnica de PCA. A nivel general estos resultados se muestran más interesantes, debido a que para todas las clases de ruido aquí estudiados los resultados obtenidos parecen presentarse mejor, en especial el ruido impulsivo el cual, a nivel general es considerado como uno de los tipo de ruido más fuertes. Incluso, realizando un análisis de correlación con el objetivo de estudiar que componentes entre las dos técnicas se presentan más correlacionadas, se obtienen mejores resultados de los datos presentados con PCA.

Las siguientes tablas muestran la comparación realizada entre las técnicas ICA y PCA en cuanto al cálculo del error cuadrático medio, para cada uno de los tipos de ruido. En la tabla 5.1 se representa la comparación cuando se trabaja con ruido gaussiano, la tabla 5.2 representa los datos comparados con ruido multiplicativo y por último la tabla 5.3 que contiene la comparación de los datos contaminados con ruido de sal y pimienta.

En estas puede verse como en realidad las diferencias de la mejoras entre las dos técnicas no es muy grande a pesar de que en algunas ocasiones las imágenes puedan presentarse mejor con la técnica PCA.

Cuadro 5.1: Tabla comparativa de las técnicas ICA y PCA en busca de las diferencias en el cálculo del error cuadrático medio. Datos extraídos de las muestras originales y de las contaminadas con ruido gaussiano. Puede apreciarse diferencias mínimas en los resultados.

Imágenes muestra	ICA	PCA
Imagen 1	1,2442e + 09	1,2475e + 09
Imagen 2	1,6330e + 09	1,6335e + 09
Imagen 3	1,2597e + 09	1,2641e + 09
Imagen 4	1,4979e + 09	1,4942e + 09
Imagen 5	1,1484e + 09	1,1548e + 09
Imagen 6	9,0135e + 08	9,0637e + 08
Imagen 7	1,2174e + 09	1,2159e + 09
Imagen 8	1,5021e + 09	1,5015e + 09
Imagen 9	9,8257e + 08	9,8440e + 08
Imagen 10	1,3736e + 09	1,3727e + 09
Imagen 11	1,1918e + 09	1,1915e + 09
Imagen 12	1,2452e + 09	1,2458e + 09
Imagen 13	1,2610e + 09	1,2637e + 09
Imagen 14	1,0637e + 09	1,0659e + 09
Imagen 15	1,3734e + 09	1,3763e + 09

Cuadro 5.2: Tabla comparativa de las técnicas ICA y PCA en busca de las diferencias en el cálculo del error cuadrático medio. Datos extraídos de las muestras originales y de las contaminadas con ruido multiplicativo. Puede apreciarse diferencias mínimas en los resultados.

Imágenes muestra	ICA	PCA
Imagen 1	$1,2501e + 09$	$1,2489e + 09$
Imagen 2	$1,6441e + 09$	$1,6418e + 09$
Imagen 3	$1,2681e + 09$	$1,2672e + 09$
Imagen 4	$1,5083e + 09$	$1,5135e + 09$
Imagen 5	$1,1496e + 09$	$1,1512e + 09$
Imagen 6	$8,9472e + 08$	$8,9443e + 08$
Imagen 7	$1,2213e + 09$	$1,2193e + 09$
Imagen 8	$1,5203e + 09$	$1,5175e + 09$
Imagen 9	$9,7484e + 08$	$9,7471e + 08$
Imagen 10	$1,3821e + 09$	$1,3815e + 09$
Imagen 11	$1,1944e + 09$	$1,1956e + 09$
Imagen 12	$1,2534e + 09$	$1,2497e + 09$
Imagen 13	$1,2689e + 09$	$1,2710e + 09$
Imagen 14	$1,0599e + 09$	$1,0573e + 09$
Imagen 15	$1,3866e + 09$	$1,3889e + 09$

Cuadro 5.3: Tabla comparativa de las técnicas ICA y PCA en busca de las diferencias en el cálculo del error cuadrático medio. Datos extraídos de las muestras originales y de las contaminadas con ruido de sal y pimienta. Puede apreciarse diferencias mínimas en los resultados.

Imágenes muestra	ICA	PCA
Imagen 1	1,2461e + 09	1,2472e + 09
Imagen 2	1,6240e + 09	1,6236e + 09
Imagen 3	1,2650e + 09	1,2640e + 09
Imagen 4	1,4910e + 09	1,4901e + 09
Imagen 5	1,1595e + 09	1,1627e + 09
Imagen 6	9,2722e + 08	9,2545e + 08
Imagen 7	1,2233e + 09	1,2259e + 09
Imagen 8	1,4914e + 09	1,4923e + 09
Imagen 9	9,9566e + 08	9,9829e + 08
Imagen 10	1,3686e + 09	1,3691e + 09
Imagen 11	1,1989e + 09	1,1986e + 09
Imagen 12	1,2480e + 09	1,2508e + 09
Imagen 13	1,2648e + 09	1,2625e + 09
Imagen 14	1,0780e + 09	1,0807e + 09
Imagen 15	1,3713e + 09	1,3746e + 09

5.1.2. Conclusiones

Con los resultados obtenidos se ha demostrado que informaciones parásitas como el ruido en imágenes puede ser eliminado en un alto porcentaje con tal sólo el uso de pocos filtros de información los cuales se obtienen a nivel general de la matriz de mezcla \mathbf{A} y de la evaluación previa de las componentes disponibles en la matriz de las fuentes \mathbf{S} . Con estos pocos filtros de información pueden ser reconstruidos los datos originales sin importar que las observaciones estén con ruido.

La técnica ICA a demostrado presentar resultados que en general son bastantes buenos, ya que permiten una reducción notable del ruido, sobretodo de la clase multiplicativa; datos similares se han obtenido después de realizar la misma comparación con la técnica PCA, en donde puede apreciarse entre éstas, las pocas diferencias en los resultados obtenidos.

Conclusiones

A lo largo del desarrollo de este trabajo, se ha enfatizado sobre la idea de que un correcto conocimiento de un problema puede permitir aplicar correctas soluciones. Es sobre este pilar donde se ha basado la importancia del mismo. En esta tesis se ha tratado de evaluar la técnica ICA aplicado al análisis de procesamiento de imágenes con ruido. El objetivo de este análisis ha sido determinar el comportamiento de los algoritmos de fastICA, JADE, AMUSE, fastGEO y la nueva herramienta Denoising ICA (propriadamente con el algoritmo dss_fastICA), en relación a tres diferentes clases de ruido que son gaussiano, multiplicativo e impulsivo (sal y pimienta). Igualmente se ha desarrollado un método para la eliminación de ruido basado en el estudio de la matriz de mezcla \mathbf{A} y la matriz de fuentes \mathbf{S} de cada clase de ruido.

El análisis del comportamiento de los algoritmos con datos de ruido ha observado:

1. A nivel general el comportamiento de la mayoría de los algoritmos propriadamente fastICA, JADE, fastGEO y dss_fastICA, en referencia al problema con ruido de sal y pimienta se presenta con unos datos los

cuales a la hora de ser evaluados por medio del histograma, muestran un claro comportamiento de ruido impulsivo; pero al realizar para cada filtro el recorte de los puntos con mayor valor con el fin de poder observar que información contienen, se tiene que no existen datos que puedan mostrar información importante diferente de ruido. Cosa contraria ocurre con AMUSE donde no se presenta la típica distribución en la representación del histograma; el comportamiento de este algoritmo bajo la influencia de dichos datos sólo parece entregar información que en primera instancia puede pensarse en relacionarse como ruido gaussiano o multiplicativo.

2. Para algoritmos como fastICA y JADE la interpretación a vista de los datos, permite de manera rápida identificar que tipo de ruido se encuentra influyendo en las observaciones, mientras que en algoritmos como AMUSE, fastGEO y dss_fastICA, esta interpretación rápida los datos se hace más complicada y en algunos casos se hace necesario un análisis profundo.
3. La reducción de dimensiones llevada a cabo en AMUSE a permitido llegar hasta la selección (para cada tipo de ruido) de 9 filtros que contienen información libre de señales parásitas, independientemente que éstas estén contaminadas.

La reconstrucción de las imágenes originales tomando como base estos 9 filtros, junto con la reducción del porcentaje de ruido en las señales prove-

nientes de la matriz **S** entrega resultados bastantes satisfactorios en los que puede apreciarse a nivel general menos contaminación.

Realizada la comparación entre las dos técnicas al utilizar este nuevo método de eliminación de ruido puede verse que si bien existe una pequeña diferencia entre las mismas, es importante también resaltar que en general, bajo cualquiera de las dos técnicas de análisis, los resultados en cuanto a la forma en como se muestran las imágenes en bordes, esquinas, detalles, etc están muy bien conservados.

Pero esto es solamente una pequeña parte de lo que el campo del procesamiento de imágenes encierra. Trabajos futuros pretenden encaminarse en el análisis de imágenes a color e imágenes con otros tipos de estructuras.

Índice alfabético

Algoritmo

AMUSE, 88, 113, 146, 185

dss_fastICA, 114, 174

fastGEO, 90, 114, 162

fastICA, 83, 107, 124

JADE, 86, 110, 137

BSS (Blind Source Separation), 2

Entropía, 22

ICA (Independent Component Analysis), 3, 5, 6

Historia de, 79

y el ruido, 69

y las imágenes, 66

Ambigüedades de, 10

Blanqueamiento, 25

Hipótesis de, 7

Substracción de la media, 25

Matriz

de separación, 34

de fuentes, 6

de mezcla, 6

de observaciones, 7

Minimización de la información, 17

Momentos y cumulantes, 18

Negentropía, 22

Paquete

DSS, 91

PCA (Principal Component Analysis),
27

Ruido

gaussiano, 74, 122

impulsivo (sal y pimienta), 75, 124

multiplicativo, 74, 123

SVH

(Sistema Visual Humano), 50

conos y bastones, 55

la corteza visual, 60, 63

la retina, 52

las neuronas, 61

nervio óptico, 58

Teorema del límite central, 16

Valoración de la máxima, 18

Bibliografía

- [1] <http://www.cis.hut.fi/projects/ica/fastica/>.
- [2] L. B. Almeida. Faster training in nonlinear ica using misep. In *Proc. of the 4nd International Workshop on Independent Component Analysis ICA '2003*, pages 113–118, 2003.
- [3] M. R. Álvarez and C. G. Puntonet. A new geometry-based procedure for blind separation of sources. In *Proc. of the 2nd International ICSC Symposium on Engineering of Intelligent System EIS'2000*, 2000.
- [4] M. R. Álvarez, C. G. Puntonet, L. Parrilla, and A. Díaz. Redes neuronales aplicadas a separación ciega de señales. In *Actas del Seminario Anual de Automática, Electrónica e Instrumentación SAAEI'98*, 1998.
- [5] S. Amari and A. Cichocki and H. Yang. A new learning algorithm for blind signal separation. In *Advances in Neural Information Processing Systems*, volume 8, pages 757–763, 1996.
- [6] T. Asakura and E. Wolf. *Progress in optics*. 1995.

- [7] H. Attias. Independent factor analysis. In *Neural Computation*, volume 803-851, 1999.
- [8] H. B. Barlow. Sensory communication. possible principles underlying the transformation of sensory messages. In *MIT Press*, pages 217–234. MIT Press, 1961.
- [9] H. B. Barlow. Unsupervised learning. In *Neural Computation*, volume Vol. 1, pages 295–311, 1989.
- [10] H. B. Barlow. Redundancy reduction revisited. In *Network: Computation in Neural Systems*, volume Vol. 12, pages 241–253, 2001.
- [11] Y. Barness, J. Carlin, and M. Steinberger. Bootstrapping adaptive cross pol cancellers for satellite communications. In *Int. Conf. On Communication*, 1982.
- [12] C. Bauer. *Independent Component Analysis of Biomedical Signals. Modern Data Analysis Techniques Can Assist Medical Decision Making*. Tesis Doctoral. Universidad de Regensburg, 2001.
- [13] A. Bell and T.J. Sejnowski. Fast blind separation based on information theory. In *Proc. of International Symposium on non-linear theory and applications*, pages 43–47, 1995.
- [14] A. Bell and T.J. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. In *Neural Computation*, pages 1129–59, 1995.

- [15] A. Bell and T.J. Sejnowski. The independent component of natural scenes are adge filters. In *Vision Research*, pages 3327–3338, 1997.
- [16] A. Bovik. *Handbook of Image and Video Processing*. Academic Press, 2000.
- [17] S. Bozinoski and H. L. Nguyễn Thi. Séparation de sources a bande large dans un mélange convolutif. In *Proc. of Ecole des Techniques Avancées en Signal Image Parole*, pages 303–310, 1996.
- [18] Edited by R. Yuste and A. Konnerth. *Imaging in neuroscience and development: A laboratory manual*. Cold Spring Harbor, N.Y, 2005.
- [19] J. F. Cardoso and B. H. Laheld. Equivariant adaptive source separation. In *IEEE Trans. on Signal Processing*, pages 3017–3030, 1996.
- [20] J. F. Cardoso and A. Souloumiac. Blind beamforming for non-gaussian signals. In *IEE Proceedings*, pages 362–370, 1993.
- [21] A. Cichocki and S. Amari. *Adaptative Blind Signal nad Image Processing. Learning Algorithms and Applications*. Wiley, 2002.
- [22] A. Cichocki, W. kasprzak, and S. Amari. Multi-layer neural networks with a local adaptive learning rule for blind separation of source signals. In *Proc. of 1995 International Symposium on Non-linear Theory and Applications NOLTA'95*, pages 61–65, 1995.

- [23] A. Cichocki and R. Unbehauen. Robust neural networks with on-line learning for blind identification and blind separation of sources. In *IEEE Trans. on Circuits and Systems*, pages 894–906, 1996.
- [24] A. Cichocki, R. Unbehauen, and E. Rummert. Robust learning algorithm for blind separation of signals. In *Electronics Letters*, pages 1386–1387, 1194.
- [25] P. Comon. Independent component analysis - a new concept. In *Signal Processing*, pages 36:287–314, 1994.
- [26] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley and Sons, 1991.
- [27] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. In *Journal of the Royal Statistical Society*, volume Series B (Methodological) 39(1), pages 1–38, 1997.
- [28] M. Feng and K.D. Kammeyer. Application of source separation algorithms for mobile communication environment. In *Proc. of the International Workshop on Independent Component Analysis ICA'99*, pages 431–436, 1999.
- [29] G. Gelle, M. Colas, and G. Delaunay. Separation of convolutive mixtures of harmonic signals with a temporal approach. application to rotating

- machine monitoring. In *Proc. of the International Workshop on Independent Component Analysis ICA'99*, pages 109–114, 1999.
- [30] A. C. Guyton and J. E. Hall. *Textbook of medical physiology*. Saunders, 2000.
- [31] S. Handel. *Listening: An introduction to the Perception of Auditory Events*. MIT Press, 1989.
- [32] G. W. Hanson and A. B. Yakovlev. *Operator theory for electromagnetics: An introduction*. Springer, 2002.
- [33] S. Haykin. *Neural Networks*. Prentice Hall, 1999.
- [34] S. Haykin. *Unsupervised Adaptive Filtering*. Wiley, 2000.
- [35] W. H. Hayt and J. A. Buck. *Teoría Electromagnética*. McGraw-Hill, 2006.
- [36] J. Héroult, C. Jutten, and B. Ans. Detection de grandeurs primitives dans un message composite par une architecture de calcul neuromimétique en apprentissage non supervisé. In *X Colloque GRETSI*, pages 1017–1022, May 1985.
- [37] D. H. Hubel and T. N̄. Wiesel. Receptive fields, binocular interaction and functional architecture in cat's visual cortex. *Journal of Physiology*, pages 106–154, 1962.

- [38] D. H. Hubel and T. N̄. Wiesel. Receptive fields and functional architecture of monkey striate cortex. *Journal of Physiology*, pages 215–243, 1968.
- [39] D. H. Hubel and T. N̄. Wiesel. Receptive fields of cells in striate cortex of very young visually inexperienced kittens. *J. of neurophysiology*, pages 994–1002, 1972.
- [40] D. H. Hubel and T. N̄. Wiesel. Sequence regularity and geometry of orientation columns in the monkey striate cortex. *Journal of Comparative Neurology*, pages 267–293, 1974.
- [41] A. Hyvärinen. Sparse code shrinkage: Denoising of nongaussian data by maximum likelihood estimation. In *Neural Computation*, pages 1739–1768, 1999.
- [42] A. Hyvärinen, P. Hoyer, and E. Oja. Image denoising by sparse code shrinkage. In *Intelligent Signal Processing*, 2001.
- [43] A. Hyvärinen and E. Oja. A fast fixed-point algorithm for independent component analysis. In *Neural Computation*, pages 1483–1492, 1997.
- [44] Aapo Hyvärinen, P. Hoyer, and Erkki Oja. Denoising of nongaussian data by independent component analysis and sparse coding. In *Proc. of the International Workshop on Independent Component Analysis ICA'99*, pages 485–490, 1999.

- [45] Aapo Hyvärinen, Juha Karhunen, and Erkki Oja. *Independent Component Analysis*. Wiley Inter-science, 2001.
- [46] S. Ikeda. Ica on noise data: A factor analysis approach. In Springer-Verlag, editor, *In M. Girolami, editor, Advances in Independent Component Analysis*, pages 201–215, 2000.
- [47] M.C. Jones and R. Sibson. What is projection pursuit? *J. of the Royal Statistical Society*, pages 150:1–36, 1987.
- [48] C. Jutten. *Calcul neuro-mimétique et traitement du signal: Analyse en composantes indépendantes*. Tesis Doctoral. Universidad Joseph Fourier, 1987.
- [49] C. Jutten and J. Herault. Blind separation of sources, part i: An adaptive algorithm based on neuromimetic architecture. In *Signal Processing*, pages 24:1–10, 1991.
- [50] C. Jutten and J. Héroult. Independent component analysis versus principal component analysis. In *Signal processing IV, EUSIPCO-1988*, pages 643–646, 1998.
- [51] J. Karhunen and J. Joutsensalo. Representation and separation of signals using nonlinear pca type learning. In *Neural Networks*, pages 113–127, 1994.
- [52] Ingo R. Keck. *ICA incompleto Paralelo: Una nueva herramienta para el análisis de datos fMRI*. Tesis Doctoral. Universidad de Granada, 2006.

- [53] J. D. Kraus and D. A. Fleisch. *Electromagnetics with applications*. WCB-Mac Graw-Hill, 1999.
- [54] J. L. Lacoume, P. O. Amblard, and P. Comon. *Statistiques d'ordre supérieur pour le traitement du signal*. Masson, 1997.
- [55] T. W. Lee. *Independent Component Analysis*. Wiley, 2001.
- [56] M. Lastra Leidinger. *Métodos Estocásticos de síntesis de Imágenes para Entornos Complejos*. Tesis Doctoral. Universidad de Granada, 2004.
- [57] I. B. Levitan and L. K. Kaczmarek. *The neuron : cell and molecular biology*. Oxford University Press, 2002.
- [58] J. S. Lim. *Two-dimensional signal and image processing*. Prentice-Hall, 1996.
- [59] S. Makeig, T. P. Jung, A. J. Bell, and S. J. Sejnowski. Blind separation of auditory event-related brain response into independent components. In *Proc. of National Academy of Sciences*, volume 98, pages 10979–10984, 1998.
- [60] G. Pajares Martinsanz and J. M. De la Cruz García. *Visión por computador. Imágenes digitales y aplicaciones*. RA-MA Editorial Madrid, 2001.
- [61] S. Hornillo Mellado. *Sobre el análisis de componentes independientes de imágenes naturales*. Tesis Doctoral. Universidad de Sevilla, 2005.

- [62] E. Moulines, J. F. Cardoso, and S. Mayrargue. Subspace methods for blind identification of multichannel fir filters. In *IEEE Trans. on Signal Processing*, pages 43:516–525, 1995.
- [63] A. K. Nandi. *Blind Estimation using Higher-Order Statistics*. Kluwer Academic Publishers, 1999.
- [64] F. Pérez Ocón. *Discriminación cromática: Funciones de mezcla*. Tesis Doctoral. Universidad de Granada, 1996.
- [65] E. Oja. The nonlinear pca learning rule in independent component analysis. In *Neurocomputing*, pages 25–46, 1997.
- [66] B. A. Olshausen and D. J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *nature*. In *Vision Research*, pages 381:607–609, 1996.
- [67] B. A. Olshausen and D. J. Field. Sparse coding with an overcomplete basis set: A strategy employed by v1? In *Vision Research*, pages 37:3311–3325, 1997.
- [68] A. Papoulis. *Probability, Random Variables, and Stochastic Processes*, volume 3rd edition. McGraw-Hill, 1991.
- [69] D.-T. Pham, P. Garrat, and C. Jutten. Separation of a mixture of independent sources through a maximum likelihood approach. *Proc. EUSIPCO*, pages 771–774, 1992.

- [70] A. Prieto, C. G. Puntonet, B. Prieto, and M. R. Álvarez. A competitive neural network for blind separation of sources based on geometric properties. In Springer-Verlag, editor, *Lecture Notes in Computer Science*, volume 1240, 1997.
- [71] B. Prieto. *Nuevos Algoritmos de Separación de Fuentes usando Métodos Geométricos*. Tesis Doctoral. Universidad de Granada, 1999.
- [72] C. G. Puntonet. *Nuevos Algoritmos de Separación de Fuentes en Medios Lineales*. Tesis Doctoral. Universidad de Granada, 1994.
- [73] C. G. Puntonet, M. R. Álvarez, and B. Prieto A. Prieto. Separation of speech signals for nonlinear mixtures. In *Lecture Notes in Computer Science*, volume 1607, pages 665–673, 1999.
- [74] C. G. Puntonet, A. Mansour, and C. Jutten. Un algorithme géométrique pour la séparation de sources. In *Proc. of the 15^{ème} Colloque GRETSI*, 1995.
- [75] C. G. Puntonet, A. Prieto, C. Jutten, M. R. Álvarez, and J. Ortega. Separation of sources: a geometry-based procedure for reconstruction of n-valued signals. In *Signal Processing*, volume 46, pages 267–284, 1995.
- [76] W. De Raeye. *Implementation of biomedical Implementation of biomedical image processing algorithms*. France, 2003.
- [77] J. R. Reitz, F. J. Milford, and R. W. Christy. *Fundamentos de la teoría electromagnética*. Addison Wesley, 1999.

- [78] L. A. Remington. *Clinical anatomy of the visual system*. Elsevier, 2005.
- [79] H. Sahlin and H. Broman. Blind separation of images. In *Proc. of the 30th ASILOMAR*, 1996.
- [80] R. M. Rodríguez Sanchez and J. A. García Soria. *Representación de imágenes digitales basada en patrones visuales*. RA-MA Editorial, 2000.
- [81] J. Särelä and H. Valpola. Denoising source separation dss, 2005.
- [82] R. B. Schmidt and G. Thews. *Fisiología humana*. McGraw-Hill, 1993.
- [83] Gilbert Strang. *Introduction to Linear Algebra*. Wellesley-Cambridge Press, 2003.
- [84] F.J. Theis, A. Jung, and E.W. Lang. A theoretic model for linear geometric ica. In *preprint*, 2001.
- [85] F.J. Theis and E.W. Lang. Geometric overcomplete ica. *Proc. of the 10th European Symposium on Artificial Neural Networks (ESANN)*, pages 217–222, 2002.
- [86] F.J. Theis, E.W. Lang, T. Westenhuber, and C.G. Puntonet. Overcomplete bss wiht a geometric algorithm. *Proc. of International Conference on Artificial Neural Networks (ICANN)*, 2002.
- [87] N. Thirion. *Séparation d’ondes en prospection sismique*. Tesis Doctoral. Instituto Nacional Politécnico de Grenoble, 1995.

- [88] L. Tong, V.C. Soon, Y.F. Huang, and R. Liu. Amuse: a new blind identification algorithm. In *Circuits and Systems, IEEE International Symposium on*, volume vol.3, pages 1784–1787, 1990.
- [89] L. M. Marín Trechera, F. Álvarez González, and J. Ollero Hinojosa. Análisis de componentes independientes y análisis de componentes principales. relaciones y diferencias. *27 Congreso Nacional de Estadística e Investigación Operativa*, 2003.
- [90] E. Trucco and A. Verri. *Introductory Techniques for 3-D Computer Vision*. Prentice Hall, 1998.
- [91] J. S. Vhalupa. *The Visual Neurosciences*. Werner editors. MIT Press, 2003.
- [92] A. Vigneron, A. Paraschiv-Ionescu, O. Sibony, and C. Jutten. Fetal electrocardiogram extraction based on non-stationary ica and wavelet denoising. In *Proc. of the Seventh International Symposium on Signal Processing and its Applications (ISSPA2003), Paris France.*, 2003.
- [93] A. Westner and V. M. Bove. Separation of real world audio signals using overdetermined mixtures. In *Proc. of the International Workshop on Independent Component Analysis ICA'99*, pages 251–256, 1999.
- [94] A. Ypma and P. Pajunen. Rotating machine vibration analysis with second-order independent component analysis. In *Proc. of the Inter-*

national Workshop on Independent Component Analysis ICA '99, pages 37–42, 1999.

[95] Felipe Zaldivar. *Fundamentos de Algebra*. Mexico, 2006.

[96] S. Zeki. *Inner vision: an exploration of art and the brain*. Oxford University Press, 1999.