Universidad de Granada

E.T.S de Ingeniería Informáticas y de Telecomunicaciones

UNIVERSIDAD DE GRANADA

Departamento de Ciencias de la Computación e Inteligencia Artificial

Programa de Doctorado en Tecnologías de la Información y la Comunicación

Modelos de Aprendizaje Profundo para el Procesamiento y Clasificación de Imágenes y Vídeo

Deep Learning Models for Image and Video Processing and Classification

presented by
Santiago López Tapia

supervised by
Rafael Molina Soriano and Aggelos K. Katsaggelos

Editor: Universidad de Granada. Tesis Doctorales
Autor: Santiago López Tapia
ISBN: 978-84-1306-760-5
URI: http://hdl.handle.net/10481/66493

Agradecimientos (Acknowledgments)

Dedico este trabajo a todas aquellas personas que me han apoyado estos años. Sin vuestra ayuda su elaboración no hubiese sido posible.

Para comenzar, me gustaría mostrar mi más sincero agradecimiento a mis directores de tesis: Rafael Molina y Aggelos K. Katsaggelos, por haber confiado en mí para la elaboración de los trabajos recogidos en esta memoria y por todo el tiempo que han invertido en mi formación estos años. Así mismo, me gustaría agradecerles profundamente sus valiosos consejos. Sé que no suelo expresarlo, así que me gustaría que estas líneas sean testigo de cuanto los aprecio. Muchas gracias por todo, para mí ha sido un privilegio trabajar con vosotros.

Así mismo, agradezco a mi tutor, Nicolás Pérez de la Blanca, su inestimable ayuda y trabajo en la dirección de esta tesis. Jamás me hubiese planteado la realización de una tesis doctoral si no llega a ser por su apoyo e insistencia. Por toda su labor estos años, no me cabe ninguna duda de que he tenido la suerte de contar con no dos, sino tres directores. Ha sido un placer y un privilegio poder trabajar contigo.

También me gustaría agradecer, del mismo modo, a todos aquellos compañeros y compañeras que han colaborado en la elaboración de los trabajos que aquí se recogen: Alice Lucas, Javier Mateos, José Aneiros y Cristóbal Olivencia. Ha sido un honor trabajar con vosotros y espero que podamos seguir manteniendo una estrecha colaboración en el futuro.

No puedo olvidarme de agradecer a mi familia y amigos todo el apoyo que me han mostrado siempre y, en especial, durante el desarrollo de esta tesis. Sé que casi nunca os digo lo mucho que significáis para mí, pero espero que aun así sepáis lo mucho que os quiero a todos. Sin vosotros, no hubiese podido llegar jamás hasta aquí. Gracias por estar siempre ahí, en los buenos y en los malos momentos. Aprovecho para disculparme por todas aquellas veces en las que no he podido trataros como os merecéis. En especial, quiero disculparme con mi amigo Rafa, por haber abusado de nuestra amistad en más de una ocasión.

Por último, mostrar mi más profundo agradecimiento a todos los compañeros y compañeras con los que he tenido el placer de trabajar durante este periodo de mi vida. Muchas gracias por estar siempre dispuestos a ayudarme y a escucharme.

Contents

List of Abbreviations

BID Blind Image Deconvolution. 1–3, 5, 10, 11, 157, 158

CNN Convolutional Neural Network. 2, 3, 6–10, 12, 37, 47, 49, 50, 71, 138, 147, 157, 158, 160

CV Computer Vision. 1, 3

DL Deep Learning. 2, 3, 5, 10–12, 114, 138, 156, 159, 160

DLA Detection using a Local Approach. 114, 159

GAN Generative Adversarial Network. 3, 9, 10, 13, 47, 48, 157

GP Gaussian Process. 2

GPU Graphic Processing Unit. 8, 9, 158

H&E Hematoxylin and Eosin. 6, 7, 11, 12, 147, 160

HR High-Resolution. 1, 6, 13

LR Low-Resolution. 1, 4, 47, 71, 156, 158, 159

ML Machine Learning. 2, 6, 12, 94, 114, 159

NN Neural Network. 2, 8, 9

PHH3 Phospho-Histone H3. 11, 12, 147, 160

PMMWI Passive Millimeter Wave Image. 2–6, 9, 12, 94, 114, 156, 159

PSF Point Spread Function. 5, 158

PSNR Peak Signal-to-Noise Ratio. 14, 37, 50, 71, 157

ReLU Rectified Linear Unit. 8, 9

RNN Recurrent Neural Network. 2, 11

SGA Segmentation using a Global Approach. 114, 159

SGD Stochastic Gradient Descent. 8

SISR Single Image Super-Resolution. 71

SR Super-Resolution. 1–3, 5, 10, 12, 47, 48, 156–158

SSIM Structural Similarity Index Measure. 14, 37, 50, 71, 157

SVM Support Vector Machine. 2

TPU Tensor Processing Unit. 8, 9

VSR Video Super-Resolution. 10–14, 37, 47, 48, 156, 157

WSI Whole-Slide Image. 3, 6, 7, 11, 12, 138, 147, 156, 160, 161

Chapter 1

Introduction

Computer Vision (CV) systems focus on developing algorithms that process images and videos to obtain high-level information from them. In recent years, advances in CV systems have enabled the automation of tasks that were previously thought to be impossible to perform without human intervention. This is the case for fields as diverse as, for instance, security surveillance, medicine or robotics. Some examples of applications of CV systems in these fields are concealed object detection, cancer detection or self-driving cars (object detection and image segmentation). In some cases, these systems are capable of even outperforming humans[1]. The increasing number of sensors that capture different length-wave ranges in image type content has significantly increased these systems' demand. The following phases can be distinguished in CV: formation and interpretation of the image.

The image formation phase includes all the processes necessary to obtain an image or video with good perceptual quality from the sensors' information. This phase includes those techniques whose objective is related to eliminating noise and deformations that occur in the image. Some examples are exposure correction, color balance, reducing noise in the image, increasing the resolution or eliminating distortions in the image due to other factors such as movement. Of these operations, the most relevant ones are related to improving image quality in general and involve solving complex mathematical problems where the solution is not uniquely determined. These tasks' objective is moving from known events (the sensor responses) back to their most probable causes (the "real" image). Some examples are creating High-Resolution (HR) images from Low-Resolution (LR) images (Super-Resolution (SR)), removing blurring from images (Blind Image Deconvolution (BID)) or image noise removal. In this dissertation, we will focus on image and video SR and BID. For a more detailed explanation of these problems, see Section 1.1.

The second phase, image interpretation, consists of obtaining high-level semantic information from the image to compute a solution for the task at hand. Image classification, object detection and object recognition are currently three of the more relevant CV tasks associated with image interpretation. From these tasks, object detection [2] is considered a key stage to image interpretation. In this dissertation, we focus on two specific applications of object detection: threat detection using Passive Millimeter Wave Images (PMMWIs) [3, 4, 5] and mitosis detection [6, 7] in medical images [8, 9]. Both are relevant problems that have attracted much attention in recent years, with many factors that make them very challenging tasks. In the case of PMMWIs, the low quality of the images is the main factor (see Fig. 1.3). Meanwhile, mitosis detection's difficulty resides in the high variation of its characteristics depending on the capture process and tissue type.

Several approaches have been proposed to deal with the problems that we address in this dissertation. In the case of SR and BID, some of them involve solving an optimization problem on a family of possible solutions where the solution reaches a compromise between its variability and adjustment error with the data [10, 11, 12]. Meanwhile, most approaches proposed for threat detection in PMMWIs [3, 4, 5] and mitosis detection [13, 6, 14] usually involve the use of filtering techniques where a detection threshold has to be manually tuned. Despite the differences, all previous methods can be framed within what is called regularized optimization. Alternatively, instead of solving this optimization problem for each new sample, the function f that solves the task can be learned from data using supervised Machine Learning (ML). Given a set of N training examples $\{(x_1, y_1), ..., (x_N, y_N)\}$, such that x_i is the input data from the i-th sample and y_i the corresponding desired output, a ML algorithm selects from a family of functions a $f : X \to Y$, where X and Y are the input and output space, respectively, that minimizes a loss function $\mathcal{L}(f(x), y)$. Some examples of these algorithms are Logistic Regression, Support Vector Machine (SVM), Gaussian Process (GP) or Random Forest. These shallow ML algorithms are not usually employed on the raw signal because of its high dimensionality. Instead, they use features extracted from the signal. These features are hand-crafted, i.e., they have to be manually designed. Although very good feature extractors have been proposed, like Local Binary Patterns (LBP) [15], Haar [16], Fisher Vectors (FV) [17] or Vector of Locally Aggregated Descriptors (VLAD) [18], its performance is limited because the feature extraction can not be optimized with the loss function $\mathcal{L}(f(x), y)$. Kernel transformation in SVM or GP can palliate this issue by increasing the feature space dimensionality, but this is not enough in some applications.

In recent years, the use of Deep Learning (DL) has shown an astonishing increment in performance compared to shallow techniques. DL, also called hierarchical learning and deep structured learning, is "a class of machine learning techniques that exploit many layers of non-linear information processing for supervised or unsupervised feature extraction and transformation, and for pattern analysis and classification"[19]. Although the number of layers for a ML method to be considered deep is ambiguous, in general the following ML algorithms are considered part of DL [20]: Feed Forward Neural Networks (NNs), Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Boltzmann Machines (deep and restricted), Deep Belief Networks and Deep Gaussian

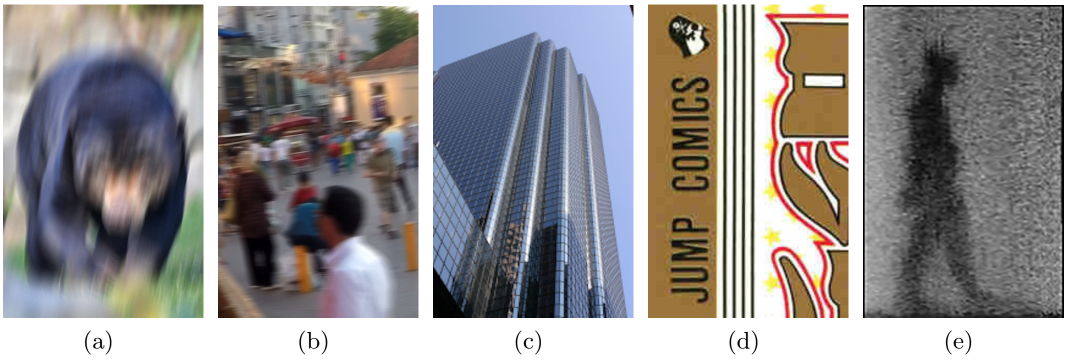
Processes [21]. From DL models, CNNs are the most successful for high level image and video related tasks like classification, segmentation, detection, parsing, etc., defining the state-of-art for many of them [22, 23, 24, 25, 26, 27]. Furthermore, they have also been applied to image and video processing tasks with similar success [28, 29, 30, 31, 32]. DL-based approaches can model complex functions with significantly less number of parameters than shallow ones [33] and they can learn feature extraction from raw signals, finding the features that minimize the loss function. However, DL models require orders of magnitude more training data than shallow ones, which is why they have not been widely used until recently, despite the first models being introduced more than 20 years ago [34].

Motivated by the success of DL-based models in image and video problems, in this dissertation, we develop DL-models for challenging image and video formation and interpretation tasks. These are image and video SR, BID, threat detection in PMMWIs and mitosis detection in Whole-Slide Images (WSIs). In this thesis, one common point to all contributions is the use of domain knowledge to improve the solution by developing and applying specialized architectures, regularizations and restrictions. In the next subsections, we present the tasks that we have addressed in this dissertation. Next, we provide a brief introduction to the main DL-based models used in this dissertation: CNNs and Generative Adversarial Networks (GANs). Finally, we present the objectives of this work and the structure of the remainder of this dissertation.

1.1 Image and Video Processing: Restoration and Super-Resolution

The number of pictures and videos taken has increased dramatically in the last years. There are two main factors behind this massive increase in image and video content: the rise in social media popularity and smartphones. They have compelled most of the population to share pieces of their personal life using visual content. Each day 95M and 350M pictures are uploaded to Instagram and Facebook, respectively. Moreover, each minute more than 300 video hours are shared through YouTube. However, these are not the only sources of new image content. The advances of CV have allowed automatic algorithms in new tasks involving images and videos: traffic control, hidden threat detection in airports, self-driving cars or cancer detection. Furthermore, the introduction of DL-based models has further improved their performance. However, these methods' performance deteriorates significantly if the content presents degradations or artifacts (noise, blur or compression artifacts) [35, 36], requiring the use of image processing techniques.

Despite the general quality leap that image and video capturing devices have undergone in the last years, the need for image and video processing techniques is still present. There are several reasons for this fact. First, an essential part of the degradations that



(a) (b) (c) (d) (e)

Figure 1.1: Examples of degradations that can affect images and videos: (a) Uniform blur, (b) Non-uniform blur, (c) LR, (d) Compression artifacts or (e) Noise and LR.

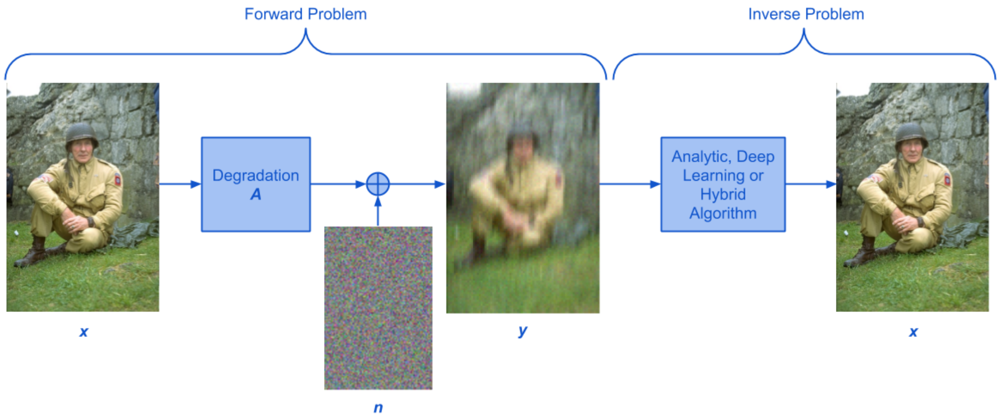


Figure 1.2: Illustration of the forward problem (1.1) and inverse problem. First, the degradation operator A is applied to the latent image x and noise n is added, obtaining the observed image y. The objective of the inverse problem is obtaining an estimation of the latent image. For illustration purposes, we use as the degradation operator A blur and downsampling with scale factor 8.

image and video content can exhibit is not due to the capturing device but rather to the conditions in which the scene is taken: movement or shaking of the camera (see Fig 1.1a) or object movement in the scene (see Fig 1.1b). Second, old image and video content were taken in a much lower resolution than today's standard. Its resolution needs to be increased to property show it in modern high-definition displays, or it will look too blurred (see Fig.1.1c). Finally, in some applications, there are physical and economic factors that limit the quality of the image or video captured, such as the compression artifacts that appear due to the high compression needed to transmit through specific channels (see Fig 1.1d) or the non-stationary noise and LR of PMMWIs (see Fig 1.1e).

In most cases, we can model the degraded observed signal (image or video) y as the result of applying a transformation A, the degradation operator, to the latent signal x

and adding noise n. This is the forward problem and can be written as

$$y = Ax + n. \tag{1.1}$$

The kind of problem to solve depends on A and n: Deconvolution or image restoration (A is a matrix performing a convolution operation at each point of x with a Point Spread Function (PSF), which is unknown in the case of BID), SR (A is a combination of a blur operator B and downsampling $\downarrow_s$ by factor s) or Denoising ($A = I$ and $n \neq 0$). The objective of most image processing techniques is to solve the inverse problem and recover x, see Fig 1.2. The difficulty of solving this inverse problem stems from the properties of A and n. These usually determine the system to be ill-posed in the Hadamard sense [37]; that is, small variations in the input data result in large variations in the solution. The presence of noise also makes it so there exist infinite possible solutions of x compatible with the observed y, even when A is known. The information loss in SR or the need to estimate A in BID only exacerbate this issue. Despite the difficulty, several algorithms have been proposed to solve these problems, both classical [10, 11, 12, 38, 39, 40, 41, 42] and DL-based [28, 29, 43, 44, 45, 30, 46, 47, 31, 48, 49, 50, 32, 51], which demonstrates their importance.

1.2 Threat Detection in Passive Millimeter-Wave Images

Millimeter waves are electromagnetic radiations in the 0.001-0.01m wavelength range (30-300GHz frequency range). Objects naturally emit them because of their heat. Millimeter-waves can go through clothes revealing concealed objects behind them. Because of this fact, they are useful in hidden object detection and have been integrated as part of theft and threat detection systems [52] in places such as airports, train and metro stations and warehouses. We can distinguish two types of millimeter-wave sensors: active, which direct waves to the subject and collect the reflected energy; and passive, which collect the radiation emitted and reflected by the objects in the scene (see Fig. 1.3 for an example of the produced images). In contrast to active millimeter systems and other alternatives, like back-scatter X-ray, passive millimeter systems fully respect the user's privacy. See [53] for a study of health and privacy issues of these systems.

Despite their advantages, PMMWIs suffer from several issues that make the task of threat detection challenging:

- Low signal to noise ratio.
- Low resolution. Although it can be increased by using a higher sampling rate, this will decrease the low signal to noise ratio even further.
- In-homogeneous signal intensity is caused by the differences in temperature between the objects in the scene.

Because of these issues, current detection algorithms have a high false-positive ratio

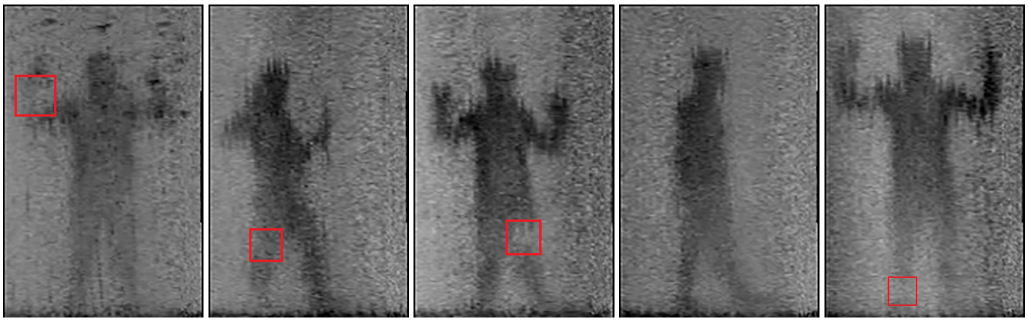


Figure 1.3: Examples of PMMWIs used for threat detection. The threat is indicated with a red box.

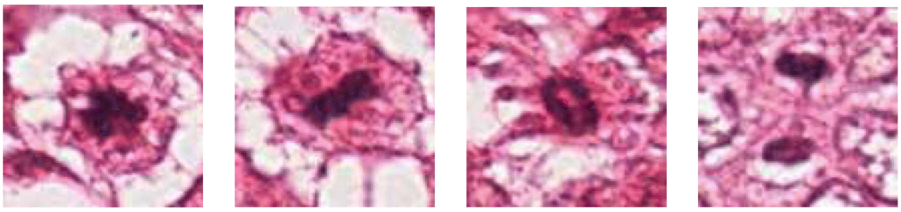
when avoiding false negatives. This harms the real-world application of these techniques. However, PMMWIs can still be used to distinguish between the body and hidden objects, as shown in [54].

Although sensor modeling, image processing and clustering techniques have been used on PMMWI threat detection problems [55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66], no ML-based detection solutions have been proposed yet, likely because of the absence of a large database of PMMWIs to work with. If such database is available, we believe that CNN models would be a good fit for this task, since they have been demonstrated to outperform other models in detection and segmentation tasks [24, 26, 25] as well as noise removal [67].

1.3 Mitosis Detection in Whole-Slide Images

In recent years, the field of digital pathology (a pathology sub-field that focuses on using information generated from digitized specimen slides) has seen a growth in research interest by the ML community, leading to significant achievements in key areas like automatic cancer detection. These achievements have been facilitated by the availability of public and private datasets of WSIs. A WSI is a scan of a complete microscope slide into a single HR image file. This process involves scanning several image tiles or strips and combining them, which may cause some artifacts. For ease of visualization and processing, the image is scanned at different resolutions reaching 0.35-0.16 microns per pixel, obtaining images of the order of 10^{10} pixels. Before scanning, the tissue must be stained. The most widely used stains in medical diagnosis [68] are Hematoxylin and Eosin (H&E). This is a combination of two stains: Hematoxylin is adsorbed by cell nuclei and has a blue color, while Eosin stains the extracellular matrix and cytoplasm in pink. Other structures in the tissue are stained with mixed colors. Notice that other stains can be used depending on the application.

One of the most relevant tasks in WSIs is mitosis detection. The quantity of mitosis



(a) Prophase (b) Metaphase (c) Anaphase (d) Telophase

Figure 1.4: Examples of mitosis in H&E WSIs for each phase of the mitosis process. Images were taken from [69]. Notice the drastic changes in shape and appearance between phases.

per mm^2 is one of the most important factors in the prognosis of several cancer types. Mitosis is the process in which one cell divides itself into two identical cells and has four distinct phases: Prophase, metaphase, anaphase and telophase. Cells undergoing this process are called mitotic figures. Some examples of mitotic figures are shown in Fig. 1.4. Notice the geometric and color variation despite being from the same tissue type. As can be seen, the manual quantification of mitosis is a very time-consuming task since the WSI has to be analyzed at a large magnification. Furthermore, several factors make it very difficult:

- The shapes and appearances of the mitosis change drastically along its four stages: prophase, metaphase, anaphase and telophase (see Fig 1.4).

- Color variability of the WSIs caused by stain intensities and differences between scanners makes it very difficult to transfer the knowledge learned. This issue is alleviated thanks to the introduction of techniques for color deconvolution and normalization [70, 71, 72].

- Further geometric deformations due to changes between different tissue types exacerbate the difficulty of transferring the knowledge learned in one kind of tissue to another.

- H&E indirectly helps mitosis identification by staining the mitotic cell nucleus intensely. However, other tissue parts are also stained similarly.

The first approaches for mitosis detection in digital images date back more than two decades [13, 6, 14]. These methods were significantly limited by their day's computer power and the lack of availability of WSI datasets. They rely on image processing techniques to detect mitosis. Although they are very fast, they suffer from low accuracy. Advancements in WSI scanning technology and CNNs have reignited interest in this problem. This has manifested in the appearance of several challenges, such as MITOS ICPR-2012 [73], MITOS-ATYPIA [74] and TUPAC-2016[75]. These challenges have provided new datasets that have been used by new CNN-based approaches [7, 76, 77, 78, 79], significantly outperforming other methods.

Having described the problems we address in this dissertation, we now provide a brief introduction to the main tools we use to solve them.

1.4 Brief Introduction to Convolutional Neural Networks and Generative Adversarial Networks

CNNs were first introduced by LeCun et al [34] in 1998. They are a class of feed-forward NNs designed to exploit the spatial dependencies among neighboring pixels in images. A cascade of previously learned convolutional filters and poolings summarizes the image in a vector of features representing information in large image regions. Local spatial weights and shifting invariance are the two main properties of the filter masks. The most common operations in a CNN, are described below:

- Convolution: This is the basic operation in a CNN. The input is convolved with a bank of learned filters. Since this is a linear transformation, after the convolution, a non-linear pointwise function is applied. Some examples of such operations are the Sigmoid or the Rectified Linear Unit (ReLU)[80]. This gives the CNN the capacity to learn non-linear transformations of the input data.

- Pooling: It reduces the input's spatial size by replacing adjacent pixels regions with a statistics summary. The most common operations are max pooling and average pooling and report the maximum and average output within a rectangular neighborhood, respectively.

- Classifier: Usually a normal NN (often called full connections or fully connected layers) or logistic regression. It performs the final classification using the features extracted from the signal using the other two operations.

Fig. 1.5 shows an example of the architecture of an image classification CNN where each one of the previously mentioned types of layers is present. The CNNs are trained like normal NNs, using back-propagation [81] and Stochastic Gradient Descent (SGD) on mini-batches of samples to minimize a loss function. By using a mini-batch size greater than one, the training can be easily accelerated by taking advantage of the massive parallel computation units like a Graphic Processing Unit (GPU) or a Tensor Processing Unit (TPU). Finding a good minimum using SGD is not a trivial task, requiring tuning of the hyper-parameters, of which the learning-rate is the most relevant. Recently, several methods have been proposed to ease this task by automatically the learning-rate for each layer in the model [82, 83, 84]. These methods, in conjecture with better initialization schemes [85, 1], have significantly eased the training of CNNs. However, hyper-parameter tuning is still necessary to properly train these models.

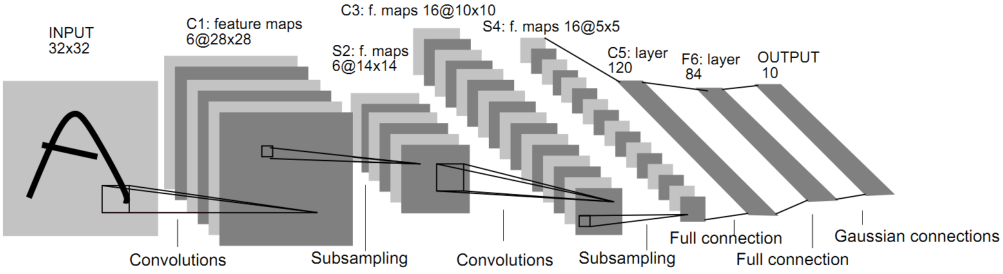


Figure 1.5: Architecture of CNN LeNet-5 [34]. This image has been taken from the same publication.

The first proposed model in 1998, LeNet[34], was pretty close to current CNNs. However, two major issues kept them from reaching today's wide-spread usage: First, the computing power in the early 2000s was not enough to train models more complex than LeNet[34]. Second, these networks' depth was seriously limited by the vanishing gradient problem [86]. In recent years, these issues have been solved with the use of massive parallel computation units (GPUs and TPUs) and the substitution of Sigmoid and Tanh activations with ReLU [80] and its variants [1, 87]. Specialized architectures have been develop using task-specific knowledge, adapting these networks to different problems like object detection [26], segmentation [88, 25], text and speech classification [89, 90] or image processing [28, 46, 31]. Most relevant works in recent years have proposed architectures compose of computing blocks. These blocks are an aggregation of layers that perform specific computations [24, 91]. One of the most important of these blocks is the residual blocks [24]. Each of them calculates a residue added to the block's input, forcing the network to transform the input progressively. As a result of these constraints, deeper networks can be trained without the overfitting or the appearance of the vanishing gradient problem. These networks have become the backbone of several detections and image restoration approaches [92, 26, 93, 43, 32]. In these approaches, the network's basic structure is retained and used to perform specific parts of the whole calculation. It is important to notice that state-of-art [92, 26, 93] detection methods cannot be directly applied to the detection problems studied in this thesis. In the case of mitosis detection, the images are too big to be processed by the model. Although PMMWIs are small enough to be processed by these models, they lack the texture information needed by these models [94] and, at the same time, the models are too big to be trained using the small datasets available.

CNNs are discriminate models that learn decision boundaries from observed data. In contrast, a generative model is capable of learning the distribution of the data. In recent years, several generative models have been proposed adapting the technology of deep NNs and CNNs [95, 96]. Of these, GANs [96] are the most widely used. They consist of two modules, generator and discriminator, each one defined by a network. Given a data distribution, their objective is to train the generator network to produce samples with the same statistics as those from the distribution. To do this, the generator is trained to fool the discriminator, whose task is to distinguish between real and generated samples, that is,

$$\min_G \max_D \mathbb{E}_{x\sim P_{\text{data}}}[\log D(x)] + \mathbb{E}_{z\sim\text{noise}}[\log(1 - D(G(z)))]\,, \tag{1.2}$$

where G is the generator, D is the discriminator, x is a sample from the data distribution P_{data} and z is a sample from a noise distribution. Research in GANs has been quite active, being applied successfully to different image and video tasks [43, 97, 98, 99, 100, 44, 101] and with new GAN models being proposed to solve some of GANs main issues: their instability and mode collapse [102, 103, 104].

From previous works in CNNs and CNN-based GANs, it can be derived that one of the critical factors that determine the performance of a CNN is the architecture. Indeed, as shown in [105, 106], CNN architectures introduce deep priors, which for many applications, explain a large portion of the model performance. However, some image and video problems present more challenging scenarios where the prior imposed by the architecture prior is not enough to obtain satisfactory results. For instance, this is the case when datasets are too small to regularize the model or the surface of the solution shows many local minima. In those cases, using knowledge of the problem to introduce further regularization or constraints in the function space can significantly increase the model's performance and reduce overfitting. Some mechanisms that have been used to this end are the introduction of a Gaussian prior in the weight space using Dropout [107], increasing the invariance of the network to geometric transformations [108], specializing parts of the network [109] or combining the DL method with other approaches [110]. Following these works, when applying DL-based models in this dissertation, we will focus on using our knowledge of the problem to develop and apply specialized architectures, regularizations and restrictions to increase the performance.

In this sense, we have analyzed previous DL-based models proposed for image and video SR, BID and mitosis detection, identifying some issues that we think can be addressed with the introduction of domain knowledge using some of the previously mentioned approaches:

- In image and video SR: First, most SR models are trained to invert only one downsampling operator (usually bicubic downsampling) and do not generalize well to new ones. This hurts their real-world application. Second, most models use the Mean Squared Error (MSE) as the loss function: $\mathcal{L}(\hat{x}, x) = \sum_i \sum_j (\hat{x}_{i,j} - x_{i,j})^2$, which is an empirical risk minimization approach. As a result, the estimated images and videos are blurred. The use of GANs and losses more correlated with the human vision has been proposed [43] to increase the estimations' sharpness. However, current approaches tend to introduce easily detectable high-frequency artifacts or do not apply to scaling factors bigger than two [44]. Finally, Video

Super-Resolution (VSR) models do not exploit long term information effectively, ignoring the recent developments in RNN.

- In BID, current DL approaches do not generalize well to unseen blurs during training. To solve this issue, hybrid models combining analytical and DL approaches are very effective, outperforming other approaches and generalizing to new degradations. However, because this combination is carried on introducing the DL model in an iterative optimization process, they are very time-consuming.
- Current DL-based approaches for mitosis detection [7, 76, 77, 78, 79] suffer from a high computational cost because of the size of the images and the fact that they exploit the multiple-scale structure of a WSI. Furthermore, they cannot generalize to new tissue types, requiring full training of the model with a new database [111]. Finally, current approaches have not explored the possibility of using other staining methods in conjecture with H&E. More specifically, the immunohistochemistry Phospho-Histone H3 (PHH3) [112] is a staining method used for mitosis detection with less false negatives than H&E for many types of cancer [113, 114].

Having presented the problems addressed and the tools used in this dissertation, we now introduce the objectives.

1.5 Objectives and Outline of the Thesis

This work's main goal is to develop new DL-based models and study their application to image and video processing and classification. In other words, we develop new DL-based models to solve the problems described in Sections 1.1-1.3. Because of this, our main objective can be broken down into sub-objectives grouped in four blocks: video super-resolution, image restoration and super-resolution, threat detection in passive millimeter-wave images and mitosis detection in whole-slide images. We now specify the sub-objectives of each block:

1. **Video super-resolution**:
   - To introduce new DL-based methods for VSR robust to multiple degradations.
   - To enhance the perceptual quality of the frames produced by DL-based models for VSR.
   - To improve current DL-based approximations for VSR by improving information propagation through different time steps.
2. **Image restoration and super-resolution**:
   - To develop a new fast and accurate DL-based algorithm for BID that is able to generalize to unseen blurs during training.

- To implement a new fast and accurate DL-based approximation that can restore an image degraded by noise, blur and downsampled.

3. **Threat detection in passive millimeter-wave images**:

   - To construct a new database of PMMWIs with significantly more instances, people and objects than the ones available. This database must be large enough to train and evaluate ML-based approaches.
   - To implement an initial ML-based approximation to threat detection on PMMWIs using shallow models. This approximation will be used as a baseline.
   - To develop a DL-based model for threat detection in PMMWIs and compare it to ML-based approaches.

4. **Mitosis detection in whole-slide images**:

   - To implement a fast DL-based model for mitosis detection in H&E stained WSIs of basocelular cancer tissue that is able to generalize to other tissue types.
   - To study the combined use of H&E and PHH3 stained tissue WSIs for mitosis detection using CNNs.

This dissertation is presented in the modality of “compendium”. The specific publications that compose it are structured in four blocks (video super-resolution, image restoration and super-resolution, threat detection in passive millimeter-wave images and mitosis detection in whole-slide images), following the same organization as the objectives. The main contributions are highlighted before each publication and the conclusions are presented together in Chapter 6. The structure of the remainder of this dissertation is as follows:

- **Chapter 2**: Presents our contributions on the topic of VSR [115, 116, 117, 118]. In Section 2.3, we present relevant works in the field where the Ph.D. candidate was not the main author but had a significant role.
- **Chapter 3**: Contributions on image restoration [119] and SR [120] are presented in this chapter.
- **Chapter 4**: This chapter presents our work on threat detection in PMMWIs [121, 122].
- **Chapter 5**: Our contributions on mitosis detection on WSIs [123, 124] are introduced in this chapter.
- **Chapter 6**: Conclusions and future work.

Chapter 2

Video Super-Resolution

2.1 A Single Video Super-Resolution GAN for Multiple Downsampling Operators based on Pseudo-Inverse Image Formation Models

2.1.1 Publication details

Authors: Santiago López-Tapia, Alice Lucas, Rafael Molina, and Aggelos K. Katsaggelos.
Title: A Single Video Super-Resolution GAN for Multiple Downsampling Operators based on Pseudo-Inverse Image Formation Models.
Publication: Digital Signal Processing, vol. 104, 102801, 2020.
Status: Published.
Quality indices:

- Impact Factor (JCR 2019): 2.871
- Rank: 102/266 (Q2) in Engineering, Electrical and Electronic

2.1.2 Main Contributions

- First, we introduce a new VSR architecture that adapts the approximation introduced in [44] to the multiple degradation VSR problem. This model uses the pseudo-inverse of the degradation to regularize the recovery of the HR frames and as an input to the network. As shown by our experiments, this enables our model to be significantly more robust to multiple degradations than current approaches.
- Second, to further increase the sharpness of the super-resolved frames, we introduce a new loss function that combines adversarial GAN loss and feature loss with a spatial smoothness constraint. This constraint forces the model to avoid introducing high-frequency artifacts in low-activity areas of the image, which are easily spotted by the human eye. This new loss enforces smoothness only in areas of the image with low activity.

- Third, we create a new training dataset for VSR, created from a subset of videos from the YouTube-8M dataset [125]. Compared to most used training datasets in VSR, like Myanmar [126], it has more diverse scenes and motions.

- Finally, we perform an extensive experimental study and compare with current state-of-art methods in terms of Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM), Perceptual Distance[127] and number of operations.

Preliminary results of this work were presented in two conference papers: The first one is [115] (GGS Rating: A-, GGS Class: 2, CORE: B). It introduces a different architecture not adapted to multiple degradations and an initial version of the proposed smoothness constraint imposed on the whole image. [116] (GGS Rating: B, GGS Class: 3, CORE: B) is the second publication. It extends the architecture proposed in [115] to multiple degradations. In [117], the architecture is improved and extended to multiple scaling factors.

A Single Video Super-Resolution GAN for Multiple Downsampling Operators based on Pseudo-Inverse Image Formation Models

Santiago López-Tapia *, Alice Lucas†, Rafael Molina*, Aggelos K. Katsaggelos†
**Dept. of Computer Science and Artificial Intelligence, University of Granada,*
Granada, Spain
Email: {sltapia, rms}@decsai.ugr.es
†*Dept. of Electrical Engineering and Computer Science Northwestern University*
Evanston, IL, USA
Email: alicelucas2015@u.northwestern.edu, aggk@eecs.northwestern.edu

Abstract

The popularity of high and ultra-high definition displays has led to the need for methods to improve the quality of videos already obtained at much lower resolutions. A large amount of current CNN-based Video Super-Resolution methods are designed and trained to handle a specific degradation operator (e.g., bicubic downsampling) and are not robust to mismatch between training and testing degradation models. This causes their performance to deteriorate in real-life applications. Furthermore, many of them use the Mean-Squared-Error as the only loss during learning, causing the resulting images to be too smooth. In this work we propose a new Convolutional Neural Network for video super resolution which is robust to multiple degradation models. During training, which is performed on a large dataset of scenes with slow and fast motions, it uses the pseudo-inverse image formation model as part of the network architecture in conjunction with perceptual losses and a smoothness constraint that eliminates the artifacts originating from these perceptual losses. The experimental validation shows that our approach outperforms current state-of-the-art methods and is robust to multiple degradations.

Index Terms

Video, Super-resolution, Convolutional Neuronal Networks, Generative Adversarial Networks, Perceptual Loss Functions

I. Introduction

The task of Super-Resolution (SR) consists of obtaining High-Resolution (HR) images from the corresponding Low-Resolution (LR) ones. This task has become one of the main problems in image and video processing because of the increasing demand for such methods from the industry. Due to the growing popularity of high-definition display devices, such as High-definition television (HDTV) and Ultra-high-definition television (UHDTV), there is an avid demand for HR videos. However, most of the content (especially, older videos) has been obtained at much lower resolution. Therefore, there is a high demand for methods able to transfer LR videos into HR ones so that they can be displayed on HR TV screens, void of artifacts and noise.

In the problem of image Super-Resolution (SR), the high-to-low image formation model can be written as:

$$y = D(x \otimes k) + \epsilon, \tag{1}$$

where y is the LR image, x is the HR image, ϵ is the noise, $x \otimes k$ represents the convolution of x with the blur kernel k and D is a downsampling operator (usually chosen to be bicubic downsampling). In the case of Video Super-Resolution (VSR), y, x, and ϵ are indexed by a time index t and additionally $\mathbf{y}_t$ is used to refer to the $2l+1$ LR frames in a time window around the HR center frame x_t, that is, $\mathbf{y}_t = \{y_{t-l}, \ldots, y_t, \ldots, y_{t+l}\}$. Due to the strongly ill-posed nature of the SR problem, the recovery of the original HR image or video sequence is a difficult task.

Current SR methods can be divided into two broad categories: model-based and learning-based approaches. Model-based approaches explicitly define and use the degradation process described by Eq. 1 by which LR image is obtained from the HR image or video sequence, see the reviews [1], [2]. With this explicit modeling, an inverse problem is solved to obtain an estimate of the reconstructed HR frame. These methods rely on careful regularization to deal with the ill-posedness of the problem. To enforce image-specific features into the estimated HR, signal priors are used, such as those controlling the smoothness or the total variation of the reconstructed image. In the case of the work presented in [3] a new multichannel image prior model is used in conjunction with a state-of-the art image prior and observation models to produce the SR image using a MAP algorithm. In [4], an SR image is obtained from the LR observations through the simultaneous estimation of the SR and the motion between the LR observations using the Bayesian framework.

On the other hand, learning-based approaches do not explicitly make use of the image formation model and use instead a large training database of HR and LR image/sequence pairs to learn the solution to the SR problem, i.e. they learn a mapping between the LR observations and the HR estimation [5], [6]. Classic learning-based models mainly focused on how to build a dictionary or manifold space to relate LR and HR images and determine what representation schemes could be used in such spaces [5], [7], [8], [9]. Recently, methods based on Convolutional Neural Networks (CNNs) have been proposed for image SR and VSR, typically outperforming classic learning-based and model-based methods. These methods try to find a function $\mathrm{f}(\cdot)$ such that $x = \mathrm{f}(y)$ (or $x = \mathrm{f}(\mathbf{y})$ for VSR), which solves the mapping from LR images (or video sequences) to HR ones. The first use of these models for image SR was proposed in [6], where a three layered CNN was used to recover an HR image from its bicubically upsampled LR observation. Following works improved the architecture of the network by introducing layers that allow the network to learn the upsample operator [10] or increase the depth of the network through the use of residual blocks [11].

Although CNN-based models typically outperform model-based methods, most of them are not as flexible as model-based methods, in the sense that they are trained for a specific type of degradation operator. More specifically, an artificially synthesized dataset with LR and HR pairs is generated using only one degradation operator A (usually bicubic downsampling) for training. In addition, the LR sequences used for testing are assumed to have been subjected to the same degradation. This limits the trained model to only one type of degradation and its performance greatly deteriorates when a mismatch between training and testing degradation models occurs (see [12], [13]). This significantly limits their practical application. Recently, some works have proposed SR models that address this issue [12], [13] (see Section II).

An additional problem with most CNN-based approaches to SR and VSR problems is that they are trained using the Mean-Squared-Error (MSE) cost function between the estimated and original HR frames. Numerous works in the literature (e.g., [14], [11], [15], [16]) have shown that while the MSE-based approach provides reasonable SR solutions, its conservative nature does not fully exploit the potential of Deep Neural Networks (DNNs) and produces blurry images. As an alternative to the MSE cost function, recent CNN-based SR methods use (during training) features learned by pre-trained discriminative networks and compute the MSE between estimated and ground truth HR features. Using such feature-based and pixel-based losses has proven to boost the quality of the super-resolved images [17]. Unfortunately, this approach tends to introduce high-frequency artifacts, as was shown in [17].

The use of Generative Adversarial Networks (GANs) [18] was also proposed as a mechanism to increase the perceptual quality of the estimated images trying to avoid again the smoothing introduced by the MSE loss (e.g., [19], [11], [20]). GANs consist of two networks: a generator network that produces the SR image and a discriminator one that distinguishes between generated images and real ones. These networks can therefore constraint the generated HR images to satisfy the distribution of the real HR images. The produced images are sharper because blurry images do not belong to the distribution of HR images. In the case of SR and VSR, these models are trained incorporating additional terms to the loss function of the generator network (see [11], [15], [16], [17]).

In this work, we propose a new GAN model that adapts the approximation proposed in [21] to Multiple-Degradation Video Super-Resolution (MDVSR). It uses the pseudo-inverse image formation model not only in the image formation model (as proposed in [21]), but also as an input to the network. Our experiments show that this model trained with the MSE loss significantly outperforms current state-of-the-art methods for bicubic degradation in terms of PSNR and SSIM metrics and it is significantly more robust to multiple degradations than current approaches. To further increase the sharpness of the resulting frames, we propose the use of a new loss function that combines adversarial GAN loss and feature loss with a spatial smoothness constraint. This new loss allows for a significant increase in the perceptual quality of the estimated frames without producing the high-frequency artifacts typically observed with the use of GANs.

For all described models, the use of an appropriate dataset is of paramount importance. While many of the current VSR learning-based models are trained using the Myanmar dataset, this dataset has limited variation of both scene types and motions. In this work we show that GAN-based VSR models significantly benefit from training with a dataset with more diverse scenes and motions. We obtain a significant increase in perceptual quality by training our best performing model on a dataset created from a subset of videos from the YouTube-8M dataset [22].

The rest of the paper is organized as follows. We provide a brief review of the current literature for learning-based VSR in Section II. In Section III, we present our baseline VSR model. In this section we detail the architecture used for our model. By additionally introducing our new spatial smoothness loss we obtain our proposed model trained to maximized perceptual quality. The training procedure, the new dataset, and experiments are described in detail in Section IV. In this section we also evaluate the performance of the proposed models by comparing them with current state-of-the-art VSR approaches for scale factors 2, 3, 4, 8 and different degradations. Our quantitative and qualitative results show that our proposed perceptual model sharpens the frames to a much greater extent than current VSR state-of-the-art DNNs without the introduction of artifacts. In addition, we show in this section that our resulting model is more robust to variations in the degradation model compared with the current state-of-the-art model. Finally, conclusions are drawn in Section V.

II. Related work

In recent years, different VSR CNN-based models have been proposed in the literature. Liao et al. [23] utilize a two-step procedure where an ensemble of SR solutions is first obtained through the use of an analytic approach. This ensemble then becomes the input to a CNN that calculates the final SR solution. Kappeler et al. [24] use a three layer CNN to learn a direct mapping between the bicubically interpolated and motion compensated LR frames in $\mathbf{y}_t$ and the corresponding HR central frame x_t. Other works have applied Recurrent Neural Networks (RNNs) to VSR. For example, in [25] the authors use a bidirectional RNN to learn from past and future frames in the input LR sequence. Although RNNs have the advantage of exploiting more effectively the temporal dependencies between frames, the challenges and difficulties associated with their training has led to CNN being the favored DNN for VSR. Li and Wang [26] exploit the benefits of residual learning with CNNs in VSR by predicting only the residuals between the high-frequency and low-frequency frame. Caballero et al. [27] jointly train a spatial transformer network and a CNN to warp video frames, so that they benefit from sub-pixel information and they avoid the use of motion compensation (MC). Similarly, Makansi et al. [28] and Tao et al. [29] found that jointly performing upsampling and MC increases the performance of the VSR model.

All these previous methods use the MSE loss as the cost function during the training phase. This is the most common practice in the literature for CNN-based models. However, the use of this loss during training causes the estimated frames to be blurred. In an attempt to solve this problem, recent works have used feature-based losses as additional cost functions, see [14]. This approach has significantly improved the sharpness and the perceptual quality of the estimations. Ledig et al. [11] proposed a combination of a GAN and feature loss for training, leading to the generation of images with superior photorealistic quality. In [17], Lucas et al.

proposed an adaptation of this approach to VSR. They introduced a new loss based on a combination of perceptual features and the use of the GAN formulation. The model has led to a new state-of-the-art VSR approach in terms of perceptual quality. To improve the quality of the predicted images, Wang et al. [15] train a GAN for image SR conditioning the output of the network using semantic information extracted by a segmentation CNN. In [16] the authors propose the Residual-in-Residual Dense Block and use it to construct a very deep network that is trained for image SR using perceptual losses. More recently, Zareapoor et al. [30] proposed a dual generator and dual discriminator GAN. Each generator is specialized in different data distributions and the first discriminator distinguishes between real and fake data while the second one assigns examples to the correct generator to be re-synthesized in case of an initial mismatch (see [30] for a complete definition of the used terminology). In [31], Shamsolmoali et al. propose to control the model parameters and mitigate the training difficulties by using a densely connected residual network that is trained using a gradual learning process, from small upsampling factors to larger ones.

As previously stated in Section I, SR and VSR methods can be classified into two groups: model based and learning based. Recently new methods that blend the two approaches have emerged. Zhang et al. [32] use the Alternating Direction Method of Multipliers (ADMM) for image recovering problems with known linear degradation models, such as image deconvolution, blind image deconvolution, and SR. ADMM methods split the recovery problem into two subproblems: a regularized recovery one (subproblem A) and a denoising one (subproblem B). The authors of [32] propose to combine learning and analytical approaches by using a CNN for the denoising problem. This allows them to use the same network for multiple ill posed inverse imaging problems. At the same time, some works have been proposed to increase the performance and the flexibility of SR learning-based models by taking into account the image formation model when training their CNNs. More specifically, Sonderby et al. [21] proposed a new approach which estimates and explicitly uses the image formation model to learn the solution modeled by the network. The blurring and downsampling process to obtain LR frames from HR ones is estimated and the Maximum a Posteriori (MAP) HR image estimation procedure is approximated with the use of a GAN. We improve over this approach by generalizing it to multiple degradation operators for the VSR problem. Although we do not make use of more complex GAN training techniques such as the ones using dual generators and discriminators [30] and gradual learning [31], our method achieves state-of-art results due to the use of the LR image formation model in conjunction with our proposed smoothness constraint. To enforce robustness to multiple degradations in the case of single image SR, Zhang et al. [13] propose to input to their CNN not only the LR image but also the Principal Component Analysis (PCA) representation of the blur kernel used in the degradation process. We adopt a similar approach in our framework, as detailed in the next section. Preliminary results of our approach can be found in [33] for bicubic downsampling. In this work we extend it to multiple degradations and improve the loss function and the architecture used in [33].

III. Model description

In this section, we first introduce the problem of VSR with multiple degradations and explain how we can adapt the Amortised MAP approximation in [21] to solve it (see Section III-A). The loss used to train our GAN model is then introduced in Section III-B. Finally, in Section III-C we describe our proposed new architecture based on the VSRResNet architecture originally introduced in [17].

As previously stated in Section I, we use x to denote a HR frame in a video sequence and y its corresponding observed LR frame. Furthermore, we use $\mathbf{y}$ to refer to the LR frames in a time window around the HR center frame x, $\mathbf{y}$ contains $2l+1$ frames (we use $l=2$ in the experiments).

In the problem of VSR, the process of obtaining a LR image from the HR one is usually modeled using Eq. 1. In this paper, we assume that the image formation noise is negligible ($\epsilon=0$) and that it has been absorbed by the downsampling process. Also, following previous works in the literature, e.g. [13], we assume that D represents bicubic downsampling and the blur k is known and has the form of an isotropic Gaussian kernel. Although more complex blurs, like motion blur, can also be considered, our downsampling and Gaussian blur model is frequently assumed to be a good representation of the high to low resolution degradation process [13].

We also assume that all the frames in the time window are degraded with the same operator. Since this operator depends mostly on the camera used, it is reasonable to assume that these conditions will not change drastically from one frame to another. However, we are not assuming that all cameras produce the same deterioration.

In summary, we assume that D (bicubic downsampling) and k (Gaussian blur) in Eq. 1 are known (or previously estimated) and that they are constant for all frames in $\mathbf{y}$. Notice that assuming that the Gaussian blur is known is not the same as assuming that it is the same for all video sequences. The use of this image forward model leads to a more challenging VSR problem than when only bicubic downsampling is considered, which is the modelling used in most previous works on VSR, see [17], [23], [26], [28], [29], [34], [24].

A. Robust VSR through the use of Amortised MAP

Let us now examine how we can approach the multiple degradations VSR problem. Most current VSR methods solve the problem by learning a function $\mathrm{f}_\theta(.)$, which maps a low resolution image to the high resolution space, using training data pairs x and $\mathbf{y}$ and optimizing the parameters θ using gradient descent over a *Mean Square Error* function. This model embeds the estimation of the downsampling process in the function $\mathrm{f}_\theta(\mathbf{y})$. We argue here that to obtain an SR network capable of dealing with multiple degradations, separating the learning of the HR video sequence from the learning of the degradation makes the whole process more manageable and increases the performance of the network, as shown by the experiments in Section IV. To achieve this, given D and k, we define $A=Dk$ and following [21] consider the function

$$\mathrm{g}_\theta(\mathbf{y}) = (I - A^+ A)\mathrm{f}_\theta(\mathbf{y}) + A^+ y, \tag{2}$$

where A^+ denotes the Moore-Penrose pseudoinverse of the degradation A. Since $AA^+A = A$ and $A^+AA^+ = A^+$, and because the rows of A are independent $AA^+ = I$, we have that

$$Ag_\theta(\mathbf{y}) = A(I - A^+A)f_\theta(\mathbf{y}) + AA^+y = y \tag{3}$$

The resulting $g_\theta(\mathbf{y})$ is an HR image which satisfies Eq. 1 with $\epsilon = 0$. With this formulation, the learning of the network is made easier by learning a residual only (A^+y can be considered as an initial approximation of the HR frame x and $f_\theta(\cdot)$ is part of the added correction according to Eq. 2). This has been exploited in other works of SR where the networks learn to predict a residual over an initial estimation, usually chosen to be the bicubic interpolation [35]. However, these other approaches are not well suited for the Multiple Degrations setting, since the quality of the initial prediction may vary significantly from one degradation to another, showing different kind of artifacts. Figure 1 illustrates this problem for bicubic interpolation.

Notice that in order to use our approach, the estimation of the A^+ operator prior to training is required. In [21], this operator is modeled using a convolution operation followed by a subpixel shuffle layer [10]. The unknown parameters w are estimated by minimizing, via stochastic gradient descent, a loss function, that is,

$$\begin{aligned}\hat{\omega} =& \arg\min_{\omega} \mathbb{E}_x \| Ax - AA^+_\omega(Ax)\|_2^2 \\ &+ \mathbb{E}_y \| A^+_\omega(y) - A^+_\omega(AA^+_\omega(y))\|_2^2,\end{aligned} \tag{4}$$

where A^+_ω denotes the pseudo-inverse with ω network parameters.

An obvious disadvantage of this approach is that one needs to learn a specific $A^+_{\hat{w}}$ for each A. In order to have a single network robust to multiple A operators, we have implemented a network that, for any given A, it predicts the corresponding $\hat{\omega}$ of its pseudoinverse. We choose this network to be composed of three hidden layers with 512, 1024 and 512 hidden units. The input to this network is the PCA representation of the kernel k. The network is trained to predict the unknown $\hat{\omega}$ which solves Eq. 4 for a given A. Our experiments demonstrated that the performance obtained by this efficient approach was equivalent to calculating $\hat{\omega}$ for each A by individually solving for each operator according to Eq. 4.

Let us now see how $g_\theta(\mathbf{y})$ is obtained using an enhanced formulation of a GAN model with increased perceptual quality.

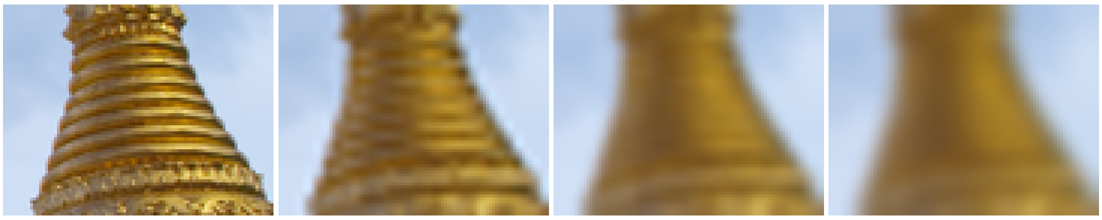
B. A GAN model with increased perceptual quality

Taking into account that the transformation $g_\theta(\mathbf{y})$ defines (from the distribution on $\mathbf{y}$) a probability distribution function $q_\theta(.)$ on the set of HR images, the Kullback-Leibler divergence between $q_\theta(.)$ and the distribution of real images $p_X(.)$, given by

$$\mathrm{KL}(q_\theta || p_X) = \int q_\theta(x) \log \frac{q_\theta(x)}{p_X(x)} dx, \tag{5}$$

is minimized using a GAN approach. This model has a maximum a posteriori approximation interpretation (see [21] for the details).

Together with the generative network, $g_\theta(\mathbf{y})$, we learn a discriminative one, $d_\phi(x)$, using the following two functions on θ and ϕ.



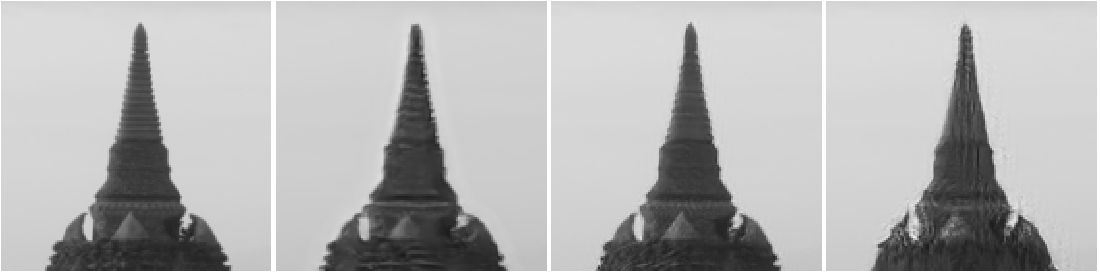
(a) HR (b) Bicubic (c) $\sigma = 2.0$ (d) $\sigma = 3.0$

Fig. 1: Example of artifacts introduced by the bicubic downsampling for a scaling factor of 3. (a) original image, (b) bicubically downsampled image in (a), (c) and (d) bicubically downsampled images which have previously been blurred with $\sigma = 2$ and $\sigma = 3$ Gaussian kernels respectively. The downsampled images have been enlarged to the size of the original one using bicubic upsampling.

$$\mathrm{L_d}(\phi;\theta) = -\mathbb{E}_{x\sim \mathrm{X}}\left[\log \mathrm{d}_\phi(x)\right] - \mathbb{E}_{\mathbf{y}\sim\mathbf{Y}}\left[\log(1-\mathrm{d}_\phi(\mathrm{g}_\theta(\mathbf{y})))\right] \tag{6}$$

$$\mathrm{L_g}(\theta;\phi) = -\mathbb{E}_{\mathbf{y}\sim\mathbf{Y}}\left[\log \frac{\mathrm{d}_\phi(\mathrm{g}_\theta(\mathbf{y}))}{1-\mathrm{d}_\phi(\mathrm{g}_\theta(\mathbf{y}))}\right], \tag{7}$$

where X and Y are the distribution of the HR and LR images, respectively. Iteratively, the algorithm updates ϕ by lowering $\mathrm{L_d}(\phi;\theta)$ while keeping θ fixed, and updates θ by lowering $\mathrm{L_g}(\theta;\phi)$ while keeping ϕ fixed.



(a) HR (b) VSRResGAN factor 2 (c) VSRResAffGAN factor 2 (d) VSRResAffGAN factor 4

Fig. 2: Example of the artifacts produced when only the adversarial loss is used for our GAN model (VSRResAffGAN) and VSRResGAN[17]. We can see how our model is able to recover the frame for factor 2 while VSRResGAN produces a lot of artifacts and a blurred frame. This issue is addressed in our GAN model by the addition of auxiliary losses to the standard GAN loss.

Notice that because the difference between $\mathbb{H}[q_\mathrm{G}, p_\mathrm{X}]$ (cross-entropy) and $\mathrm{KL}[q_\mathrm{G}|p_\mathrm{X}]$ is $\mathbb{H}[q_\mathrm{G}]$, this approximation is expected to lead to solutions with higher entropy and thus produce more diverse frames (see [21]). As can be seen in Fig. 2, this solution leads to a good results for factor 2, however, this is not the case for larger scale factors such as 3 and 4, where the GAN failed to converge. This is most likely caused by the discriminator's ability to easily distinguish between real and generated frames (furthermore, notice that the generator has to produce 16 pixels in the HR frame for each pixel in the input LR frame, which is a considerably more challenging task).

To address this issue, we regularize [36] the training of our GAN network following the approach described in [17] and use the Charbonnier loss between two images u and v (in a given space) defined as

$$\gamma(u,v)=\sum_k\sum_i\sum_j\sqrt{(u_{k,i,j}-v_{k,i,j})^2+\epsilon^2}. \tag{8}$$

The Charbonnier loss is calculated in both pixel and feature spaces. We define our feature space to correspond to the activations provided by a CNN trained for discriminative tasks. For our model, we use the 3rd and 4th convolutional layers of VGG-16 [37] (denoted as VGG(.)). The generator loss then becomes:

$$\begin{aligned}\mathrm{L}_{\mathrm{g\,combined}}(\theta;\phi) &= \alpha \sum_{(x,\mathbf{y})\in T}\gamma(\mathrm{VGG}(\mathrm{x}),\mathrm{VGG}(\mathrm{g}_\theta(\mathbf{y})))\\ &+\beta\mathbb{E}_{\mathbf{y}}\left[\log\frac{1-\mathrm{d}_\phi(\mathrm{g}_\theta(\mathbf{y}))}{\mathrm{d}_\phi(\mathrm{g}_\theta(\mathbf{y}))}\right]+(1-\alpha-\beta)\sum_{(x,\mathbf{y})\in T}\gamma(x,\mathrm{g}_\theta(\mathbf{y})),\end{aligned} \tag{9}$$

where $\alpha,\beta>0$, $\alpha+\beta<1$ and T is the dataset formed by pairs of LR sequences $\mathbf{y}$ and HR images x.

While the use of this loss successfully produces sharper frames, it also introduces high frequency artifacts, especially in smooth areas of the image. They are easily detectable and unpleasant to the human eye (see Fig. 3 for examples of such artifacts). While increasing the weight of the pixel-content loss ($\sum_{(x,\mathbf{y})\in T}\gamma(x,\mathrm{g}_\theta(\mathbf{y}))$) significantly reduces these artifacts, it also smoothes the frame.

Because these artifacts are more prominent in the smooth regions of the frame, we propose the substitution of the pixel-content loss ($\sum_{(x,\mathbf{y})\in T}\gamma(x,\mathrm{g}_\theta(\mathbf{y}))$) with a spatial smoothness constraint. This spatial smoothness constraint is calculated with a weight matrix $\mathrm{M}(x)$ that assigns a larger weight to the pixel-content loss in the smooth areas of the real HR frame x during training. With the incorporation of this spatial smoothness constraint, the generator is penalized heavier during training when generating unwanted "noise" in smooth regions of the frame. We compute this weight matrix as $\mathrm{M}(x)=1-\mathrm{S}(x)$, where $S(x)$ is the Sobel operator applied to the image x. Therefore, the new proposed loss for the generator becomes:

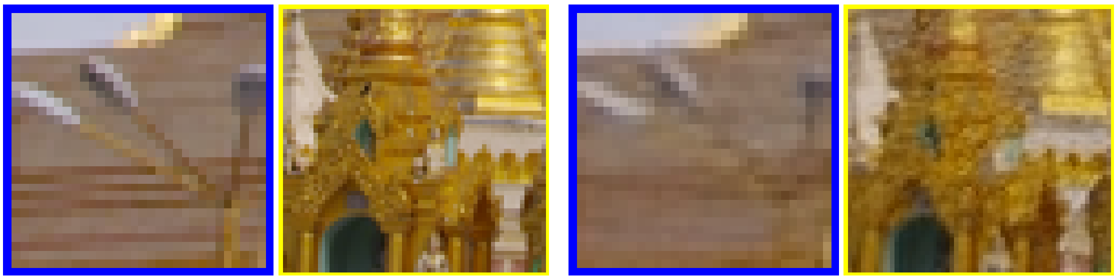
$$\begin{aligned}\mathrm{L}_{\mathrm{g\,combined\,smooth}}(\theta;\phi) &= \alpha \sum_{(x,\mathbf{y})\in T}\gamma(\mathrm{VGG}(\mathrm{x}),\mathrm{VGG}(\mathrm{g}_\theta(\mathbf{y})))\\ &+\beta[\mathbb{E}_{\mathbf{y}}[\log\frac{1-\mathrm{d}_\phi(\mathrm{g}_\theta(\mathbf{y}))}{\mathrm{d}_\phi(\mathrm{g}_\theta(\mathbf{y}))}]]+(1-\alpha-\beta)\sum_{(x,\mathbf{y})\in T}\mathrm{M}(x)\odot\gamma(x,\mathrm{g}_\theta(\mathbf{y})),\end{aligned} \tag{10}$$

where $\alpha,\beta>0$ and $\alpha+\beta<1$ and $\odot$ denotes element-wise multiplication. $\mathrm{L}_{\mathrm{g\,combined\,smooth}}(\theta;\phi)$ is the generator loss that we use to train our GAN model.

In the next section, we describe in detail the CNN architecture used to approximate $\mathrm{f}_\theta(\cdot)$.

C. Architecture

To implement $\mathrm{f}_\theta(\cdot)$, from which we will obtain $g_\theta(y)$ using Eq. 2 which in turn will be used in Eq. 10, we adapt the VSRResNet model introduced in [17]. The authors of [17] found that the VSRResNet architecture results in state-of-the-art performance on the VideoSet4 dataset [38], the test dataset commonly used for evaluating VSR models. The VSRResNet model corresponds to a deep residual CNN that consists of three 3 × 3 convolutional layers



(a) HR (b) VSRResFeatGAN[17]

Fig. 3: Examples of artifacts produced by VSRResFeatGAN[17] for scale factor 3. Notice the high frequency artifacts on smooth areas of the image. With the introduction in the loss function of the spatial smoothness constraint term these artifacts are considerably reduced.

each followed by a ReLU activation, 15 Residuals Blocks with no batch normalization and a final 3 × 3 convolutional layer. Padding is used at each convolution step in order to keep the spatial extent of the feature maps fixed across the network.

We note here that using as input the bicubically upsampled frames as in [17] is not well suited for the multiple degradation setting established in our work. Bicubic upsampling oversmoothes the images and introduces artifacts, making the learning process more difficult. Furthermore, as shown in Fig. 1, these artifacts are degradation operator dependent. Instead, we decided to input the LR video sequence to the network and use the sub-pixel shuffle layer introduced in [10] in the network architecture to learn the up-scaling operation. This avoids the previously mentioned problem and increases training and inference speed.

Figure 4 shows the proposed architecture. Based on the VSRResNet[17], our model is a deep residual CNN with three 3 × 3 convolutional layers each followed by a ReLU activation, 15 Residuals Blocks with no batch normalization and a final 3 × 3 convolutional layer. However, instead of using a bicubically interpolated frame as an input, we upscale the feature maps using the sub-pixel shuffle layer. As can be seen, we do not add the sub-pixel shuffle layer at the end of the network as in [10], but rather we introduce it between the residual blocks of the network. This allows our network to extract features in the LR and HR spaces, increasing its performance. Furthermore, in the case of higher scaling factors like 4 and 8, we introduce several of these layers to allow us to perform progressive upsampling. As shown in Section IV-B the progressive upsampling results in a significant increase in performance. We use only one sub-pixel shuffle layer before the 10th Residual Block for factors 2 and 3. For factor 4 we use a sub-pixel shuffle layer with upscale factor 2 before the 5th and 10th residual blocks and for factor 8 we use an additional one before the last convolutional layer (see Section IV-A for details on model training).

The architecture defined above still suffers from a major problem: the parameters of the network θ depend on the choice of A. Although we ease the training procedure by only predicting the residual using $(I - A_{\hat{\omega}}^{+} A) f_\theta(\mathbf{y})$, the network parameters are dependent on $A_{\hat{\omega}}^{+} y$. In the MDVSR setting, it is necessary for the network parameters to be independent of A. This will allow any input video sequence to be provided to the trained network at test time. To this end, we modify the network architecture such that knowledge of A is provided. This

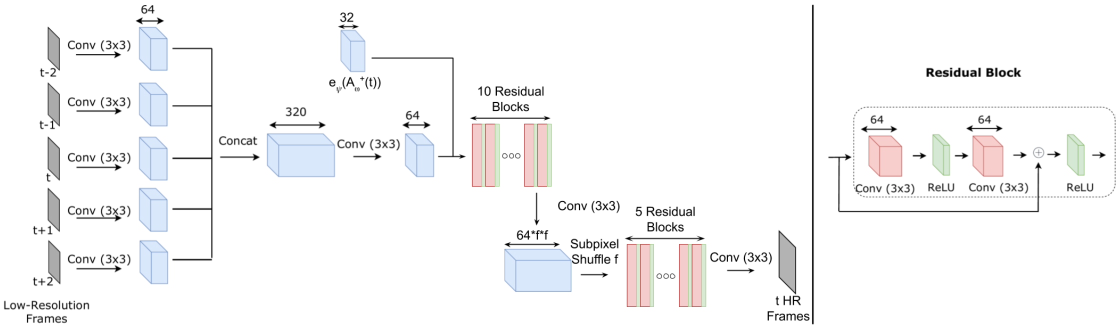


Fig. 4: The MD-AVSR architecture based on VSRResNet[17]. The network consists of a series of convolution operations with 64 kernels of size 3×3, applied to each input frame. The resulting feature maps are then concatenated to obtain 320 feature maps. This is followed by two convolution operations and 15 residual blocks. Each residual block consists of two convolutional operations with 64 kernels of size 3×3, each followed by a ReLU layer. Following the definition of a residual block, the inputted feature maps are added to the output feature maps to obtain the final output of the residual block. For scaling factors 2 and 3, before the 10th Residual block, we up-scale the feature maps by a factor f using a sub-pixel shuffle layer [10]. In the case of factor 4, a sub-pixel shuffle layer with upscale factor 2 is introduced before the 5th and 10th Residual Blocks and an additional one is used before the last convolution for scale factor 8.

will allow the network parameters to be learned for all As and to adapt to any given A at test time. More specifically, we encode $A_{\hat{\omega}}^{+}y$ using a network $\mathrm{e}_{\psi}(\cdot)$ (e_{ψ} in Fig. 4) and feed this compressed representation of $A_{\hat{\omega}}^{+}y$ to the CNN. This encoder network $\mathrm{e}_{\psi}(\cdot)$ seeks to extract the relevant and significant information from the degradation operator to guide the SR process. Notice that other approaches, such as utilizing the Principal Components of k (see [13]) may be considered. However, we argue that using an encoder network is more appropriate as more useful information can be extracted than by using a PCA representation. Our encoder $\mathrm{e}_{\psi}(A_{\hat{\omega}}^{+}y)$ consists of three convolutions of 3×3 and 32 filters with zero-padding, followed by the ReLU activation. Our best results were obtained by incorporating $\mathrm{e}_{\psi}(A_{\hat{\omega}}^{+}y)$ by concatenating the resulting feature maps before the first residual block of the VSRResNet architecture, as shown in Fig. 4. To ensure that the spatial size matches that of $\mathbf{y}$ we use a convolution stride equal to the scaling factor used for training. We jointly train the encoder $\mathrm{e}_{\psi}(\cdot)$ and the super-resolving $\mathrm{f}_{\theta}(\cdot)$ network.

IV. Experimental Results

In this section we detail the training process of the proposed model and show that our proposed approach significantly outperforms current state-of-the-art models for bicubic degradation in addition to being robust to variation on the Gaussian blur deviation.

A. Training hyper-parameters

We use two datasets to train our models: The training sequences from the Myammar dataset and a second one extracted from a subset of the YouTube-8M dataset, which will be used to refine our best performing model.

The Myammar training dataset is formed by 10^6 patches of size 48×48 pixels. Patches with variance less than 0.0035 were determined to be uninformative and were hence removed from the dataset. For each HR patch at time t, we obtain the corresponding LR sequence of patches at time $t-2$, $t-1$, t, $t+1$, and $t+2$. These LR patches are obtained following Eq. 1, i.e., by first blurring the HR frames with a Gaussian kernel and then downsampling the images using bicubic downsampling.

We can distinguish two phases during training. During the first one, only the Myanmar training sequences are used. This phase consists of training the Generator using only the MSE loss ($\mathbb{E}_{x,\mathbf{y}}[\|x-\mathrm{g}_\theta(\mathbf{y})\|^2]$) for 100 epochs. We use the Adam optimizer [39] with the learning rate set to 10^{-3} for the first 50 epochs and then divided by 10 at the 50th and 75th epochs. The weight decay parameter was set to 10^{-5} for all our trained models.

Finally, the second phase uses the proposed GAN framework in Eq. 10. We fine-tuned the model already trained with MSE during the first phase for 30 epochs. The learning rate and weight decay for the discriminator are set to 10^{-4} and 10^{-3}, respectively, while for the generator they are both set to 10^{-4}. The learning rate of both the generator and discriminator are divided by 10 after 15 epochs. During this phase, the models are trained using 1,306,844 blocks of size 48×48 from 5 consecutive images extracted from a subset of YouTube-8M dataset. This subset was constructed by randomly selecting a total of 4358 videos. We consider all categories except those corresponding to video games and cartoons since these categories do not provide an accurate representation of natural scenes we are interested in recovering.

The complete training process takes roughly three days using a Nvidia Titan X GPU with 12 GB of RAM.

B. Ablation study of the proposed modifications of the architecture

To determine the contribution of each proposed component of the new architecture, we perform an ablation study by testing their effects adding one at a time. For ease of comparison, in these experiments, we will use neither the GAN framework nor the perceptual losses, i.e., we will focus on the first phase only (see Section IV-A). For training, we fix the degradation operator to correspond to the bicubic downsampling (no Gaussian blur). Because we only use one degradation for these experiments, we temporarily remove the use of $\mathrm{e}_\psi(A_\omega^+ y))$ from our architecture and instead use as input the LR $\mathbf{y}$ only.

We name the model that uses $\mathrm{g}_\theta(\cdot)$, i.e., the model that utilizes an affine projection, Affine VSR (AVSR) and the model that uses $\mathrm{f}_\theta(\cdot)$ No Affine VSR (NoAVSR) (i.e., it minimizes $\mathbb{E}_{x,\mathbf{y}}\|x-f_\theta(\mathbf{y})\|^2$). Both models include the sub-pixel shuffle layer. To determine the contribution of the sub-pixel shuffle layer in the affine network, we also train an architecture similar to AVSR but using the bicubic up-scaled frames at the input instead of using the subpixel shuffle layer to up-scale the frames. We name this model Bicubic-AVSR. Finally, to test the effects of using the subpixel shuffle layer in a non-affine network, we compare NoAVSR to the a non-affine architecture that uses the bicubic up-scaled frames at the input. We call this model Bicubic-NoAVSR. Notice Bicubic-NoAVSR is equivalent to VSRResNet[17].

Table I contains a quantitative comparison of these models for multiple degradations and up-scaling factors 2, 3 and 4 on the Myanmar video test sequences. We also include a study of the time complexity of each method using the number of Multiplication and Additions (MACs) as a metric. The smaller the number of MACs, the faster the algorithm is. From this table, it is clear that the proposed affine networks (Bicubic-AVSR and AVSR) significantly outperform the other models with a slight increase in MACs. Moreover, AVSR outperforms Bicubic-AVSR in all factors and degradations, showing that, in the case of affine networks, the use of the subpixel shuffle layer is always better than using bicubic interpolation, not only in terms of speed (as reflected by the significant reduction in MACs), but also in terms of fidelity. This experiment shows that even the best performing model, AVSR, is not robust to the use, during testing, of images which have been degraded with Gaussian blur when trained only with bicubic downsampling.

Notice that we did not perform experiments analyzing key components of the underlying residual architecture since it is based on VSRResNet and those elements were studied in [17].

TABLE I: Comparison of the proposed and state-of-the-art models on Myanmar video test sequences and factors 2, 3, and 4. The models were trained using only bicubic degradation. σ refers to the Gaussian blur deviation used during testing. The time complexity of each method is indicated by the number of Multiplications and Additions (MACs) performed for each frame (lower is better).

	Factor	PSNR/SSIM			MACs
		Bicubic	$\sigma = 1.3$	$\sigma = 2.6$	
Bicubic-NoAVSR (VSRResNet[17])	×2	40.58/0.9807	31.97/0.9058	27.85/0.7930	$6.73 * 10^{11}$
	×3	35.97/0.9481	31.77/0.8968	27.78/0.7918	$6.73 * 10^{11}$
	×4	32.85/0.9075	30.73/0.8700	27.64/0.7868	$6.73 * 10^{11}$
NoAVSR	×2	40.52/0.9792	31.97/0.9051	27.85/0.7932	$\mathbf{3.31 * 10^{11}}$
	×3	35.92/0.9474	31.92/0.8965	27.79/0.7924	$\mathbf{2.65 * 10^{11}}$
	×4	33.31/0.9077	31.48/0.8720	27.69/0.7881	$\mathbf{2.82 * 10^{11}}$
Bicubic-AVSR	×2	40.69/0.9811	32.59/0.9162	27.90/0.7935	$6.73 * 10^{11}$
	×3	36.09/0.9487	32.35/0.9065	27.80/0.7925	$6.73 * 10^{11}$
	×4	33.38/0.9077	31.95/0.8902	27.71/0.7883	$6.73 * 10^{11}$
AVSR	×2	**41.23/0.9833**	**32.75/0.9261**	**28.34/0.8180**	$\mathbf{3.31 * 10^{11}}$
	×3	**36.38/0.9527**	**32.53/0.9070**	**27.93/0.8043**	$\mathbf{2.65 * 10^{11}}$
	×4	**33.89/0.9170**	**32.18/0.8927**	**27.77/0.7965**	$\mathbf{2.82 * 10^{11}}$

C. Experiments with Multiple Degradations

Let us now address the multiple degradations scenario. As mentioned in Section I, one of the main factors that significantly lowers the robustness of CNN-based methods to different degradation is the use of only bicubic downsampling during training. Therefore, we will now train the following models with multiple degradations. The degradations considered here are a combination of Gaussian blurs with different kernels k and bicubic downsampling. We generated random Gaussian kernels with σ using a step of 0.1 in the range [0.2, 2.0] for factor 2, [0.2, 3.0] for factor 3, and [0.2, 4.0] for factor 4. The HR video sequences are blurred with these kernels and bicubic downsampling is applied to them to generate the LR samples. We retrain AVSR using these degradations and call this model Blind-MD-AVSR. As stated in Section III-C, we expect it to have a significant loss in performance compared to the one degradation case, since the parameters of the network θ depend on not only $\mathbf{y}$ but also $A^{+}y$ and this information is not provided to the network. This issue is resolved with the introduction in the architecture of our encoding network $\mathrm{e}_{\psi}(A^{+}_{\omega}y))$, as explained in Section III-C. We call this affine network $(\mathrm{g}_{\theta}(\cdot))$ that incorporates the encoded information $\mathrm{e}_{\psi}(A^{+}_{\omega}y))$, MD-AVSR.

We compare our model with current state-of-the-art (SOTA) methods for SR with multiple degradations: IRCNN[32] and SRMDNF[13]. The authors of IRCNN[32] propose the use of ADMM to split the SR problem into two subproblems: a regularized recovery one(subproblem A) and a denoising one (subproblem B). To combine learning and analytical approaches they propose the use of a CNN for the denoising subproblem. SRMDNF[13] consists of a CNN for SR which takes as input the LR image and PCA(k) to provide the network information about the degradation, making it robust to multiple degradations. Notice that all previous SOTA methods are designed for Single Image Super-Resolution (SISR). As mentioned in Section I, this work is the first one to propose a VSR CNN-based method for multiple degradations. Therefore, to provide a fair comparison, we adapt SRMDNF[13] to VSR using our non-affine network ($f_\theta(\cdot)$). We have experimentally determined that the optimal place to add PCA(k) is before the first residual block. We trained this network as we did with MD-AVSR. We call this model VSRMDNF.

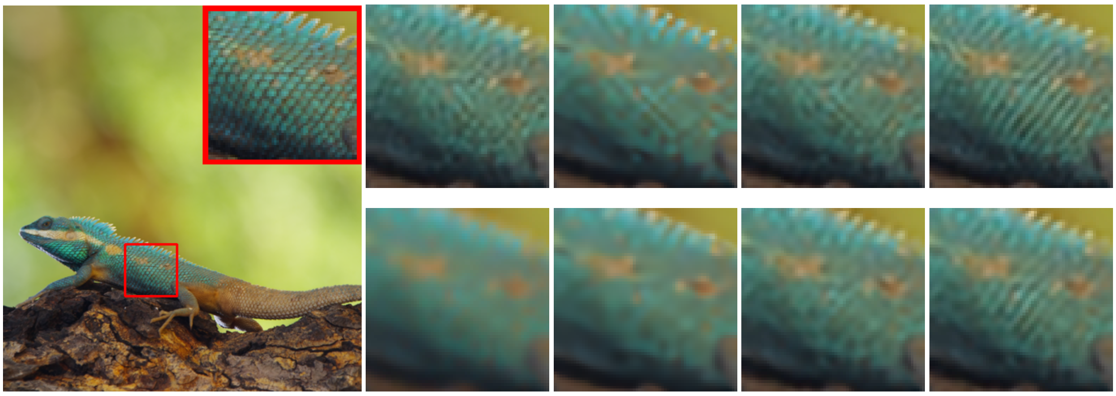


Fig. 5: Qualitative results of our MD-AVSR model compared to current state-of-the-art methods for factor 3 using bicubic downsampling (first row) and Gaussian blur with $\sigma = 2.0$ and bicubic downsampling (second row). Notice how MD-AVSR recovers more details compared to the rest.

Table II shows an experimental comparison with current SOTA models for multiple degradations and factors 2, 3, and 4 on the Myanmar video test sequences. Blind-MD-AVSR has a similar performance across the different degradations, showing that training with multiple degradations significantly increases robustness. However, we observe a sharp decrease in performance for bicubic degradation compared to AVSR (see Table I). This is not the case for the proposed MD-AVSR, which is able to outperform all the other models for all values of σ considered, see also Table I. This indicates the need to incorporate the knowledge about the degradation information if we intend to utilize the same network with multiple degradations (adding them to the training phase is not enough). See also the good performance of the other model which uses information of the degradation process (VSRMDNF). Notice also that MD-AVSR outperforms VSRMDNF in a similar manner as AVSR outperforms NoAVSR and VSRResNet. This indicates that the benefits of using the image formation model are carried over to the multiple degradation setting. It is important to observe in Tables I and II that AVSR and MD-AVSR are the best performing methods when compared to similar approaches. Furthermore, for bicubic downsampling MD-AVSR slightly outperforms AVSR which is expected

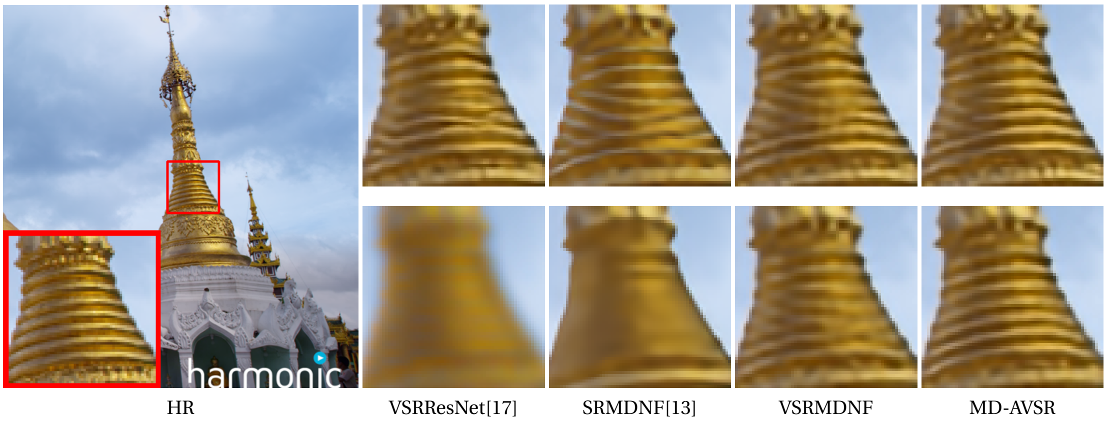


Fig. 6: Qualitative results of our MD-AVSR model compared to current state-of-the-art methods for factor 3 using bicubic downsampling (first row) and Gaussian blur with $\sigma = 3.0$ and bicubic downsampling (second row). The pattern highlighted in the HR image is difficult to recover when only bicubic downsampling is used because the lines either disappear or do not look straight (see Fig. 1). In this case, our MD-AVSR is able to almost fully recover the correct structure while the rest struggle. Furthermore, it is able to fully recover the correct pattern when a strong Gaussian blur of $\sigma = 3.0$ is used during acquisition, while the other methods fail to do so.

TABLE II: Comparison of the proposed and state-of-the-art models on Myanmar video test sequences and factors 2, 3, and 4. Models were trained using multiple degradations. σ refers to the Gaussian blur deviation used. The time complexity of each method is indicated by the number of Multiplications and Additions (MACs) performed for each frame (the lower the better).

	Factor	PSNR/SSIM			MACs
		Bicubic	$\sigma = 1.3$	$\sigma = 2.6$	
IRCNN[32]	×2	37.35/0.9589	35.98/0.9304	26.00/0.4927	$2.94 * 10^{12}$
	×3	34.41/0.9240	34.38/0.9222	31.58/0.8304	$2.94 * 10^{12}$
	×4	31.56/0.8820	31.57/0.8814	31.06/0.8639	$2.94 * 10^{12}$
SRMDNF[13]	×2	39.01/0.9697	38.63/0.9656	35.17/0.9268	$\mathbf{1.96 * 10^{11}}$
	×3	35.08/0.9299	35.12/0.9289	34.03/0.9082	$\mathbf{8.81 * 10^{10}}$
	×4	32.98/0.8952	33.01/0.8949	32.80/0.8889	$\mathbf{5.03 * 10^{10}}$
Blind-MD-AVSR	×2	37.65/0.9523	37.31/0.9503	37.10/0.9458	$3.31 * 10^{11}$
	×3	34.97/0.9330	34.59/0.9275	34.27/0.9200	$2.65 * 10^{11}$
	×4	33.08/0.9002	32.96/0.8992	32.81/0.8972	$2.82 * 10^{11}$
VSRMDNF	×2	40.60/0.9809	39.78/0.9741	37.51/0.9596	$3.33 * 10^{11}$
	×3	35.95/0.9471	35.67/0.9415	34.99/0.9318	$2.65 * 10^{11}$
	×4	33.37/0.9077	33.05/0.9043	32.86/0.8992	$2.82 * 10^{11}$
MD-AVSR	×2	**41.25/0.9834**	**40.11/0.9778**	**37.79/0.9626**	$3.44 * 10^{11}$
	×3	**36.52/0.9525**	**36.17/0.9480**	**35.25/0.9355**	$2.76 * 10^{11}$
	×4	**34.01/0.9185**	**33.86/0.9153**	**33.30/0.9025**	$2.93 * 10^{11}$

to deal with this degradation only. Notice also that although the performance of MD-AVSR slightly deteriorates when blur is introduced, it is more robust than its AVSR counterpart, in other words, MD-AVSR can be safely implemented in systems that are expected to deal with video sequences degraded by different acquisition models. However, this comes with an increase in the number of MACs due to the inclusion of $e_{\psi}(A_{\hat{\omega}}^{+}y))$.

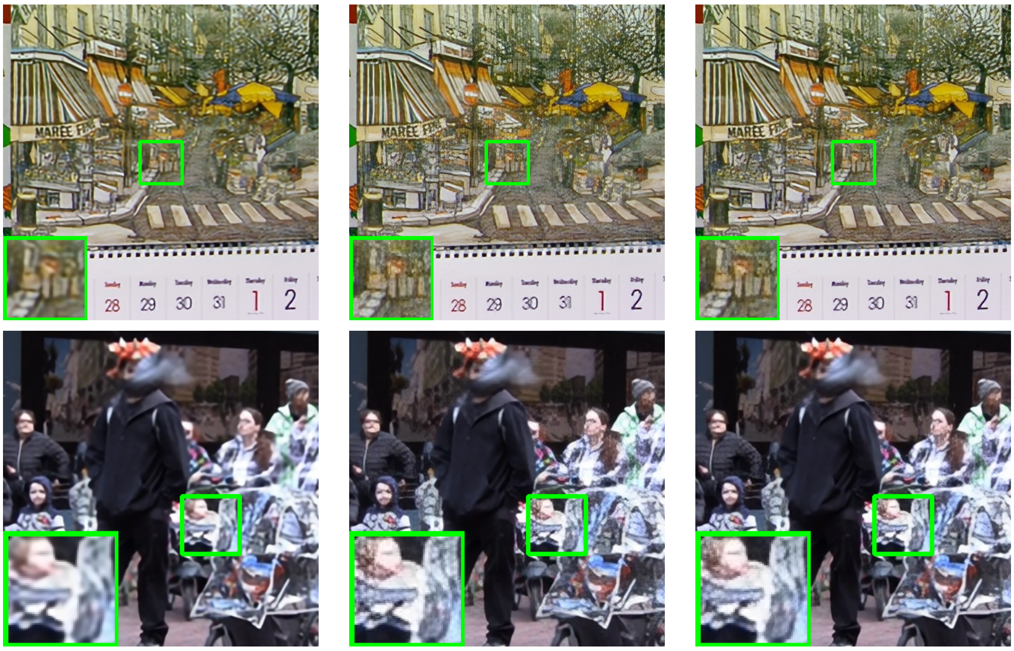
A qualitative comparison of VSRResNet, SRMDNF[13], VSRMDNF and MD-AVSR can be seen in figures 5 and 6. This comparison reveals how our methods are more robust to the multiple degradations and produce HR images of higher quality.

D. Experiments using Perceptual losses

TABLE III: Comparison with state-of-the-art methods on the VideoSet4 [38] dataset on scale factors 2, 3, 4 and 8. The comparison is done in terms of PSNR, SSIM, Perceptual Distance as defined in [40] and the number of Multiplications and Additions (MACs) required for each frame. A smaller Perceptual Distance implies better perceptual quality. MACs indicates the time complexity of the method (lower is better).

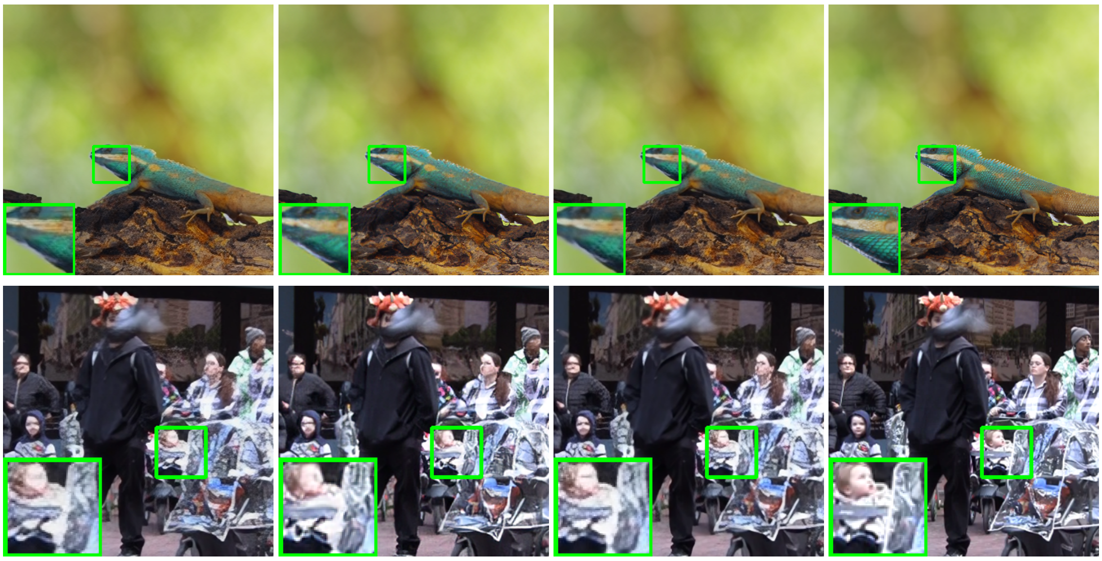
	Factor	PSNR	SSIM	PercepDist	MACs
VDSR[35]	×2	31.61	0.9335	0.0541	$2.52 * 10^{11}$
	×3	26.65	0.8091	0.1355	$2.52 * 10^{11}$
	×4	25.05	0.7292	0.1860	$2.52 * 10^{11}$
RCAN[41]	×2	32.58	0.9414	0.0519	$1.45 * 10^{12}$
	×3	27.71	0.8404	0.1180	$6.52 * 10^{11}$
	×4	25.44	0.7405	0.1695	$3.77 * 10^{11}$
	×8	22.33	0.5053	0.3277	$1.07 * 10^{11}$
VESPCN[27]	×3	27.25	0.8253	0.1533	$\mathbf{5.74 * 10^{9}}$
	×4	25.35	0.7309	0.2022	$\mathbf{3.27 * 10^{9}}$
SPMC-SR[29]	×2	30.92	0.9235	0.0899	$\mathbf{4.97 * 10^{10}}$
	×3	27.49	0.84	-	$4.88 * 10^{10}$
	×4	25.63	0.7709	0.1908	$4.85 * 10^{10}$
TAN[34]	×4	25.53	0.7475	0.1798	-
VSRResNet[17]	×2	31.87	0.9426	0.0407	$4.91 * 10^{11}$
	×3	27.80	0.8571	0.1209	$4.91 * 10^{11}$
	×4	25.51	0.7530	0.1766	$4.91 * 10^{11}$
	×8	22.35	0.5072	0.3286	$4.91 * 10^{11}$
VSRResFeatGAN[17]	×2	30.90	0.9241	0.0283	$4.91 * 10^{11}$
	×3	26.53	0.8148	0.0668	$4.91 * 10^{11}$
	×4	24.50	0.7023	0.1043	$4.91 * 10^{11}$
	×8	22.12	0.5025	0.2773	$4.91 * 10^{11}$
ERSGAN[16]	×4	22.98	0.6336	0.0993	$4.24 * 10^{11}$
MD-AVSR	×2	**33.00**	**0.9496**	0.0292	$2.51 * 10^{11}$
	×3	**28.31**	**0.8751**	0.1081	$2.01 * 10^{11}$
	×4	**26.17**	**0.7895**	0.1655	$2.13 * 10^{11}$
	×8	**22.74**	**0.5126**	0.3243	$\mathbf{7.16 * 10^{10}}$
MD-AVSR-FG	×2	31.54	0.9309	0.0229	$2.51 * 10^{11}$
	×3	26.73	0.8237	0.0588	$2.01 * 10^{11}$
	×4	25.10	0.7414	0.0939	$2.13 * 10^{11}$
	×8	22.28	0.5047	0.2715	$\mathbf{7.16 * 10^{10}}$
MD-AVSR-P	×2	31.82	0.9345	0.0216	$2.51 * 10^{11}$
	×3	27.20	0.8383	0.0586	$2.01 * 10^{11}$
	×4	25.26	0.7501	0.0927	$2.13 * 10^{11}$
	×8	22.36	0.5056	0.2708	$\mathbf{7.16 * 10^{10}}$
MD-AVSR-PY8	×2	31.81	0.9391	**0.0210**	$2.51 * 10^{11}$
	×3	27.09	0.8412	**0.0571**	$2.01 * 10^{11}$
	×4	25.18	0.7555	**0.0916**	$2.13 * 10^{11}$
	×8	22.31	0.5064	**0.2699**	$\mathbf{7.16 * 10^{10}}$

Let us now test the GAN framework (see Eq. 10) and analyze the performance of the new loss in comparison to the one proposed in [17] (see Eq. 9). This corresponds to the 2nd phase of the training described in Section IV-A. We first use during this 2nd phase the loss proposed in [17], to fine-tune the MD-AVSR model with $\alpha = 0.998$ and $\beta = 0.001$ for the hyper-parameters of the loss function. We call this model, which incorporates features loss (F) and a GAN (G), MD-AVSR-FG. Then, we fine-tuned in the same fashion the MD-AVSR model with the new proposed perceptual (P) loss (see Eq. 10) and call it MD-AVSR-P. The



MD-AVSR MD-AVSR-FG MD-AVSR-P

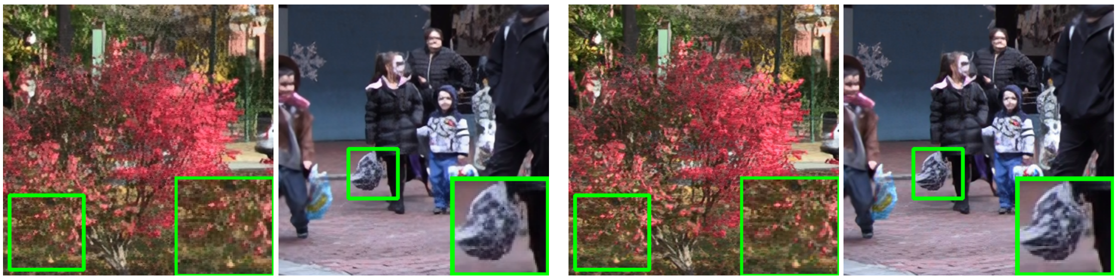
Fig. 7: Qualitative comparison between MD-AVSR, MD-AVSR-FG and MD-AVSR-P for factor 4 with bicubic downsampling. We can see that MD-AVSR-P does not produce artifacts like MD-AVSR-FG does and produces frames that look much sharper than MD-AVSR.



VSRResFeatGAN[17] ERSGAN[16] MD-AVSR-P HR

Fig. 8: Comparison between VSRResFeatGAN[17], ERSGAN[16] and MD-AVSR-P for factor 4 with bicubic downsampling.

values of the loss hyper-parameters are: $\alpha = 0.049$ and $\beta = 0.001$. Both models are trained using the Myammar dataset, instead of the YouTube-8M dataset. Finally, to demonstrate the influence of the diversity of the training dataset for GAN-based VSR models, we use the YouTube-8M dataset and our new loss in Eq. 10 to fine-tune MD-AVSR and obtain our final model MD-AVSR-PY8.



MD-AVSR-P

MD-AVSR-PY8

Fig. 9: Comparison between MD-AVSR-P and MD-AVSR-PY8 for factor 4 with bicubic downsampling.

Table III contains a comparison of these models with multiple state-of-the-art methods in terms of PSNR, SSIM, Perceptual Distance [40] and the number MACs per frame for the VideoSet4 video sequences [38] and factors 2, 3, 4 and 8. These include SISR methods with Deep CNN like VDSR[35], RCAN[41] and ERSGAN[16]. VDSR[35] uses a very deep convolutional neural network that predicts residuals between the upscaled low-resolution image and the high-resolution image. RCAN[41] consists of a very deep residual CNN that uses a new type of block that exploits channel attention to further increase the performance. ERSGAN[16] proposes the use of new residual-in-residual block to construct a very deep residual CNN and train it within a GAN framework to obtain photo-realistic SR images. Both methods use an architecture several times deeper than the one proposed here. We also include the following VSR methods: VESPCN[27], SPMC-SR[29], TAN[34] and VSRResNet and VSRResFeatGAN from [17]. VESPCN[27] incorporates time information in the network by using the motion compensated future and past frames in a time window and uses a sub-pixel convolution to upscale the output. SPMC-SR[29] proposes a convolution-LSTM neural network with efficient motion compensation learned jointly with the network. Lastly, TAN[34] uses a Temporal Adaptive Network that consists of a network with multiple SR branches, each responsible for super-resolving frames at a temporal scale. Then, a temporal modulation branch fuses the multiple VSR solutions into a single one. Notice that all these methods work only with bicubic downsampling, in contrast with SOTA methods in the previous section.

In Table III we observe how the proposed MD-AVSR-P outperforms all models trained with perceptual losses (VSRResFeatGAN[17], ERSGAN[16] and MD-AVSR-FG) for all figures of merits when trained on Myanmar training sequences. Furthermore, a close examination of the generated frames and a comparison to the MD-AVSR-FG ones (see Fig. 7) shows that they are almost artifact-free. The model also shows a noticeable increase in sharpness compared to MD-AVSR. However, this increase in sharpness (reflected on the improvement on Perceptual Distance) comes with a decrease in PSNR and SSIM. This decrease aligns with the one shown in other models that use perceptual losses, like VSRResFeatGAN[17] or ERSGAN[16]. Figure 8 shows a qualitative comparison of VSRResFeatGAN[17] and ERSGAN[16] with MD-AVSR-P. We can see that the proposed model MD-AVSR-P outperforms current state-of-the-art methods. Notice that values for factors 2 and 3 for ERSGAN[16] could not be calculated since the authors did provide weights for those. Notice also how the proposed model outperforms, in terms of picture quality, models much more complex (with two times more MACs) like ERSGAN[16].

Table III also shows that MD-AVSR-PY8 outperforms MD-AVSR-P in terms of SSIM and Perceptual Distance significantly, although it suffers from a minor decrease in PSNR. This increase shows the importance of using a very diverse dataset during training for GAN models in contrast to experiments carried out with the non GAN model MD-AVSR, where the training with this new dataset did not produce results different enough to be significant for any factor. These results show that GAN-based VSR models require more data that other CNN. Figure 9 shows a qualitative comparison between MD-AVSR-P and MD-AVSR-PY8. It can be seen that MD-AVSR-PY8 is able to produce more detailed and realistic looking images.

V. Conclusions

In this work we have first introduced a multiple degradation Video Super-Resolution approach that explicitly utilizes the LR image formation model as an input to the network. The model, named MD-AVSR, has been trained with MSE only. The experiments show that MD-AVSR outperforms current state-of-the-art methods in terms of PSNR and SSIM for both multiple degradation and bicubic degradation only settings. We have then proposed a GAN-based approach that uses a new perceptual loss combining an adversarial loss, a feature loss, and a spatial smoothness constraint (MD-AVSR-P). This method improves the quality of the super resolved frames without the introduction of noticeable high frequency artifacts. The results show that it outperforms current state-of-the-art methods in terms of perceptual quality and in all metrics used by GAN methods trained without using the proposed perceptual loss. Finally, we use a much more diverse dataset created from a subset of the YouTube-8M dataset to train MD-AVSR-P and show that the new MD-AVSR-PY8 obtains significantly better results in terms of perceptual quality.

Acknowledgements

This work was supported in part by the Sony 2016 Research Award Program Research Project. The work of Santiago López-Tapia and Rafael Molina was supported by the the Spanish Ministry of Economy and Competitiveness through project DPI2016-77869-C2-2-R and the Visiting Scholar program at the University of Granada. Santiago López-Tapia received financial support through the Spanish FPU program.

References

[1] C. Liu and D. Sun, "On Bayesian adaptive video super resolution," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 2, pp. 346–360, 2014.

[2] A. K. Katsaggelos, R. Molina, and J. Mateos, "Super resolution of images and video," *Synthesis Lectures on Image, Video, and Multimedia Processing*, vol. 1, no. 1, pp. 1–134, 2007.

[3] S. P. Belekos, N. P. Galatsanos, and A. K. Katsaggelos, "Maximum a posteriori video super-resolution using a new multichannel image prior," *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1451–1464, 2010.

[4] S. D. Babacan, R. Molina, and A. K. Katsaggelos, "Variational Bayesian super resolution," *IEEE Transactions on Image Processing*, vol. 20, no. 4, pp. 984–999, 2011.

[5] W. T. Freeman, T. R. Jones, and E. C. Pasztor, "Example-based super-resolution," *IEEE Computer Graphics and Applications*, vol. 22, no. 2, pp. 56–65, 2002.

[6] C. Dong, C. C. Loy, K. He, and X. Tang, “Learning a deep convolutional network for image super-resolution,” in *European Conference on Computer Vision*, pp. 184–199, Springer, 2014.

[7] Hong Chang, Dit-Yan Yeung, and Yimin Xiong, “Super-resolution through neighbor embedding,” in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, vol. 1, pp. I–I, 2004.

[8] J. Yang, J. Wright, T. S. Huang, and Y. Ma, “Image super-resolution via sparse representation,” *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp. 2861–2873, 2010.

[9] J. Yang, Z. Wang, Z. Lin, S. Cohen, and T. Huang, “Coupled dictionary training for image super-resolution,” *IEEE Transactions on Image Processing*, vol. 21, no. 8, pp. 3467–3478, 2012.

[10] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, “Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network,” *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1874–1883, 2016.

[11] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, *et al.*, “Photo-realistic single image super-resolution using a generative adversarial network,” *arXiv preprint arXiv:1609.04802*, 2016.

[12] G. Riegler, S. Schulter, M. Rüther, and H. Bischof, “Conditioned regression models for non-blind single image super-resolution,” in *IEEE International Conference on Computer Vision*, pp. 522–530, Dec 2015.

[13] K. Zhang, W. Zuo, and L. Zhang, “Learning a single convolutional super-resolution network for multiple degradations,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[14] J. Johnson, A. Alahi, and L. Fei-Fei, “Perceptual losses for real-time style transfer and super-resolution,” in *European Conference on Computer Vision*, pp. 694–711, Springer, 2016.

[15] X. Wang, K. Yu, C. Dong, and C. Change Loy, “Recovering realistic texture in image super-resolution by deep spatial feature transform,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 606–615, 2018.

[16] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, C. C. Loy, Y. Qiao, and X. Tang, “Esrgan: Enhanced super-resolution generative adversarial networks,” in *ECCV Workshops*, 2018.

[17] A. Lucas, S. López-Tapia, R. Molina, and A. K. Katsaggelos, “Generative adversarial networks and perceptual losses for video super-resolution,” *IEEE Transactions on Image Processing*, vol. 28, pp. 3312–3327, July 2019.

[18] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in *Advances in neural information processing systems*, pp. 2672–2680, 2014.

[19] P. Isola, J. Zhu, T. Zhou, and A. A. Efros, “Image-to-image translation with conditional adversarial networks,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5967–5976, July 2017.

[20] T. M. Quan, T. Nguyen-Duc, and W.-K. Jeong, “Compressed sensing mri reconstruction with cyclic loss in generative adversarial networks,” *arXiv preprint arXiv:1709.00753*, 2017.

[21] C. Sonderby, J. Caballero, L. Theis, W. Shi, and F. Huszar, “Amortised MAP inference for image super-resolution,” in *International Conference on Learning Representations*, 2017.

[22] S. Abu-El-Haija, N. Kothari, J. Lee, A. P. Natsev, G. Toderici, B. Varadarajan, and S. Vijayanarasimhan, “Youtube-8m: A large-scale video classification benchmark,” in *arXiv:1609.08675*, 2016.

[23] R. Liao, X. Tao, R. Li, Z. Ma, and J. Jia, “Video super-resolution via deep draft-ensemble learning,” in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 531–539, 2015.

[24] A. Kappeler, S. Yoo, Q. Dai, and A. K. Katsaggelos, “Video super-resolution with convolutional neural networks,” *IEEE Transactions on Computational Imaging*, vol. 2, no. 2, pp. 109–122, 2016.

[25] Y. Huang, W. Wang, and L. Wang, “Bidirectional recurrent convolutional networks for multi-frame super-resolution,” in *Advances in Neural Information Processing Systems*, pp. 235–243, 2015.

[26] D. Li and Z. Wang, “Video super-resolution via motion compensation and deep residual learning,” *IEEE Transactions on Computational Imaging*, 2017.

[27] J. Caballero, C. Ledig, A. Aitken, A. Acosta, J. Totz, Z. Wang, and W. Shi, “Real-time video super-resolution with spatio-temporal networks and motion compensation,” *arXiv preprint arXiv:1611.05250*, 2016.

[28] O. Makansi, E. Ilg, and T. Brox, “End-to-end learning of video super-resolution with motion compensation,” in *German Conference on Pattern Recognition*, pp. 203–214, Springer, 2017.

[29] X. Tao, H. Gao, R. Liao, J. Wang, and J. Jia, “Detail-revealing deep video super-resolution,” *arXiv preprint arXiv:1704.02738*, 2017.

[30] M. Zareapoor, H. Zhou, and J. Yang, “Perceptual image quality using dual generative adversarial network,” *Neural Computing and Applications*, pp. 1–11, 2019.

[31] P. Shamsolmoali, M. Zareapoor, R. Wang, D. K. Jain, and J. Yang, “G-ganisr: Gradual generative adversarial network for image super resolution,” *Neurocomputing*, vol. 366, pp. 140 – 153, 2019.

[32] K. Zhang, W. Zuo, S. Gu, and L. Zhang, “Learning deep CNN denoiser prior for image restoration,” *CoRR*, vol. abs/1704.03264, 2017.

[33] S. Lopez-Tapia, A. Lucas, R. Molina, and A. K. Katsaggelos, “Gan-based video super-resolution with direct regularized inversion of the low-resolution formation model,” in *ICIP*, 2019.

[34] D. Liu, Z. Wang, Y. Fan, X. Liu, Z. Wang, S. Chang, and T. Huang, “Robust video super-resolution with learned temporal dynamics,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2507–2515, 2017.

[35] J. Kim, J. K. Lee, and K. M. Lee, “Accurate image super-resolution using very deep convolutional networks,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1646–1654, June 2016.

[36] T. Che, Y. Li, A. P. Jacob, Y. Bengio, and W. Li, “Mode regularized generative adversarial networks,” in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017.

[37] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[38] C. Liu and D. Sun, “A bayesian approach to adaptive video super resolution,” in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 209–216, IEEE, 2011.

[39] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *CoRR*, vol. abs/1412.6980, 2014.

[40] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, “The unreasonable effectiveness of deep networks as a perceptual metric,” in *CVPR*, 2018.

[41] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, “Image super-resolution using very deep residual channel attention networks,” in *ECCV*, 2018.

Santiago López-Tapia received the bachelor’s and master’s degrees in computer science from the University of Granada in 2014 and 2015, respectively. He is currently pursuing the Ph.D. degree with the Visual Information Processing Group, Department of Computer Science and Artificial Intelligence, University of Granada. His research mainly focusses in the use of deep learning models for image restoration and classification.

Alice Lucas received the B.S. degree in applied math, engineering, and physics from the University of Wisconsin–Madison in 2015 and the M.S. degree in electrical engineering from Northwestern University in 2017, where she is currently pursuing the Ph.D. degree with the Image and Video Processing Laboratory (IVPL). Her research at IVPL is centered on the use of deep learning models for various image processing tasks, with focus on the task of video super-resolution (VSR). She received the Certificate in Computer Science from the University of Wisconsin–Madison in 2015.

Rafael Molina received the degree in mathematics and the Ph.D. degree in optimal design in linear models from the University of Granada, Granada, Spain, in 1979 and 1983, respectively. He was the Dean of the Computer Engineering School, University of Granada, from 1992 to 2002. In 2000, he joined the University of Granada, as a Professor of computer science and artificial intelligence. He was the Head of the Computer Science and Artificial Intelligence Department, University of Granada, from 2005 to 2007. His research focuses on using Bayesian modeling and inference in problems like image restoration, active learning, and machine learning.

Aggelos K. Katsaggelos received the Diploma degree in electrical and mechanical engineering from the Aristotelian University of Thessaloniki, Greece, in 1979, and the M.S. and Ph.D. degrees in electrical engineering from the Georgia Institute of Technology in 1981 and 1985, respectively. In 1985, he joined the Department of Electrical Engineering and Computer Science, Northwestern University, where he is currently a professor. He was the Ameritech Chair of information technology and the AT&T Chair, and he is the Joseph Cummings Chair. He has published extensively in the areas of signal processing and communications, computational imaging, and machine learning (over 300 journal papers).

2.2 Gated Recurrent Networks for Video Super Resolution

2.2.1 Publication details

Authors: Santiago López-Tapia, Alice Lucas, Rafael Molina, and Aggelos K. Katsaggelos.
Title: Gated Recurrent Networks for Video Super Resolution.
Publication: 28th European Signal Processing Conference, EUSIPCO 2020, Amsterdam (Netherlands), January 2021.
Status: Accepted for publication.
Quality indices:

- GGS Rating: B
- GGS Class: 3
- CORE: B

2.2.2 Main Contributions

- Inspired by the deformable convolutions introduced in [128], we develop a new attention mechanism that we name deformable attention.
- We propose a new recurrent CNN for VSR. This model adapts the minimal Gated Recurrent Unit [129] to VSR: current time step state is calculated by combining features extracted using a deep residual CNN and last time step state. Deformable attention is used to align the features and the previous state so that they can be combined with a gated sum. By using this system, our model can more effectively reuse information extracted from previous frames in the sequence.
- We perform experiments using standard VSR datasets and compare to the current state-of-art in terms of PSNR, SSIM and temporal consistency (T-RRED[130]).

Gated Recurrent Networks for Video Super Resolution

Santiago López-Tapia [*], Alice Lucas[†], Rafael Molina[*], Aggelos K. Katsaggelos[†]
[*] *Dept. of Computer Science and Artificial Intelligence, University of Granada,*
Granada, Spain
Email: {sltapia, rms}@decsai.ugr.es
[†] *Dept. of Electrical Engineering and Computer Science Northwestern University*
Evanston, IL, USA
Email: alicelucas2015@u.northwestern.edu, aggk@eecs.northwestern.edu

Abstract

Despite the success of Recurrent Neural Networks in tasks involving temporal video processing, few works in Video Super-Resolution (VSR) have employed them. In this work we propose a new Gated Recurrent Convolutional Neural Network for VSR adapting some of the key components of a Gated Recurrent Unit. Our model employs a deformable attention module to align the features calculated at the previous time step with the ones in the current step and then uses a gated operation to combine them. This allows our model to effectively reuse previously calculated features and exploit longer temporal relationships between frames without the need of explicit motion compensation. The experimental validation shows that our approach outperforms current VSR learning based models in terms of perceptual quality and temporal consistency.

Index Terms

Video, Super-resolution, Convolutional Neuronal Networks, Recurrent Neural Networks

I. Introduction

$$z \sim \mathcal{N}(0, 1)$$

Image Super-Resolution (SR) is one of the fundamental low-level vision problems. It consists of recovering a High-Resolution (HR) image from a given set of Low-Resolution (LR) images. In recent years, the introduction of high and ultra high definition displays has increased the demand for methods to convert already existing LR videos into HR ones. This is known as Video Super-Resolution (VSR), a special case of SR where the input and output are

This work was supported in part by the Sony 2016 Research Award Program Research Project. The work of SLT and RM was supported by the Spanish Ministry of Economy and Competitiveness through project DPI2016-77869-C2-2-R and the Visiting Scholar program at the University of Granada. SLT received financial support through the Spanish FPU program.

sequences of LR and HR video frames, respectively. The formation of each LR image from its corresponding HR image can be written as:

$$y = \downarrow (x \otimes k) + \epsilon, \qquad (1)$$

where x is the HR image, y is the LR one, $x \otimes k$ represents the convolution of x with the blur kernel k, $\downarrow$ is a downsampling operator (typically bicubic downsampling) and ϵ is the noise, usually Additive White Gaussian Noise (AWGN).

We can divide current (V)SR methods into two categories: model-based and learning-based. Model-based approaches explicitly define the LR image formation model (see Eq. 1) and typically recover the HR image by optimizing an energy function built from the LR image formation model [1], [2], [3]. In the case of learning-based algorithms, most of them do not explicitly define or use the image formation model, instead they use a large training databases of HR and LR image/sequence pairs to learn a mapping from the LR observations to the HR. When learning from data, Convolutional Neural Networks (CNN) have become a popular tool due to their high performance in other vision based tasks. In recent years, several VSR CNN-based models have been proposed: Caballero et al. [4] jointly train a spatial transformer network and an SR network to warp the video frames, the approach benefits from sub-pixel information. Tao et al. [5] propose to increase the performance of [4] by jointly upsampling and motion compensation (MC). Liu et al. [6] propose to train a neural network to learn the temporal dependency between input frames increasing the quality of the HR prediction. Kappeler et al. [7] propose to train a CNN which takes bicubically interpolated LR frames as input and learn the direct mapping that reconstructs the central HR frame. Following [7], Lucas et al. introduce in [8] a deep residual network trained using feature and adversarial losses that increased the perceptual quality of the output. Finally, in [9] we introduce a CNN GAN model that use the LR image formation model and a spatial-constraint to further increase the perceptual quality without the introduction of visual artifacts.

Despite all the work carried out in VSR in recent years using CNN-based models, few of them have used recurrent architectures. Recurrent Neural Networks (RNN) have been applied with great success to speech recognition and other tasks which require processing a time sequence, see [10], [11] for video related problems. One of the most frequently use recurrent units is the Gated Recurrent Unit (GRU) [12], which efficiently exploits the existing correlation within long sequences. However, in the case of VSR, these types of architectures have been hardly used and all the proposed methods employ simpler recurrent units (the result of processing the previous frame is used as input to the current one in contrast to general GRUs which use gated operations to update an internal state). In [13], the authors propose to use Recurrent Convolutional Layers to exploit the relations between frames more effectively. The network is very shallow since it does not use residual blocks. More recently, a very deep residual was proposed in [14]. The models use a Convolutional Recurrent Neural Network (RCNN) that uses the super-resolved frame in the previous time step.

In this work, we propose a new RCNN that employs more effectively the previous time step information. Our model, inspired by the deformable convolutions introduced in [15], defines and utilizes fast deformable attention modules. These modules are used to align previously calculated features to the current time step ones. Then, our model combines them using a gated sum. Experiments show that this new architecture outperforms current VSR state-of-art methods in terms of PSNR and time consistency.

The rest of the paper is organized as follows: Section II-A presents our new recurrent model for VSR. In section III, we describe and discuss our experiments with the proposed model and comparison with state-of-art VSR algorithms. Finally, conclusions are presented in section IV.

II. Model description

In this work we use y_t to denote the LR frame at time t in a video sequence and x_t its corresponding HR. We model the process of obtaining a LR observation from a HR using Eq. 1. Following the literature [4], [5], [6], [7], [8], we assume that the noise is negligible ($\epsilon = 0$) and that the downsampling process ($\downarrow (x \otimes k)$) can be modeled using bicubic downsampling.

A. Architecture

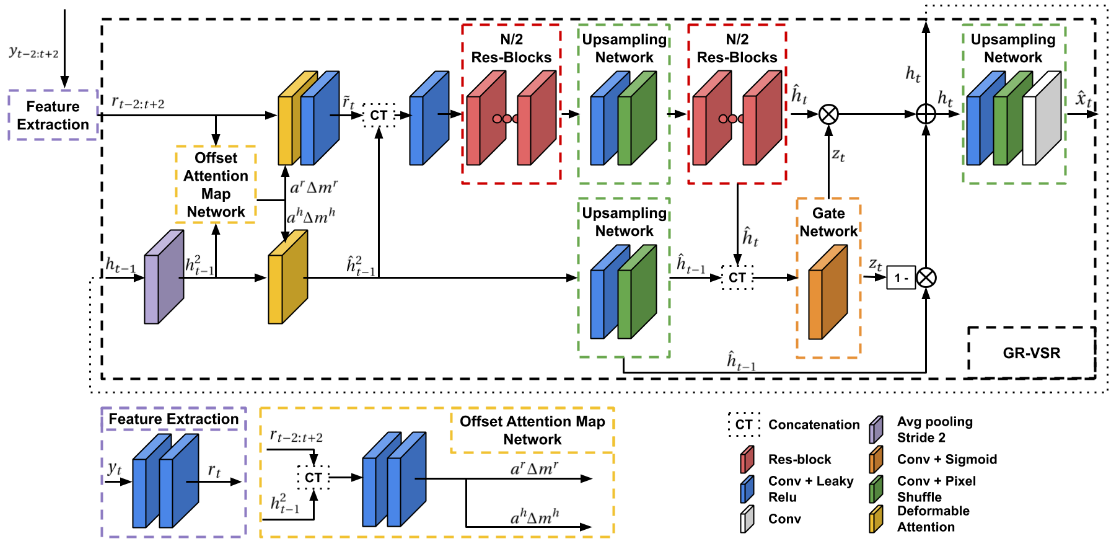


Fig. 1: The proposed GR-VSR architecture. Each convolution operation uses 64 kernels of size 3×3, with the exception of the offset calculation, that uses $6 \times 3 \times 3 \times 3$ filters. Each residual block consists of two convolutional operations with 64 kernels of size 3×3, each followed by a ReLU layer. See text for details.

The architecture of our proposed model, Gated Recurrent Video Super Resolution (GR-VSR), can be seen in Fig. 1. Our network is a recurrent network that uses the features h_{t-1} obtained at the previous time step from the convolution operation located right before the last upsampling module (they constitute the hidden state of our network) together with the features $r_{t-2:t+2}$ calculated using $y_{t-2:t+2}$. Using both, h_{t-1} and $r_{t-2:t+2}$, and a deformable attention module (see Section II-B) we calculate and align with y_t the features $\hat{h}^2_{t-1}$ and $\hat{r}_{t-2:t+2}$. These features are then concatenated and transformed using 8 residual blocks to produce $\hat{h}_t$. The hidden state h_t is finally calculated as:

$$h_t = z_t \odot \hat{h}_t + (1 - z_t) \odot \hat{h}_{t-1}, \tag{2}$$

where $\odot$ indicates element-wise multiplication and z_t is a weight matrix with the same size as $\hat{h}_{t-1}$ and $\hat{h}_t$ which is calculated using a gate network. By using this update rule, our model is able to decide how much information from the current state will be added to the hidden state and how much of such information will be forgotten. Notice that this can be seen as a long skip connection. It is used in conjunction with residual blocks to construct deep CNNs[16]. It is important to mention that this connection is not used to increase the depth of the model (see [16]) but to allow the feedback provided by future frames to reach previous time steps during training. Furthermore, notice also that the use of this residual connection is not effective without alignment. Without alignment, the movement of the objects in the scene will cause the model to combine information from different locations in the scene, this will damage the HR prediction on those locations.

As it can be seen, the proposed architecture propagates the features contained in h_t instead of the predicted pixel values of $\hat{x}_t$ as done in [14]. Our approximation allows to propagate more information through time, potentially encoding information from multiple time steps. For each spatial location our model encodes and transmits to the next step a vector of information, instead of just a value. This combined with the previous update rule (see Eq. 2) allows our model to propagate thought time much more information and during more time steps.

The use of a hidden state poses the problem that it has to be initialized. Though it can be initialized by a 0 vector, like in RNN used for text processing, we have detected in our experiments that this not only damages the prediction of the first couple of frames, but also, and more importantly, it makes the training very unstable. Thus, for the first frame prediction we propose the use of an auxiliary network g_ϕ identical to the proposed one but without the recurrent connection and related modules (deformable attention and gate network). Note that, although the use of this method increases the processing time, this is negligible for long video sequences since it is only needed for the first frame.

It is interesting to note that we can establish a parallelism between our proposed architecture and one of the GRU modifications: minimal GRU (see [17]). This minimal GRU performs the following operations:

$$\begin{aligned}\hat{h}_t &= \phi(W_h s_t + U_h h_{t-1} + b_h) \\ z_t &= \sigma(W_z s_t + U_z h_{t-1} + b_z) \\ h_t &= z_t \odot \hat{h}_t + (1 - z_t) \odot h_{t-1},\end{aligned} \tag{3}$$

where s_t is an input vector, W_* and U_* are learnable weight matrices, b_* biases, σ is the sigmoid activation and ϕ the tanH activation. As it can be seen, the first calculation depends on the input and the previous state. This operation corresponds in our model to the 8 residual blocks. Finally, the second and third calculation corresponds to our gated residual connection. Notice that the GRU it is not appropriated for image processing due to the lack of depth (no residual connections) and, in contrast to our model, it does not have a mechanism to align features.

B. Deformable attention

As we have already indicated in section II-A our model makes use of the deformable attention to align the features. Our deformable attention module is a modification of the deformable convolutions. Deformable convolutions were first proposed in [18] and enhanced in [15] to handle with more deformations. They correspond to the use of the following convolution operation

$$f^l(m) = \sum_{n=1}^{N} w_n a_{m,n} f^{l-1}(m + n + \Delta m_{m,n}), \tag{4}$$

where $f^l(m)$ denotes the feature vector in layer l at location m, n is a position of the convolutional kernel and $\Delta m_{m,n}$ and $a_{m,n}$ are learnable offsets and modulation scalar, respectively. These offsets and modulation parameters are calculated using another convolution layer for each m location in the image.

In our deformable attention module we impose to the model in Eq. 4 the following constraints: $\sum_n^N a_{m,n} = 1$ and weights $w_n = 1$. By doing so we have an attention mechanism similar to the one proposed in [19] but with the advantage of being able to use locations outside the window around location m:

$$f^l(m) = \sum_{n=1}^{N} a_{m,n} f^{l-1}(m + n + \Delta m_{m,n}). \tag{5}$$

This attention mechanism, unlike other attention methods, can handle fast motion without increasing the size of the window N and the computation time. Since no convolution transformation is applied, we can use it to align h_{t-1} and $\hat{r}_{t-2:t+2}$ to y_t much faster than other techniques. Our approximation only adds two extra convolution layers, while optical-flow calculation [14] requires an entire sub-network with several convolutions.

III. Experimental Results

The training dataset was constructed by extracting 10^6 sequences of 6 patches of size 128×128 pixels from the Myanmar training sequences. For each HR patch sequence we obtain its corresponding 10 LR patch sequence, so each HR patch at time t has the corresponding LR sequence of patches at time $t-2$, $t-1$, t, $t+1$, and $t+2$. To remove uninformative patches from our training dataset, patches with variance less than 0.0035 were not considered.

We train the network for 60 epochs using a batch size of 64 and sampling 10000 batches per epoch. For our recurrent networks, we first pre-train the auxiliary network g_ϕ for 10 epochs. The loss we use to train our network is, instead of the Mean Squared Error (MSE), the Charbonnier loss:

$$\gamma(\hat{x}, x) = \sum_k \sum_i \sum_j \sqrt{(\hat{x}_{k,i,j} - x_{k,i,j})^2 + \epsilon^2}\ , \tag{6}$$

where $\hat{x}$ and x are the estimated and the real high-res frames respectively and ϵ an hyperparameter, which in our experiments is set to 10^{-3}. This loss is more robust to outliers and more stable than MSE [21]. We use Adam optimizer [22] with the learning rate set to 10^{-3} for the first 20 epochs and then divided by 10 at the 20th and 40th epoch. The weight decay

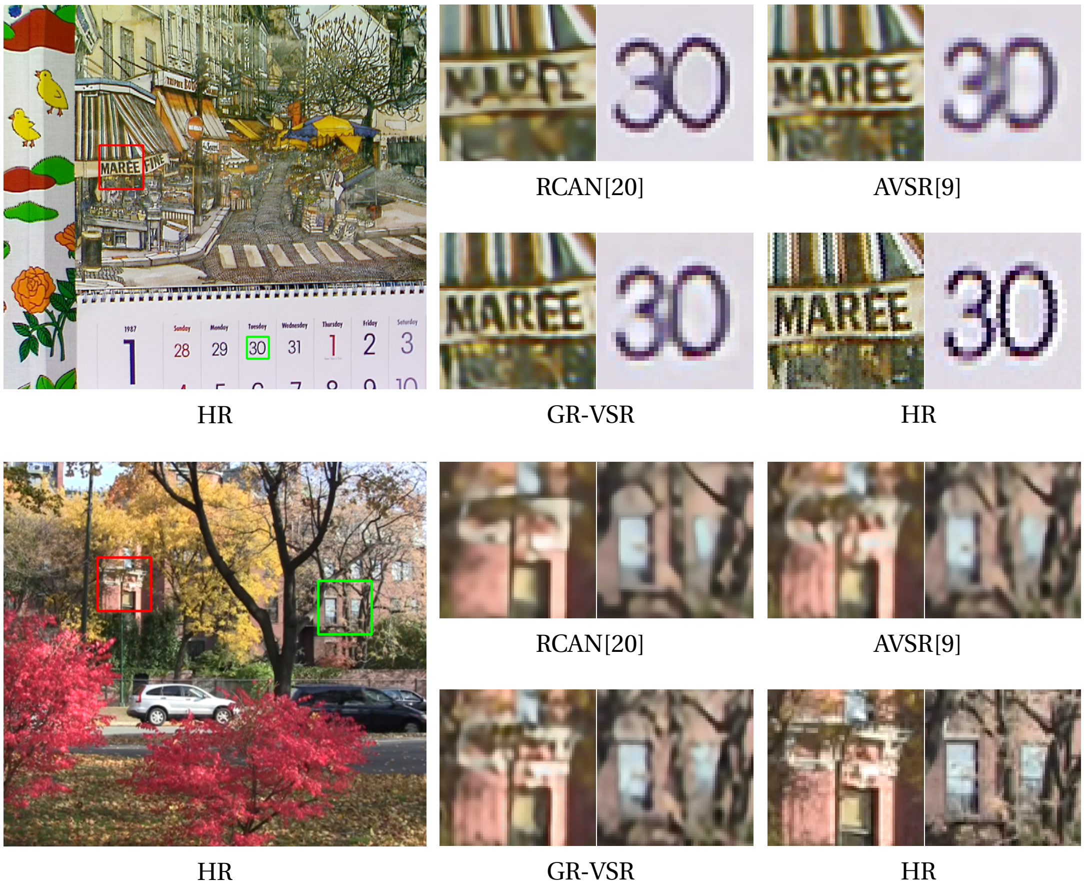


Fig. 2: Qualitative results of our GR-VSR model compared to current state-of-the-art methods for factor 4. In this case, GR-VSR is able to recover more details of the original high-res frame than the other non-recurrent methods. However, the results obtained by GR-VSR are less noisy and closer to the original HR image.

parameter was set to 10^{-4}. We focus on upscaling factor 4 for all the experiments shown in this section.

To determine the contribution of each of the components of our proposed architecture GR-VSR, we perform an ablation study. First, we train a non-recurrent model without deformable attention but with the same depth as our model. We call this model No-R-VSR. This model is similar to VSRResNet but with half the number of residual blocks and two-step upsampling using a subpixel shuffle layer [23]. To check the contribution of the recursion, we add recursion to No-R-VSR creating a new model R-VSR. Notice that R-VSR uses neither deformable attention nor a gate. The model R-VSR-Att incorporates deformable attention to R-VSR. Finally, the model GR-VSR-No-Att calculates the h_t using the gate network without deformable attention.

The first part of Table I shows the results of this study. All models are compared in terms of PSNR, SSIM and T-RRED[24]. The T-RRED metric is used to quantitatively measure temporal consistency (the lower the better). The most significant increase in the spatial quality metrics comes from the inclusion of the recursion, since R-VSR outperforms No-R-VSR by 0.8 dB with almost the same computational cost. As expected, the inclusion of deformable attention in R-VSR-Att further boosts the performance of the network. However, this is not the case when using only the gate network: we expected GR-VSR-No-Att to perform worse than R-VSR, since without aligning moving objects in h_{t-1} and $\hat{h}_t$ have different space locations. Despite this, both GR-VSR-No-Att and R-VSR-Att have a similar performance. This indicates that the gate network in GR-VSR-No-Att has learned to only use h_{t-1} information for objects that did not change location in the next time step. Finally, the gate network when used together with deformable attention can significantly increase both the quality of the frames and the temporal consistency, as shown by the difference between our full model GR-VSR and R-VSR-Att.

We now compare our GR-VSR model with the state-of-art. The second part of Table I shows this comparison. Notice that, in the case of FRVSR[14], the results are not directly comparable since the authors use a degradation different from bicubic downsampling. It can be seen that our GR-VSR significantly outperforms for all figures of merit all the other state-of-art methods for bicubic downsampling degradation, even AVSR[9] that uses a much deeper network. This difference can be observed in the predicted frames, as shown in Fig. 2. In the case of FRVSR[14], even though our proposed network performs far less operations per time step (8 residual blocks with 64 filters each vs 10 residual blocks with 128 filters each), we obtain comparable results. It is worth noticing that the degradation used in [14] is less aggressive than bicubic downsampling, so we can expect that our model will still sightly outperform FRVSR[14] when applied to the same degradation.

TABLE I: Results on the VidSet4 video sequences [25] for a scale factor of four in terms of PSNR, SSIM and T-RRED[24]. The first part of the table shows the ablation study for the proposed GR-VSR and the second part a comparison with current state-of-art methods. Notice that RCAN[20] is a SR method. VESPCN[4], TAN[6] and FRVSR[14] results are the ones reported in the original paper. FRVSR[14] uses a different degradation (not bicubic downsampling).

	↑ PSNR	↑ SSIM	↓ T-RRED[24]
No-R-VSR	25.77	0.7679	1.42
R-VSR	26.61	0.8115	1.37
R-VSR-Att	26.68	0.8145	1.34
GR-VSR-No-Att	26.61	0.8123	1.36
GR-VSR	**26.76**	0.8158	**1.27**
RCAN[20]	25.44	0.7405	1.71
VESPCN[4]*	25.35	0.7309	-
SPMC-SR[5]	25.63	0.7709	1.74
TAN[6]*	25.53	0.7475	-
VSRResNet[8]	25.51	0.7530	1.47
FRVSR[14]*	26.69	**0.822**	-
AVSR[9]	26.17	0.7895	1.30

IV. Conclusions

We have introduced a new RCNN VSR model that adapts the recurrent unit GRU. Our model uses deformable attention to align the previous hidden state with the current one and a gated operation to combine them. This allows our model to better reuse features and exploit longer time relationships between the frames. The experimecnts show that our model, GR-VSR, outperforms current state of the art methods in terms of PSNR, SSIM and temporal consistency. Temporal consistency is naturally achieved, without the use of losses that explicitly impose temporal consistency. In the future, perceptual losses will be incorporated into the training of this model to further increase the perceptual quality of the predicted frames.

References

[1] S. D. Babacan, R. Molina, and A. K. Katsaggelos, "Variational Bayesian super resolution," *IEEE Transactions on Image Processing*, vol. 20, no. 4, pp. 984–999, 2011.

[2] S. P. Belekos, N. P. Galatsanos, and A. K. Katsaggelos, "Maximum a posteriori video super-resolution using a new multichannel image prior," *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1451–1464, 2010.

[3] C. Liu and D. Sun, "On Bayesian adaptive video super resolution," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 2, pp. 346–360, 2014.

[4] J. Caballero, C. Ledig, A. Aitken, A. Acosta, J. Totz, Z. Wang, and W. Shi, "Real-time video super-resolution with spatio-temporal networks and motion compensation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 07 2017, pp. 2848–2857.

[5] X. Tao, H. Gao, R. Liao, J. Wang, and J. Jia, "Detail-revealing deep video super-resolution," in *IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[6] D. Liu, Z. Wang, Y. Fan, X. Liu, Z. Wang, S. Chang, and T. Huang, "Robust video super-resolution with learned temporal dynamics," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2507–2515.

[7] A. Kappeler, S. Yoo, Q. Dai, and A. K. Katsaggelos, "Video super-resolution with convolutional neural networks," *IEEE Transactions on Computational Imaging*, vol. 2, no. 2, pp. 109–122, 2016.

[8] A. Lucas, S. López-Tapia, R. Molina, and A. K. Katsaggelos, "Generative adversarial networks and perceptual losses for video super-resolution," *IEEE Transactions on Image Processing*, vol. 28, no. 7, pp. 3312–3327, July 2019.

[9] S. Lopez-Tapia, A. Lucas, R. Molina, and A. K. Katsaggelos, "Multiple-degradation video super-resolution with direct inversion of the low-resolution formation model," in *2019 27th European Signal Processing Conference (EUSIPCO)*, Sep. 2019, pp. 1–5.

[10] D. Nilsson and C. Sminchisescu, "Semantic video segmentation by gated recurrent flow propagation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[11] A. Pfeuffer, K. Schulz, and K. Dietmayer, "Semantic segmentation of video sequences with convolutional lstms," in *2019 IEEE Intelligent Vehicles Symposium (IV)*, June 2019, pp. 1441–1447.

[12] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv*, 2014.

[13] Y. Huang, W. Wang, and L. Wang, "Video super-resolution via bidirectional recurrent convolutional networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 1015–1028, April 2018.

[14] M. S. M. Sajjadi, R. Vemulapalli, and M. Brown, "Frame-Recurrent Video Super-Resolution," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[15] X. Zhu, H. Hu, S. Lin, and J. Dai, "Deformable convnets v2: More deformable, better results," *arXiv preprint arXiv:1811.11168*, 2018.

[16] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced deep residual networks for single image super-resolution," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, vol. 1, no. 2, 2017, p. 3.

[17] J. C. Heck and F. M. Salem, "Simplified minimal gated unit variations for recurrent neural networks," in *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*, Aug 2017, pp. 1593–1596.

[18] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," *arXiv preprint arXiv:1703.06211*, 2017.

[19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17. Red Hook, NY, USA: Curran Associates Inc., 2017, p. 6000–6010.

[20] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *European Conference on Computer Vision (ECCV)*, 2018.

[21] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Deep laplacian pyramid networks for fast and accurate super-resolution," in *IEEE Conferene on Computer Vision and Pattern Recognition*, 2017.

[22] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980

[23] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1874–1883, 2016.

[24] R. Soundararajan and A. C. Bovik, "Video quality assessment by reduced reference spatio-temporal entropic differencing," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 4, pp. 684–694, April 2013.

[25] C. Liu and D. Sun, "A bayesian approach to adaptive video super resolution," in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on.* IEEE, 2011, pp. 209–216.

2.3 Collaborations

In this section, we present some works in which the Ph.D. candidate has had a relevant role in their elaboration but is not the main author. Since they are not part of this dissertation, they will only be mentioned with their relevant contributions.

2.3.1 Generative Adversarial Networks and Perceptual Losses for VSR

Publication: Alice Lucas, Santiago López-Tapia, Rafael Molina, and Aggelos K. Katsaggelos. Generative Adversarial Networks and Perceptual Losses for Video Super-Resolution. *IEEE Transactions on Image Processing* 28 (7): 3312-3327, 2019 (Impact Factor (JCR 2019): 9.340, Rank: 11/266 (Q1 and D1) in Engineering, Electrical and Electronic). Preliminary results were presented in [131] (GGS Rating: A-, GGS Class: 2, CORE: B).

Contributions:

- A new CNN-based architecture is proposed for VSR. Compared to other CNN models for VSR, this is the first one to incorporate the use of residual blocks [24] to obtain a very deep architecture.

- To increase the perceptual quality of the produced frames, we introduce feature-based losses and GANs to the VSR problem.

- We perform several experiments on standard VSR dataset Myanmar [126] with different architecture configurations (number of layers, input frames, ...), hyper-parameters and raw or motion-compensated frames. The results show that the use of feature-based losses in a GAN framework greatly increases the perceptual quality of the SR frames. However, the produced frames present some high-frequency noise and artifacts that can be easy to spot in low activity places.

2.3.2 Self-supervised Fine-tuning for Correcting Super-Resolution Convolutional Neural Networks

Publication: Alice Lucas, Santiago López-Tapia, Rafael Molina, and Aggelos K. Katsaggelos. Self-supervised Fine-tuning for Correcting Super-Resolution Convolutional Neural Networks. *IEEE Transactions on Image Processing*, 2020 (Submitted) (Impact Factor (JCR 2019): 9.340, Rank: 11/266 (Q1 and D1) in Engineering, Electrical and Electronic). Preliminary results were presented in [132] (GGS Rating: A-, GGS Class: 2, CORE: B).

Contributions:

- We introduce a new method that allows us to fine-tune a trained SR/VSR CNN using a single LR image by optimizing a loss function based on the image formation model.

- This technique can be used to remove artifacts introduced by a GAN model and to perform SR/VSR on samples degraded with a different degradation than the one used to train the model.
- We perform an extensive experimental study on different scenarios using standard SR and VSR test sets. These experiments show that our proposal is competitive with current state-of-art methods.

Chapter 3

Image Restoration and Super-Resolution

3.1 Combining Analytical and Deep Learning Methods in Blind Image Deconvolution

3.1.1 Publication details

Authors: Santiago López-Tapia, Javier Mateos, Rafael Molina, and Aggelos K. Katsaggelos.
Title: Combining Analytical and Deep Learning Methods in Blind Image Deconvolution.
Publication: IEEE Transactions on Image Processing, 2020.
Status: Under review.
Quality indices:

- Impact Factor (JCR 2019): 9.340
- Rank: 11/266 (Q1 and D1) in Engineering, Electrical and Electronic

3.1.2 Main Contributions

- In this work, we propose a fast, accurate and robust blind image deconvolution model that combines analytical and deep learning approaches. It uses a sequential scheme; it first estimates the blur kernel using an analytical model and then it uses a CNN to restore the image.

- We show that a Wiener filter reconstruction can approximate the Moore-Penrose pseudo-inverse reconstruction of the original image. This image, together with the observed image, constitutes the input to our CNN.

- Our model obtains the estimated image as a linear combination of the Wiener filter reconstruction and a residual image, which is obtained as a linear transformation of the network output. Using this approximation, we translate the CNN task from a deconvolution one into a denoising one where a CNN removes ringing and other artifacts in the Wiener filtered image. This allows us to better separate the image formation model from the network parameters' learning, overcoming the poor performance of deconvolution CNNs with blurs not seen during training.

- Finally, we remove artifacts in the restored image caused by inaccuracies in estimating the blur and other image formation model discrepancies using the kernels predicted by a Dynamic Filtering Network [133].

- We perform experiments on standard blind image deconvolution datasets and compare them with current state-of-art in terms of PSNR, SSIM and computation time.

Combining Analytical and Deep Learning Methods in Blind Image Deconvolution

Santiago López-Tapia *, Javier Mateos*, Rafael Molina*, Aggelos K. Katsaggelos†
*Dept. of Computer Science and Artificial Intelligence, University of Granada, Granada, Spain
Email: {sltapia, jmd, rms}@decsai.ugr.es
†Dept. of Electrical Engineering and Computer Science Northwestern University
Evanston, IL, USA
Email: aggk@eecs.northwestern.edu

Abstract

In this paper we propose a fast, accurate, and robust blind image deconvolution model. The model is based on the combination of analytical and deep learning recovering approaches. It employs a sequential scheme, first estimates the blur kernel using an analytical model and then obtains the restored image using a robust deep learning one. The input to the network is, together with the observed image, an approximation to the Moore-Penrose pseudo-inverse reconstruction of the original image. This approximation is shown to correspond to a Wiener filter reconstruction. The estimated image is a linear combination of the Wiener filter reconstruction and a residual image which is obtained as a linear transformation of a network output. This allows us to better separate the image formation model from the learning of the network parameters and thus to allow the network to generalize to blurs not present in the training process. By using this representation of the estimated image, we cast our problem into a denoising one where ringing and other artifacts in the Wiener filtered image are removed by a Convolutional Neural Network. To remove additional artifacts caused by inaccuracies in the blur estimation and other image formation model discrepancies like saturated and underexposed areas, a further improvement of the reconstructed image is obtained with the use of the kernels predicted by a Dynamic Filter Network. The extensive experiments, carried out on several synthetic and real image datasets, assert the performance and robustness of the proposed method and demonstrate the advantage of the proposed method over existing ones.

Index Terms

Blind image deconvolution, deep learning, convolutional neural network, affine projection.

I. Introduction

Mathematically, Blind Image Deconvolution (BID) is the problem of restoring an image $\mathbf{x}$ from its blurred and noisy version $\mathbf{y}$ when the blur $\mathbf{H}$ is unknown [1]. Generally, the image $\mathbf{y}$ is

This work was supported in part by the Spanish Ministerio de Economía y Competitividad under contract DPI2016-77869-C2-2-R and the Visiting Scholar Program at the University of Granada. Santiago López-Tapia received financial support through the Spanish FPU program.

modeled as

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}\,, \tag{1}$$

where $\mathbf{n}$ is the noise. Both $\mathbf{y}$ and $\mathbf{x}$, as well as $\mathbf{n}$, are $N \times 1$ vectors representing lexicographically ordered corresponding images, and $\mathbf{H}$ represents an $N \times N$ matrix. In this paper we assume that $\mathbf{H}$ is a spatially invariant two-dimensional convolution operator of unknown point spread function (PSF), $\mathbf{h}$, of size $M \times 1$.

Analytical techniques for solving BID problems have been studied for a long time. When using an analytical approach, the forward model is explicitly described, the criteria for obtaining a solution are decided, and an optimization procedure is chosen. At the high level, one can group analytical techniques into deterministic and stochastic ones. Within the first class, an optimization criterion is typically chosen, such as the minimization of the l_2 error norm $\parallel \mathbf{y} - \mathbf{H}\mathbf{x} \parallel^2$. Then, prior (or domain) knowledge is incorporated into the solution process through regularization. That is, additional terms are included in the optimization functional imposing, for example, smoothness or sparsity on the restored image as well as the blur estimate. With stochastic approaches, the unknowns are treated as stochastic quantities and then a maximum likelihood, or a maximum *a posteriori* (MAP) or a fully (hierarchical) Bayesian approach is followed (see [1] for a review). In the latter case an estimate of the full posterior $\mathrm{p}(\mathbf{x}, \mathbf{h}|\mathbf{y})$ is obtained. This posterior is usually approximated by the product of distributions $\mathrm{q}(\mathbf{x})\mathrm{q}(\mathbf{h})$. Over the years, carefully designed image priors were based on constraints on the image gradients while flat priors were used for the blur, see however [2]. Some examples of such priors are hyper-Laplacian priors [3], [4], log-TV priors [5], mixture of Gaussians (MoG) [6], Super Gaussian (SG) [7], Scale Mixture of Gaussian (SMG) [8], and generalized ℓ_p/ℓ_q norm-based priors [9]. Approximations of the l_0 function (see [10], extended with the use of a dark channel prior [11] and an extreme channel prior [12]) have also been successfully used. The previously described prior models are frequently used by analytical methods to calculate the blur and latent image either in an alternate or a sequential fashion. The first approach (see, for instance, [1], [5], [4]) alternates between the estimation of the blur and the latent image in an iterative optimization process while the second one (see, for instance, [6], [10], [2], [11], [7], [12], [9]), first estimates the blur using a filtered version of the image and then uses this estimation with a non blind deconvolution algorithm to recover the image.

The above described methods assume that the model in (1) is accurate for all image pixels. However, the convolution model can be violated locally by saturated pixels, underexposed areas or salt and pepper noise and methods to handle this kind of outliers have been proposed (see, for instance, [13], [14], [15] and references therein). Another source of outliers is kernel inaccuracy. Blur estimation algorithms usually estimate an inaccurate kernel that can produce severe artifacts such as ringing and distortions [16] even for small errors in the kernel. Kernel inaccuracy has been modeled as zero-mean white noise added to the kernel [17]. In [18] a residual term is added to (1) to model the errors produced by the kernel in the image. Then, a sparse prior is imposed on this residual and estimates of both the original image and the residual are obtained. Although artifacts are well suppressed by this method, details are usually over-smoothed.

Analytical techniques need to optimize an energy function for each new image and blur. This provides them with high flexibility, allowing them to adapt to a wide variety of blurs, but at a high computational cost. Unfortunately, they are not amortized procedures. There is no general fast procedure that given an observed image produces the corresponding original image and blur. In recent years, the fields of machine and Deep Learning (DL) have gained a lot of momentum in solving inverse problems [19]. Unlike analytical methods for which the problem is explicitly defined and domain-knowledge carefully is engineered into the solution, Deep Neural Networks (DNNs) do not benefit from such prior knowledge and instead make use of large data sets to learn for each $\mathbf{y}$ the unknown solution to the inverse problem. Works on image denoising [20], inpainting [21] and superresolution [22] have shown that these methods can outperform analytical ones, while being significantly faster. For image recovering problems with known linear degradation models like image denoising, non-blind image deconvolution, super resolution, and compressive sensing, variable splitting techniques such as alternating direction method of multipliers (ADMM) [23] and half-quadratic splitting (HQS) [24] offer a sound model to combine analytical and DL techniques. Using these techniques, the recovery problem is split into two subproblems [25]: a regularized recovery one (*subproblem A*) which uses as penalty the squared Euclidean distance to an image. This image is estimated using an appropriate denoising technique (*subproblem B*). This splitting has some advantages: subproblems A and B are easier to solve than the original one: for shift invariant degradation, subproblem A can easily be solved using Discrete Fourier Transform (DFT), while for subproblem B we can select any denoising algorithm. Instead of hand-engineering the prior term, deep neural networks have been recently proposed to solve the denoising problem. In [26] and [27] the proximal operator [28] associated to subproblem B is replaced by a denoising neural network. In a similar way, [29] uses a residual network to estimate the noise in subproblem B, which is, then, subtracted from the currently estimated image to obtain the restored one. In [30] a combination of adversarial learning and a denoising autoencoder is chosen by combining a denoiser with a Generative Adversarial Network (GAN) to project the images into the space of the natural images as proximal operator. In [31] the authors opted for a denoising autoencoder network inspired by the U-Net architecture. For non-blind image deconvolution, RGDN [32] integrates both subproblems into a recurrent convolutional network, where the gradient of the image prior unit is replaced by a common CNN block. The combination of analytical and DL methods using this approach allows for high flexibility and efficiency. However, these methods do not provide a fully amortized approach, since for each new degradation the solutions to subproblems B and A (with the new degradation) must be found iteratively.

The use of DL models for blind image deconvolution took longer to take off since, together with the image, the network must be able to estimate the blur as well. The first proposed models deal with specific types of blur, like motion blur or Gaussian blur. The authors of [33] use a Convolutional Neural Network (CNN) to predict a motion vector per pixel and, then, an analytical method to estimate the underlying image. In [34], three parametric blur models are estimated using a General Regression Neural Network (GRNN) together with an additional CNN which is used to select one of the three model for a given observed image. More recent models are not constrained to the type of blur. In [35] a CNN is used to learn features that improve the estimation of both blurring kernel and image in the Fourier space. Following this approach, in [36] a CNN directly predicts several deconvolution filters in the Fourier space. These filters are subsequently combined to obtain the original image. In [37] a deep CNN is used to learn a discriminative regularizer.

This regularizer provides an output in the interval $[0, 1]$ which distinguishes between clear (0) and blurred images (1). Two additional penalty terms are used for BID, the ℓ_0-pseudonorm of the original image and the ℓ_2 norm of the blur. The image and blur which optimize the sum of the fidelity to the observation term plus weighted versions of the three described regularizers are found using ADMM [23]. The authors of [38] address the issue of catastrophic forgetting [39], that is, the tendency of neural networks of completely and abruptly forgetting previously learned information upon learning new one. They adopt a vanilla residual CNN to make an aggressive propagation towards the optimum. Another approach consists replacing the prior model by a deep CNN that generates the output image. Following this approach, [40] extends to image deconvolution the deep image prior in [41]. In [42] a generalization of the traditional iterative total-variation regularization method in the gradient domain is subsequently unrolled to construct a neural network. Notice that, with the exception of [33], [34], all described models use an alternate approach to estimate the blur and the latent image.

Recently, the authors of [43] have proposed a sequential approach to combine both analytical and DL models which is robust to inaccuracies in the blur estimation. A CNN is fed with the (inaccurate) blur, previously estimated by a blind image deconvolution algorithm, and several estimates of the latent image obtained using the method in [3] with different prior strengths. Those image estimates provide complementary information that the network combines into the restored image.

In contrast to the large variety of BID methods combining analytical and DL approaches, few papers apply only DL constrained to specific blur types, like motion blur, without explicitly estimating the blur. In [44] a multi-scale residual CNN is used to eliminate non-uniform motion blur from images. The use of a GAN is proposed in [45] and enhanced in [46]. For video, a spatio-temporal recurrent architecture is presented in [47] (see also [48] where the authors use a convolutional recurrent network to eliminate non-uniform motion blur in video sequences). Although these models obtain state-of-art results in far less time than analytical and combined models, they lack flexibly thus limiting their applicability to only specific cases.

In this paper we propose a fast, accurate, and robust blind image deconvolution model. It employs a sequential scheme; it first estimates the blur kernel using an analytical model and then it obtains the restored image using a robust deep learning one. In contrast to [43], in which DL is used to fuse several noisy image estimates, the input to the network is, both the observed image and an approximation to the Moore-Penrose pseudo-inverse reconstruction of the original image. This approximation is shown to correspond to a Wiener filter reconstruction. The estimated image is a linear combination of the Wiener filter reconstruction and a residual image which is obtained as a linear transformation of a network output. This allows us to better separate the image formation model from the learning of the network parameters and thus to allow the network to generalize to blurs not present in the training process. By using this representation of the estimated image, we cast our problem into a denoising one where ringing and other artifacts in the Wiener filtered image are removed by a CNN. To remove artifacts caused by inaccuracies in the estimated blur and other image formation model discrepancies like saturated and underexposed areas, we use the kernels predicted by a Dynamic Filter Network (DFN) [49].

The rest of the paper is organized as follows: in section II, the used notation and the proposed

DL approach is presented. This model depends on an estimation of the blur that is obtained by the analytical method described in section III. Section IV describes the network training procedure and an ablation experiment is performed in Sect. V to determine the contribution of each component of the proposed CNN model. The performance of the proposed method is tested and compared with other classical and state-of-the-art deconvolution methods in Sect. VI. Finally, Sect. VII concludes the paper.

II. Deep learning model

As previously stated, BID is a very challenging task for analytical and deep learning models due to the variability of the degradations and the ill-posed nature of the problem. Even if the blur is exactly known, noise in the data results in large perturbations in the solution. While analytical BID methods estimate image and blur either sequentially or in an alternate manner and use the image formation model as well as prior knowledge to solve the problem, the task of DL is more daunting. DL methods which use the image formation model explicitly, see (1), become blur dependent to poorly generalize to other blurs. To ease the problem, the image formation model has to be separated from the learning of the network parameters.

In [50] the authors consider the image Super Resolution (SR) problem where the observed downsampled image $\mathbf{v}$ is related to the high resolution one $\mathbf{s}$ by

$$\mathbf{v} = \mathbf{A}\mathbf{s}, \tag{2}$$

where $\mathbf{A}$ is a downsampling operator and no noise is assumed to be present. The authors transform a risk based approach into a Maximum A Posteriori (MAP) estimation by defining the SR solution as

$$\mathrm{c}_\omega(\mathbf{v}) = (\mathbf{I} - \mathbf{A}^+\mathbf{A})\mathrm{b}_\omega(\mathbf{v}) + \mathbf{A}^+\mathbf{v}, \tag{3}$$

where $\mathbf{A}^+$ is the Moore-Penrose pseudo-inverse [51] of $\mathbf{A}$ and $\mathrm{b}_\omega(\cdot)$ denotes a network whose parameters ω are to be learned. The authors note that

$$\mathbf{A}\mathrm{c}_\omega(\mathbf{v}) = \mathbf{A}(\mathbf{I} - \mathbf{A}^+\mathbf{A})\mathrm{b}_\omega(\mathbf{v}) + \mathbf{A}\mathbf{A}^+\mathbf{v} = \mathbf{0} + \mathbf{v}, \tag{4}$$

since $\mathbf{A}\mathbf{A}^+\mathbf{A} = \mathbf{A}$ and $\mathbf{A}$ is a full row rank matrix, i.e. $\mathbf{A}\mathbf{A}^+ = \mathbf{I}$. Then they use (3) to define a probability distribution on $\mathbf{s}$ which is learned with the use of a GAN. Notice that the input to the generator is not noise but $\mathbf{v}$. The authors call their approach amortized MAP because the generator produces a MAP solution which, for a given $\mathbf{A}$ and $\mathbf{v}$, provides a fast and amortized procedure to obtain $\mathbf{s}$. Unfortunately, changing $\mathbf{A}$ requires to reestimate the generator. However, for methods formulated within the analytical context it is even worse: for each $\mathbf{A}$ and $\mathbf{v}$ a complex estimation procedure for the corresponding $\mathbf{s}$ must be derived (no general -amortized- generator of SR images is provided).

In [52], [53] we extended the above modelling to the Video Super Resolution (VSR) problem. Noise was added to the image formation process in (2). We designed an amortized GAN model for even varying $\mathbf{A}$, that is, the network can be used on different $\mathbf{A}$s and $\mathbf{v}$s. In (3), $\mathbf{A}^+\mathbf{v}$ can be seen as an initial approximation of the high resolution image. Thus, the network only has to learn a residual image. Since this residual depends on $\mathbf{A}^+\mathbf{v}$, an amortized model for varying $\mathbf{A}$ can be obtained by modifying the input of the network to include $\mathbf{A}^{+}\mathbf{v}$ and using the proper architecture. Notice that the MAP interpretation is lost due to the addition of noise to the modelling and the inclusion of additional constraints to learn the generator. See [52], [53] for details.

Let us now examine, how to adapt the above ideas to BID. For the time being we assume that we have access to $\mathbf{H}$, its estimation is deferred to section III. Let us consider the image formation model in (1) and let $\mathbf{H}^{+}$ be the Moore-Penrose pseudo-inverse of the blur $\mathbf{H}$. Let us denote by $\mathrm{f}_\theta : \mathcal{R}^L \rightarrow \mathcal{R}^N$ a CNN with parameters θ, whose input $\mathbf{z}$ will be clearly specified later. We can then define the model

$$\mathrm{g}_\theta(\mathbf{z}) = (\mathbf{I} - \mathbf{H}^{+}\mathbf{H})\mathrm{f}_\theta(\mathbf{z}) + \mathbf{H}^{+}\mathbf{y}. \tag{5}$$

Note that, since $\mathbf{H}\mathbf{H}^{+}\mathbf{H} = \mathbf{H}$, we have

$$\mathbf{H}\mathrm{g}_\theta(\mathbf{z}) = \mathbf{H}(\mathbf{I} - \mathbf{H}^{+}\mathbf{H})\mathrm{f}_\theta(\mathbf{z}) + \mathbf{H}\mathbf{H}^{+}\mathbf{y} = \mathbf{H}\mathbf{H}^{+}\mathbf{y}. \tag{6}$$

Unfortunately, since $\mathbf{H}$ is not a full row rank matrix, unlike in SR and VSR, $\mathbf{H}\mathbf{H}^{+} \neq \mathbf{I}$. It can be shown [54] that

$$\mathbf{H}^{+} = \lim_{\delta \to 0^{+}} (\mathbf{H}^{\mathbf{T}}\mathbf{H} + \delta\mathbf{I})^{-1}\mathbf{H}^{\mathbf{T}} \tag{7}$$

and so, for the i-th Discrete Fourier frequency, we have

$$\mathcal{F}_{\mathbf{H}\mathbf{H}^{+}\mathbf{y}}(i) = \begin{cases} \mathcal{F}_{\mathbf{y}}(i) & \text{if } \ \mathcal{F}_{\mathbf{H}}(i) \neq 0 \\ 0 & \text{if } \ \mathcal{F}_{\mathbf{H}}(i) = 0 \end{cases} \tag{8}$$

Note also that $\mathbf{H}^{+}\mathbf{y}$ can be considered a rough approximation of the original image $\mathbf{x}$. Hence, using this formulation, by minimizing, for instance, $\parallel \mathbf{x} - \mathrm{g}_\theta(\mathbf{z}) \parallel^2$ the network $\mathrm{f}_\theta(\cdot)$ has to only learn a residual that when added to $\mathbf{H}^{+}\mathbf{y}$ results in a sharp, artifacts-free image. Furthermore, since

$$\begin{aligned} \frac{1}{N} \parallel \mathbf{y} - \mathbf{H}\mathrm{g}_\theta(\mathbf{z}) \parallel^2 &\leq \frac{1}{N}(\parallel \mathbf{H}\mathbf{x} - \mathbf{H}\mathrm{g}_\theta(\mathbf{z}) \parallel^2 + \parallel \mathbf{n} \parallel^2) \\ &\leq \frac{1}{N}(\parallel \mathbf{H} \parallel^2 \parallel \mathbf{x} - \mathrm{g}_\theta(\mathbf{z}) \parallel^2 + \parallel \mathbf{n} \parallel^2) \\ &\leq \parallel \mathbf{x} - \mathrm{g}_\theta(\mathbf{z}) \parallel^2 + \frac{\parallel \mathbf{n} \parallel^2}{N}, \end{aligned} \tag{9}$$

by reducing $\parallel \mathbf{x} - \mathrm{g}_\theta(\mathbf{z}) \parallel^2$ we also reduce the data fidelity error. Notice that, since we have separated the degradation from the learning of the network parameters, with the proper network design, we can deal with different degradation models. Notice also that we are making an implicit use of the observation model. This restoration approach constrains the functions that the network learns to only those that produce images consistent with the observation $\mathbf{y}$.

Unfortunately, for the deconvolution problem at hand, the presence of noise and the fact that the blur is unknown hamper the use of the model in (5) as is (notice that we will have to use an estimate of the blur, the process of which has not been described yet). Furthermore, from (8) we observe that in $\mathbf{H}^{+}\mathbf{y}$ the information at those frequencies for which $\mathbf{H}$ is zero has been lost. For those reasons, we define

$$\mathbf{H}_\epsilon^{+} = (\mathbf{H}^{T}\mathbf{H} + \epsilon\mathbf{I})^{-1}\mathbf{H}^{\mathbf{T}} \tag{10}$$

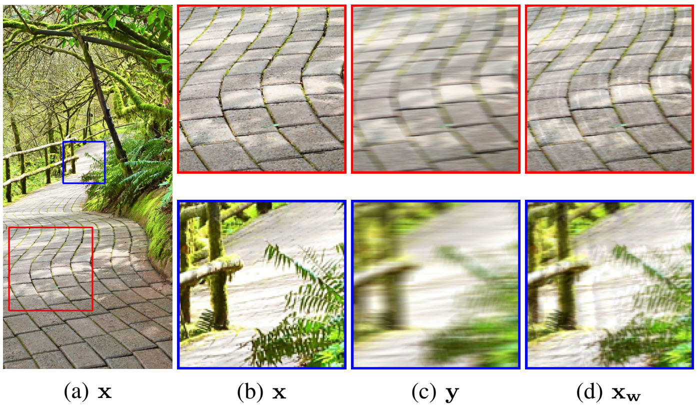


Fig. 1. An illustration of the artifacts caused by the Wiener filter. Ringing artifacts in the red box are compatible with natural images, being difficult to distinguish from real structures, unlike the ones in the blue box.

and

$$\mathbf{x_w} = \mathbf{H}_\epsilon^+ \mathbf{y}, \tag{11}$$

where $\epsilon > 0$ is a regularization parameter. Then, we modify (5) to finally write

$$\begin{aligned} \mathrm{g}_\theta(\mathbf{z}) &= (\mathbf{I} - \mathbf{H}_\epsilon^+ \mathbf{H})\mathrm{f}_\theta(\mathbf{z}) + \mathbf{H}_\epsilon^+ \mathbf{y} \\ &= (\mathbf{I} - \mathbf{H}_\epsilon^+ \mathbf{H})\mathrm{f}_\theta(\mathbf{z}) + \mathbf{x_w}. \end{aligned} \tag{12}$$

We now have an interpretation of the goal of the network. Taking into account that $\mathbf{x_w}$ is an approximation of the Wiener filter (ϵ represents an approximation of the ratio between the power spectral density of the noise and the power spectral density of the image), it learns the “missing” frequencies in $\mathbf{x_w}$ and how to remove the noise and the artifacts introduced by the Wiener filter approximation (for simplicity we use the term Wiener filter instead of Wiener filter approximation through out the paper). These artifacts, although dependent on the image, blur, and noise combination, are all very similar in nature, consisting mainly on amplified noise and ringing. Focusing on detecting and filtering these artifacts is easier than removing blur and makes the network able to better generalize for blurs unseen during training. Furthermore, note that $\mathbf{x_w}$, although noisy, contains information on the high frequencies of the original image $\mathbf{x}$ which makes the recovery task easier.

Let us now see what the input to the network is. The function $\mathrm{f}_\theta(\cdot)$ (and consequently $\mathrm{g}_\theta(\cdot)$) takes as input both $\mathbf{x_w}$ and $\mathbf{y}$, in other words,

$$\mathbf{z} = (\mathbf{x_w}, \mathbf{y}). \tag{13}$$

The use of the blurred image $\mathbf{y}$ is necessary since some artifacts that appear in the Wiener filter solution, $\mathbf{x_w}$, are difficult to distinguish from real world-like structures, making them difficult to remove. Figure 1 presents an example of these artifacts. Incorporating $\mathbf{y}$ as an input helps the network distinguish between artifacts and real scene objects, thus removing structures not present in $\mathbf{y}$ and preserving the consistency with the observation.

Since the image formation model in (1) is only approximate since it does not include, for instance, saturated pixels or underexposed areas, the deconvolved image obtained by the operator

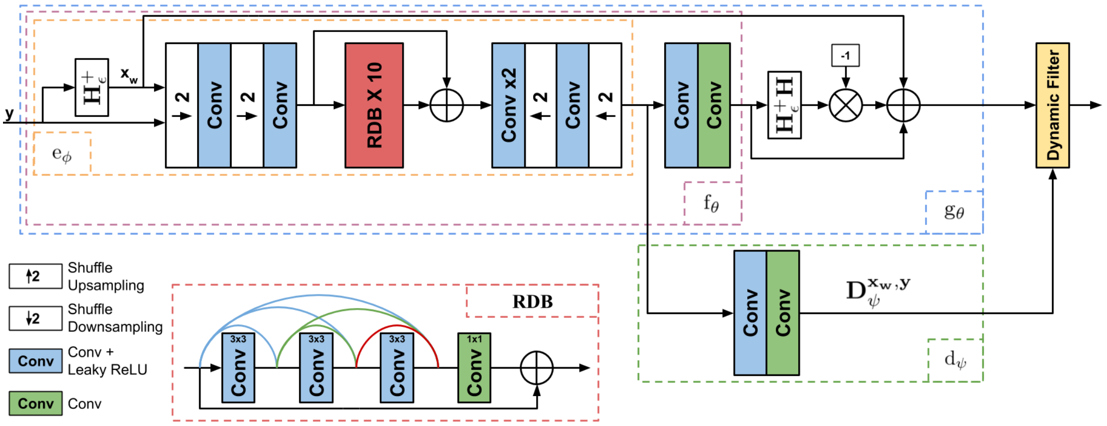


Fig. 2. The proposed DDNet architecture. RDB indicates a Residual Dense Block (RDB) and Dynamic Filter refers to the operation in (15). Each non-output convolution outside a RDB has 64 × 3 × 3 filters. With the exception of the last one, that uses 64 × 1 × 1 filters, convolutions in RDBs have 32 × 3 × 3 filters. The size of the filters used during the dynamic filtering is 5 × 5. See text for details.

$\mathrm{g}_\theta(\cdot)$ still exhibits some spatially variant artifacts. To remove them we propose to use a Dynamic Filter Network (DFN) [49]. The DFN is composed by a filter generating network and a dynamic filtering layer. The former dynamically generates filters which depend on the network input values and pixel position. The latter then applies those filters to the input.

For a pixel (i, j), let $\mathbf{d}_\psi^{\mathbf{u}_{i,j}}$ be the filter of support $(2L + 1) \times (2M + 1)$ generated by the filter generating network with parameters ψ from the input image $\mathbf{u}$ (notice that we are using bidimensional notation and that the network generates the filter from a region centered around $u(i, j)$). The set of all filters generated from image u is denoted as $\mathbf{d}_\psi^{\mathbf{u}}$. Then, the pixel at position (i, j) of the filtered image $\mathbf{r}$ obtained by the dynamic filtering layer on the input image $\mathbf{q}$ with the set of filters $\mathbf{d}_\psi^{\mathbf{u}}$ (notice that $\mathbf{u}$ and $\mathbf{q}$ do not have to coincide) is defined as

$$r(i,j) = \sum_{l=-L}^{L} \sum_{m=-M}^{M} d_\psi^{\mathbf{u}_{i,j}}(l+L, m+M) q(i+l, j+m) \tag{14}$$

Hence, the application of the dynamic filtering in (14) to $\mathrm{g}_\theta(\cdot)$ is expected to be a sharp, artifact-free restored image given by

$$\hat{\mathbf{x}} = \mathrm{r}_{\theta,\psi}(\mathbf{x_w}, \mathbf{y}) = \mathbf{D}_\psi^{\mathbf{x_w},\mathbf{y}} \mathrm{g}_\theta(\mathbf{x_w}, \mathbf{y}), \tag{15}$$

where $\mathbf{D}_\psi^{\mathbf{x_w},\mathbf{y}}$ is the (spatially variant) convolution matrix associated with the set of kernels $\mathbf{d}_\psi^{\mathbf{x_w},\mathbf{y}}$. Notice that a DFN is a perfect suit for this final refinement since it is able to produce spatially-variant filters, while normal CNN filters are spatially-invariant, requiring far more parameters to adapt to the spatially variant nature of these artifacts. The whole model used to deconvolve blurred images, which corresponds to $\mathrm{r}_{\theta,\psi}(\mathbf{x_w}, \mathbf{y})$, will be denoted by DDNet.

The architecture of DDNet is shown in Figure 2. The blue box represents the model $\mathrm{g}_\theta(\mathbf{x_w}, \mathbf{y})$ defined in (12). To reduce the number of computations, its main branch, $\mathrm{e}_\phi(\cdot)$, is shared with the filter generating network of the DNF, $\mathrm{d}_\psi(\cdot)$. This main branch follows an encoder-decoder structure. During the encoding phase we extract features from the image. At the same time, we reduce its spatial resolution by a factor of 4 using two space-to-depth operations defined as

$\mathrm{S}_2 : [0,1]^{2H\times 2W\times C} \rightarrow [0,1]^{H\times W\times 4C}$. In this projected space, the features are transformed using 10 Residual Dense Blocks (RDB) [55]. A RDB is a modified Residual Block (RB) with dense connections between layers. These dense connections allow to reuse features and provide a better performance. The features are finally up-scaled to the original image size using two sub-pixel convolutions [56] of factor 2. Notice that the spatial size reduction not only allows the network to process the image much faster but also to increase the receptive field without the need to increase the depth of the CNN model. The features obtained by the main branch are used to calculate the initial image estimation, $\mathrm{g}_\theta(\mathbf{x_w}, \mathbf{y})$, and the filters $\mathrm{d}_\psi(\mathbf{x_w}, \mathbf{y})$. The result of applying the filters, calculated via the dynamic filter module, produces the final estimation $\hat{\mathbf{x}}$ in (15) of our proposed $\mathrm{r}_{\theta,\psi}(\mathbf{x_w}, \mathbf{y})$ model.

All of the non-output convolutions outside an RDB block use $64\times 3\times 3$ filters and are followed by a Leaky ReLU with negative slope set to 0.2. Convolutions in each RDB only use $32 \times 3 \times 3$ filters, with the exception of the last convolution that uses $64\times 1\times 1$ filters, to control the increment in the number of parameters due to the use of dense connections. The size of the filters predicted by $\mathrm{d}_\psi(\cdot)$ is 5×5, $M = L = 2$.

III. Analytical method for blur estimation

The proposed deep learning approach for blind image deconvolution needs an estimate of the PSF. In the literature there are many methods to handle this estimation problem. In this paper we use a variation of the Bayesian BID method in [7]. This is a fast and robust method that provides an accurate estimation of the blur. It enforces sparsity on high pass filtered reconstructions, that is, on the edges of the image, using a Huber Super Gaussian (HSG) prior. For the blur, no prior knowledge other than non-negativity and that the blur coefficients should add to one is assumed.

The blur is estimated using a multiscale coarse-to-fine approach to avoid local minima. At each scale, an EM approach, starting from the upsampled blur and sharp image estimated from the previous scale, is used. Since the direct application of the EM inference to the joint probability function is infeasible due to the form of the HSG prior model, a Gaussian-like lower bound is obtained for the prior model which allows for the expectation of the joint distribution to be calculated analytically. This leads to an iterative algorithm where first the blur and the model parameters and then the real image are efficiently estimated [57].

In this paper, we modify the method in [7] by adding a kernel cleaning step at the end of each scale [12]. This step removes small valued isolated noisy pixels from the kernel and contributes to obtaining blur estimates comprised of connected pixels.

IV. Model Training

To train the proposed model we generated an image dataset based on the COCO 2017 Dataset[1]. It has been used for object segmentation and recognition and is comprised of a large number of sharp natural images divided in a train set of 118287 images, a validation set of 5000 images and a test set of 40670 images. To simulate the degraded images we proceeded as follows. First, we generated 1024 PSFs using the method in [58] of size between 11×11 and 65×65

[1] http://cocodataset.org/

pixels, with $T = 0.8$ and anxiety $10^r/1000$, where r is a random number from a uniform $[0,1]$ distribution. Then, we used 736 kernels for training, 32 for validation and 256 for testing, following approximately the same proportion of the images in each subset of COCO 2017. Each image in the training and validation sets was blurred with exactly 3 PSFs on their own set, that is, training images with training kernels and validation images with validation kernels, and Gaussian noise of standard deviation $\sigma = 0.01$ was added to the blurred image. Images with size smaller than 320 pixels in the shorter dimension were discarded to avoid boundary artifacts. Finally, the 256×256 central part of each image was cropped. This process generated a training set of 347436 images and a validation set of 14637 images. For testing, we created a reduced set of 512 images, obtained by degrading two randomly chosen test images for each kernel in the test set and adding Gaussian noise with $\sigma = 0.01$.

During training, data augmentation was performed by random vertically and horizontally flip and rotating each instance by multiples of 90º. We trained our models for 35 epochs using Adam optimizer [59] with weight decay set to 10^{-4}. Each epoch of the training consisted of 5428 batches of 64 images. The learning rate was set to 5×10^{-4} for the first 5 epochs, to 10^{-4} for the next 20 epochs, and to 10^{-5} for the last 10 epochs. During training, we set the regularization parameter for the Wiener filter, ϵ, equal to to 0.01.

Finally, to train the DDNet we use the Charbonnier loss defined as

$$\mathcal{L}(\hat{\mathbf{x}}, \mathbf{x}) = \sum_{i=1}^{N} \sqrt{(\hat{\mathbf{x}}(i) - \mathbf{x}(i))^2 + \varepsilon^2} \ , \tag{16}$$

where $\hat{\mathbf{x}}$ and $\mathbf{x}$ are, respectively, the estimated and the real image and ε is a constant, which in our experiments was set equal to 10^{-3}. We use this loss instead of the Mean Squared Error (MSE) since it is more robust to outliers [60]. Let us now describe how the blur is estimated.

V. Ablation Experiments

To determine the contribution of each component of the proposed DDNet to the final solution, we performed an experimental ablation study where we added each main component one at a time. We consider the following models:

1) $\mathrm{f}_\theta(\mathbf{y})$: The base model. We use our proposed CNN architecture using only the blurred image as input without the Wiener filtered image, no affine projection approximation [50] and no dynamic filtering.
2) $\mathrm{f}_\theta(\mathbf{x_w}, \mathbf{y})$: The base CNN architecture using both the Wiener filtered image and the blurred one as inputs for the network.
3) $\mathrm{f}_\theta(\mathbf{x_w}, \mathbf{y}) \times 2$: The base CNN architecture using both the Wiener filtered image and the blurred one as inputs, but using 20 RDBlocks instead of 10.
4) $\mathrm{g}_\theta(\mathbf{x_w}, \mathbf{y})$: We test our proposed adaptation of the affine projection approximation [50] (see (12)).
5) $\mathrm{r}_{\theta,\psi}(\mathbf{x_w}, \mathbf{y})$: Our complete DDNet.

The results of testing each model on the validation dataset are shown in Table I. From the results in the table, it is clear that each one of the proposed components significantly increases the performance of the model when added. The most remarkable performance increase is due to

TABLE I
Ablation study of components of the proposed DDNet. The comparison is done using our validation dataset and in terms of PSNR and SSIM. Best result is marked in bold. Second best result is underlined.

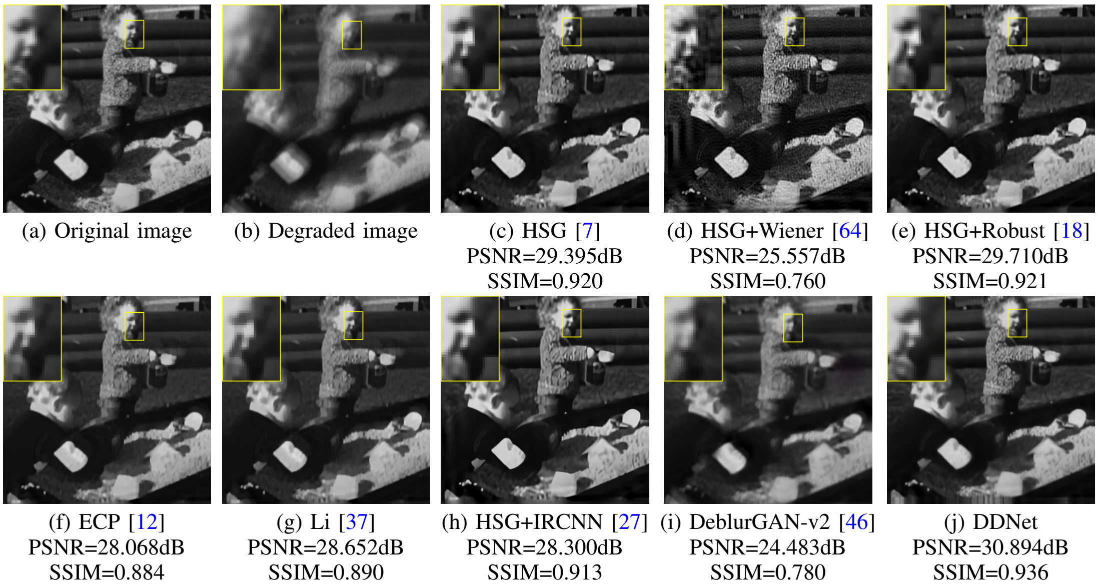
	$\mathrm{f}_\theta(\mathbf{y})$	$\mathrm{f}_\theta(\mathbf{x_w}, \mathbf{y})$	$\mathrm{f}_\theta(\mathbf{x_w}, \mathbf{y}) \times 2$	$\mathrm{g}_\theta(\mathbf{x_w}, \mathbf{y})$	$\mathrm{r}_{\theta,\psi}(\mathbf{x_w}, \mathbf{y})$
PSNR	20.12	26.15	$\underline{26.42}$	26.37	**26.59**
SSIM	0.6053	0.7267	$\underline{0.7355}$	0.7341	**0.7516**

the addition of the Wiener filtered image, $\mathbf{x_w}$, to the model inputs. However, this behavior was expected since $\mathrm{f}_\theta(\mathbf{y})$ solves a much harder problem than the rest. By using $\mathbf{x_w}$ the problem is reduced to a denoising one. Notice also that incorporating the affine projection approximation allows $\mathrm{g}_\theta(\mathbf{x_w}, \mathbf{y})$ to obtain a performance similar to $\mathrm{f}_\theta(\mathbf{x_w}, \mathbf{y}) \times 2$ with half the number of learnable parameters and being almost twice as fast as $\mathrm{f}_\theta(\mathbf{x_w}, \mathbf{y}) \times 2$. Finally, our complete DDNet, $\mathrm{r}_{\theta,\psi}(\mathbf{x_w}, \mathbf{y})$, clearly outperforms all others models while keeping a complexity similar to $\mathrm{g}_\theta(\mathbf{x_w}, \mathbf{y})$.

VI. Experimental Results

To assess the performance and robustness of the proposed DDNet method, we have tested it on several image datasets including synthetic datasets with spatially invariant and variant blurs as well as real image datasets. More concretely, we have used the dataset proposed by Levin *et al.* [61] composed of 4 grayscale images of size 255 × 255, blurred with 8 different kernels from 13 × 13 to 27 × 27 pixels, and with additive Gaussian noise with standard deviation 0.01. Also, we used a larger dataset, proposed by Sun *et al.* [62], with 80 grayscale images blurred with the 8 different kernels from [61] and with additive Gaussian noise with standard deviation 0.01. Image size is 1024 on its larger side. We have also used the datasets proposed by Lai *et al.* [63]. These include a set of 25 color images of sizes ranging from 350 × 500 to 1024 × 768 that fall in 5 different categories: man-made, natural, people, saturated and text. For the spatially invariant degraded dataset, each image was blurred with 4 different kernels (from 31 × 31 to 75 × 75 pixels) and Gaussian noise with standard deviation 0.01 was added to obtain a set of 100 degraded images. The same 25 crisp images were blurred with 4 different trajectories and Gaussian noise with standard deviation 0.01 was added to obtain 100 spatially variant degraded images. We also used the GoPro [44] dataset where a set of 1111 blurred images was obtained by averaging a number of consecutive frames, varying from 7 to 13, from 240fps videos captured with a GOPRO4 Hero Black camera. The sharp latent image is defined as the mid-frame among the sharp frames used to make the blurry image. A set of 2103 pairs of blurred/sharp images is also provided for training but it was not used by our method. While those datasets are synthetically generated, we also tested the proposed method on the real images dataset proposed in Lai *et al.* [63], a set of 100 real images from multiple sources and different categories.

The proposed DDNet method was compared with classic and state-of-the-art deconvolution methods. These include two analytical blind deconvolution methods, the Huber super Gaussian prior (HSG) method in [7] and the extreme channel prior (ECP) method in [12], the combined analytical and DL (Li) method in [37] and the GAN based (DeblurGAN-v2) method in [46]. As a baseline method we used the Wiener [64] method on the kernels estimated by the HSG method


(a) Original image

(b) Degraded image

(c) HSG [7]
PSNR=29.395dB
SSIM=0.920

(d) HSG+Wiener [64]
PSNR=25.557dB
SSIM=0.760

(e) HSG+Robust [18]
PSNR=29.710dB
SSIM=0.921

(f) ECP [12]
PSNR=28.068dB
SSIM=0.884

(g) Li [37]
PSNR=28.652dB
SSIM=0.890

(h) HSG+IRCNN [27]
PSNR=28.300dB
SSIM=0.913

(i) DeblurGAN-v2 [46]
PSNR=24.483dB
SSIM=0.780

(j) DDNet
PSNR=30.894dB
SSIM=0.936

Fig. 3. Visual comparison of the proposed DDNet and competing methods on an image of the Levin [61] dataset.

in [7], that we will refer to as HSG+Wiener. Using these same kernels, we also compared with the non-blind deconvolution deconvolution method handling inaccurate kernels (HSG+Robust) in [18] and the CNN based non-blind deconvolution (HSG+IRCNN) method in [27]. Since the Robust method in [18] is designed to handle only grayscale images, the method is applied to each one of the RGB channels independently. All methods were ran using the default parameters or the parameters suggested by the authors in the corresponding paper. For DDNet, we used $\epsilon = 0.01$ for all the datasets except for the Lai dataset were $\epsilon = 0.003$ was used since images degraded with large PSFs need a smaller regularization parameter. Note that, for large PSFs, the values of $\mathbf{H}$ are smaller compared to ϵ.

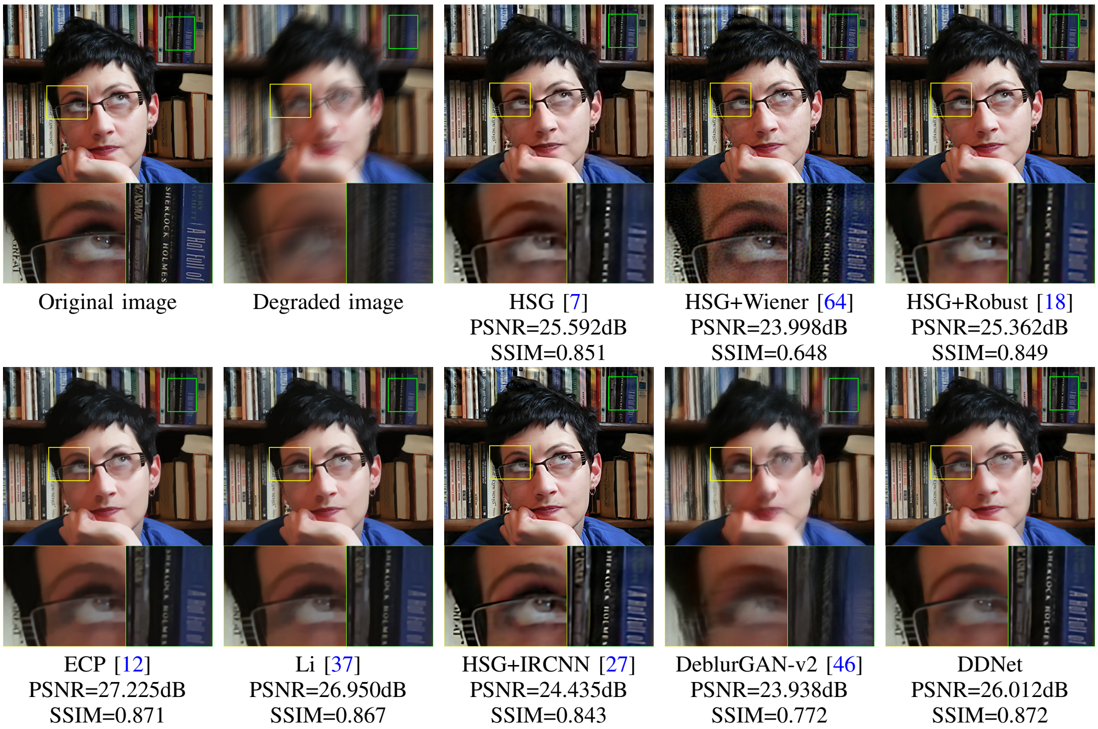
For quantitative comparison, PNSR and SSIM [65] quality measures were used. To avoid the inherent shifting ambiguity in blind deconvolution [66], we calculated both measures in the central part of the deconvolved images, trimming 25 pixels from each side for the Levin and Sun datasets and 37 pixels for the rest of the datasets. This trimmed image was shifted in a window of size 25×25 or 37×37 to find the position returning the best PSNR with respect to the original image. PSNR and SSIM results are reported for this position. The resulting figures of merit are summarized in table II. From this table it is clear that the proposed DDNet method outperforms all competing methods for spatially invariant degraded datasets and it is competitive in spatially variant ones. We want to emphasize that the proposed method provides the second best results in spatially variant degraded datasets in spite of not being trained on spatially variant degradations (see Fig. 7 for an example). Also, note that DeblurGAN-v2 was trained to deal with spatially variant degradations and, specifically, it was trained on a set of images that contained the GOPRO dataset so it is not strange that it outperforms all other methods on this dataset. It is worth mentioning that DDNet increases the PSNR up to 11.29 dB and the SSIM up to 0.242 with respect to the baseline method HSG+Wiener.

Figure 3 shows the original, observed and restored images with the competing and the proposed

TABLE II
NUMERICAL COMPARISON OF THE PROPOSED DDNET AND COMPETING METHODS ON DIFFERENT IMAGE DATASETS. BEST RESULT IS MARKED IN BOLD. SECOND BEST RESULT IS UNDERLINED.

Dataset	Method	PSNR	SSIM	time
COCO2017	HSG [7]	23.291	0.692	24.146
	HSG+Wiener [64]	22.220	0.557	23.228
	HSG+Robust [18]	22.923	0.676	323.128
	ECP [12]	22.141	0.678	124.903
	Li [37]	21.621	0.660	1095.140
	HSG+IRCNN [27]	22.936	0.711	23.456
	DeblurGAN-v2 [46]	21.847	0.572	0.258
	DDNet	**24.097**	**0.739**	23.138
Levin [61]	HSG [7]	29.148	0.900	11.229
	HSG+Wiener [64]	25.467	0.724	10.008
	HSG+Robust [18]	29.272	0.902	193.648
	ECP [12]	29.164	0.893	99.984
	Li [37]	27.930	0.857	149.063
	HSG+IRCNN [27]	27.968	0.889	10.268
	DeblurGAN-v2 [46]	24.034	0.737	0.187
	DDNet	**30.661**	**0.917**	10.026
Sun [62]	HSG [7]	29.649	0.843	130.127
	HSG+Wiener [64]	18.687	0.621	118.600
	HSG+Robust [18]	29.484	0.832	1940.991
	ECP [12]	27.730	0.820	1514.355
	Li [37]	27.393	0.814	1250.911
	HSG+IRCNN [27]	29.624	0.852	120.982
	DeblurGAN-v2 [46]	26.023	0.680	0.420
	DDNet	**29.975**	**0.852**	118.297
Lai [63] spatially invariant blur	HSG [7]	**19.958**	0.658	169.206
	HSG+Wiener [64]	18.844	0.523	157.716
	HSG+Robust [18]	19.795	0.642	4235.478
	ECP [12]	19.324	0.625	1010.981
	Li [37]	19.607	0.642	980.802
	HSG+IRCNN [27]	19.325	0.652	157.578
	DeblurGAN-v2 [46]	16.862	0.470	0.374
	DDNet	19.818	**0.670**	156.587
Lai [63] spatially variant blur	HSG [7]	17.063	0.527	184.929
	HSG+Wiener [64]	16.153	0.372	175.485
	HSG+Robust [18]	17.224	0.513	4256.082
	ECP [12]	16.801	0.530	1061.430
	Li [37]	16.432	0.515	1222.643
	HSG+IRCNN [27]	16.430	0.501	175.232
	DeblurGAN-v2 [46]	**18.747**	**0.571**	0.403
	DDNet	18.301	0.566	174.283
GOPRO [44]	HSG [7]	22.980	0.764	276.954
	HSG+Wiener [64]	21.246	0.585	259.659
	HSG+Robust [18]	23.004	0.758	8472.072
	ECP [12]	22.290	0.747	1057.433
	Li [37]	21.350	0.719	2175.334
	HSG+IRCNN [27]	21.921	0.720	263.933
	DeblurGAN-v2 [46]	**27.669**	**0.877**	0.448
	DDNet	25.494	0.827	258.490

method on an image of the Levin dataset. The inset shows a detail of the area marked in yellow. Although all compared methods produce good results on this image, most of them present artifacts that reduce their quality. Those artifacts include ringing artifacts (see Fig. 3(d)), blurriness (Fig. 3(i)), excessive smoothness (Fig. 3(c), (g)), phantoms (see the area of the elbow patch in Fig. 3(f), (g)) or washed out details (see, for instance, the jumper in Fig. 3(e), (f), (g) and (h)). The proposed method, however, is able to recover the small details removing all the ringing and



Original image

Degraded image

HSG [7]
PSNR=25.592dB
SSIM=0.851

HSG+Wiener [64]
PSNR=23.998dB
SSIM=0.648

HSG+Robust [18]
PSNR=25.362dB
SSIM=0.849

ECP [12]
PSNR=27.225dB
SSIM=0.871

Li [37]
PSNR=26.950dB
SSIM=0.867

HSG+IRCNN [27]
PSNR=24.435dB
SSIM=0.843

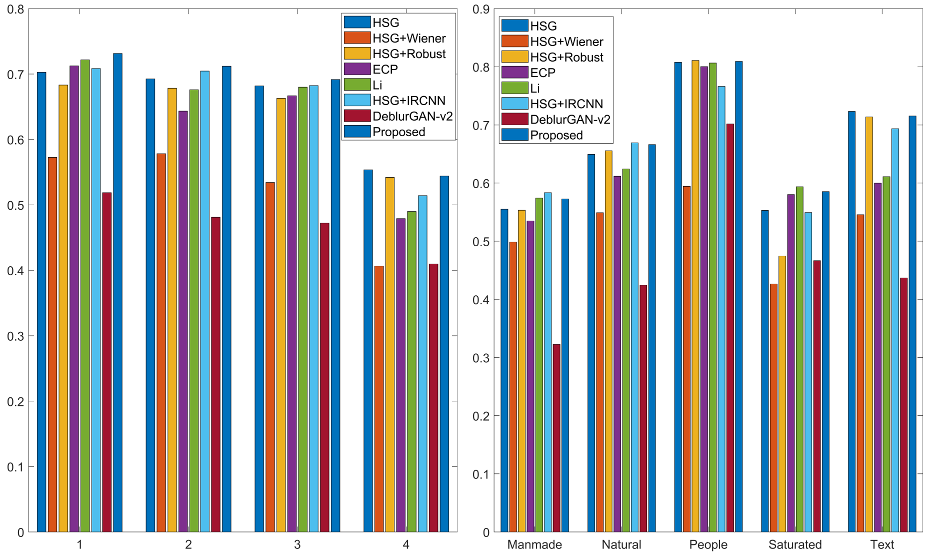
DeblurGAN-v2 [46]
PSNR=23.938dB
SSIM=0.772

DDNet
PSNR=26.012dB
SSIM=0.872

Fig. 4. Visual comparison of the proposed DDNet and competing methods on an image of the Lai [63] spatially invariant degraded dataset.

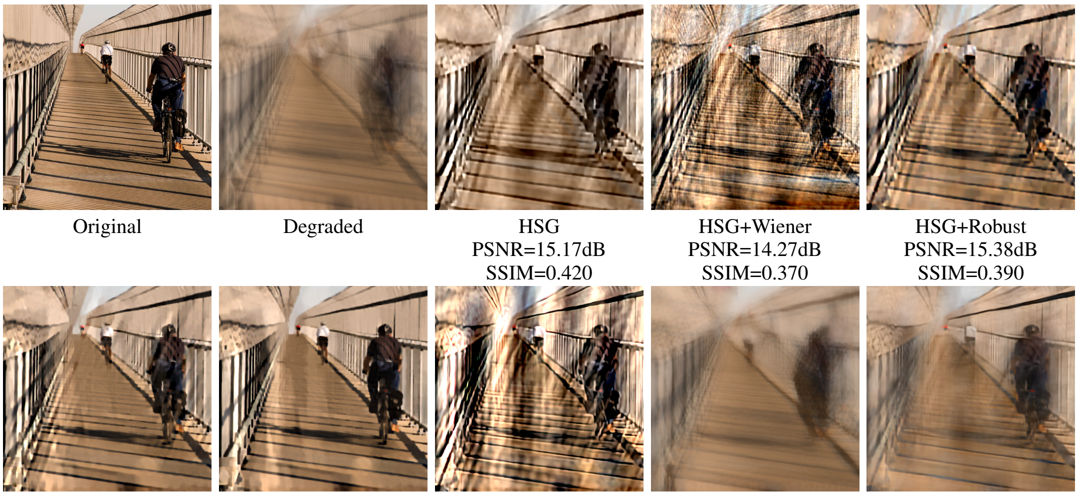
other artifacts.

We also present results on an image of the Lai [63] spatially invariant degraded dataset. The original image is shown in Fig. 4(a), the degraded image in Fig. 4(b), and Fig. 4(c)-(j) depict the deconvolved images with the competing and DDNet method. From these images it is clear that DDNet produces better visual results than the competing ones. The proposed method produces a sharp, artifacts free image. The method is able to recover small details (see, for instance, the books titles) without any of the artifacts of HSG+Wiener restoration. It is interesting to study the methods behavior in this dataset. First, we study how the PSF support affects the restoration quality. Figure 5(a) shows the SSIM of the proposed and competing methods for the different PSFs. All methods result in a lower SSIM for the images degraded with the PSF number 4, which has a much larger support than the other ones. However, the DDNet method, the HSG and the HSG+Robust methods are more robust to this problem. In general, the DDNet method outperformed the competing ones for kernel 1, performs slightly better for kernels 2 to 3 while performed slightly worse for kernel 4. If the PSF support is large, as is the case in the Lai spatially invariant dataset, the kernel is not always accurately estimated or solutions close to $\mathbf{H} = \mathbf{I}$ are produced. When trying to remove the artifacts, the proposed method further removes some details in the image. An example of this behavior can be observed in Fig. 6. Note that most of the competing methods also fail to recover this image or produce strong artifacts. Second, we study how robust is the proposed method to different types of images. Figure 5(b) shows the SSIM of the proposed DDNet and competing methods for the different categories of images. All methods, except deblurGAN-v2 and HSG+Wiener, perform similarly for man-made, natural and



(a) (b)

Fig. 5. Comparison of the proposed DDNet and competing methods on the Lai dataset in terms of mean SSIM. (a) For different PSFs. (b) For different image categories.



Original

Degraded

HSG
PSNR=15.17dB
SSIM=0.420

HSG+Wiener
PSNR=14.27dB
SSIM=0.370

HSG+Robust
PSNR=15.38dB
SSIM=0.390

ECP
PSNR=16.15dB
SSIM=0.479

Li
PSNR=17.82dB
SSIM=0.564

HSG+IRCNN
PSNR=13.41dB
SSIM=0.430

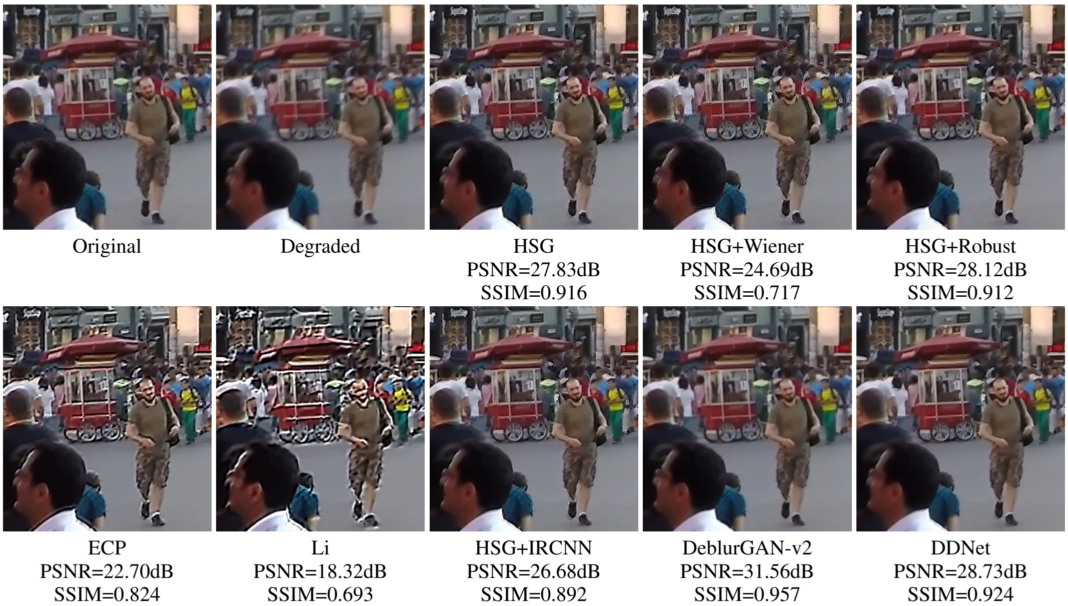
DeblurGAN-v2
PSNR=13.37dB
SSIM=0.181

DDNet
PSNR=14.92dB
SSIM=0.352

Fig. 6. Visual comparison of the proposed DDNet and competing methods on an image of the Lai [63] spatially variant degraded dataset.

people images. The SSIM values of HSG, HSG+Robust and DDNet methods are consistently high for those images, being the mean SSIM of the proposed method higher the one of HSG and HSG+Robust for these categories. For saturated images, all methods produce lower SSIM results. The proposed method produces better results than the competing methods based on HSG kernel estimation and DeblurGAN-v2 method, despite the low quality of the Wiener reconstruction. Finally, we want to note that the DDNet method produces also very good results for text images, although it was not trained on this kind of images.

Figure 8 depicts three details of challenging real images as well as the deconvolutions with the proposed and competing methods. The DDNet method provides artifact free images and, compared



Original

Degraded

HSG
PSNR=27.83dB
SSIM=0.916

HSG+Wiener
PSNR=24.69dB
SSIM=0.717

HSG+Robust
PSNR=28.12dB
SSIM=0.912

ECP
PSNR=22.70dB
SSIM=0.824

Li
PSNR=18.32dB
SSIM=0.693

HSG+IRCNN
PSNR=26.68dB
SSIM=0.892

DeblurGAN-v2
PSNR=31.56dB
SSIM=0.957

DDNet
PSNR=28.73dB
SSIM=0.924

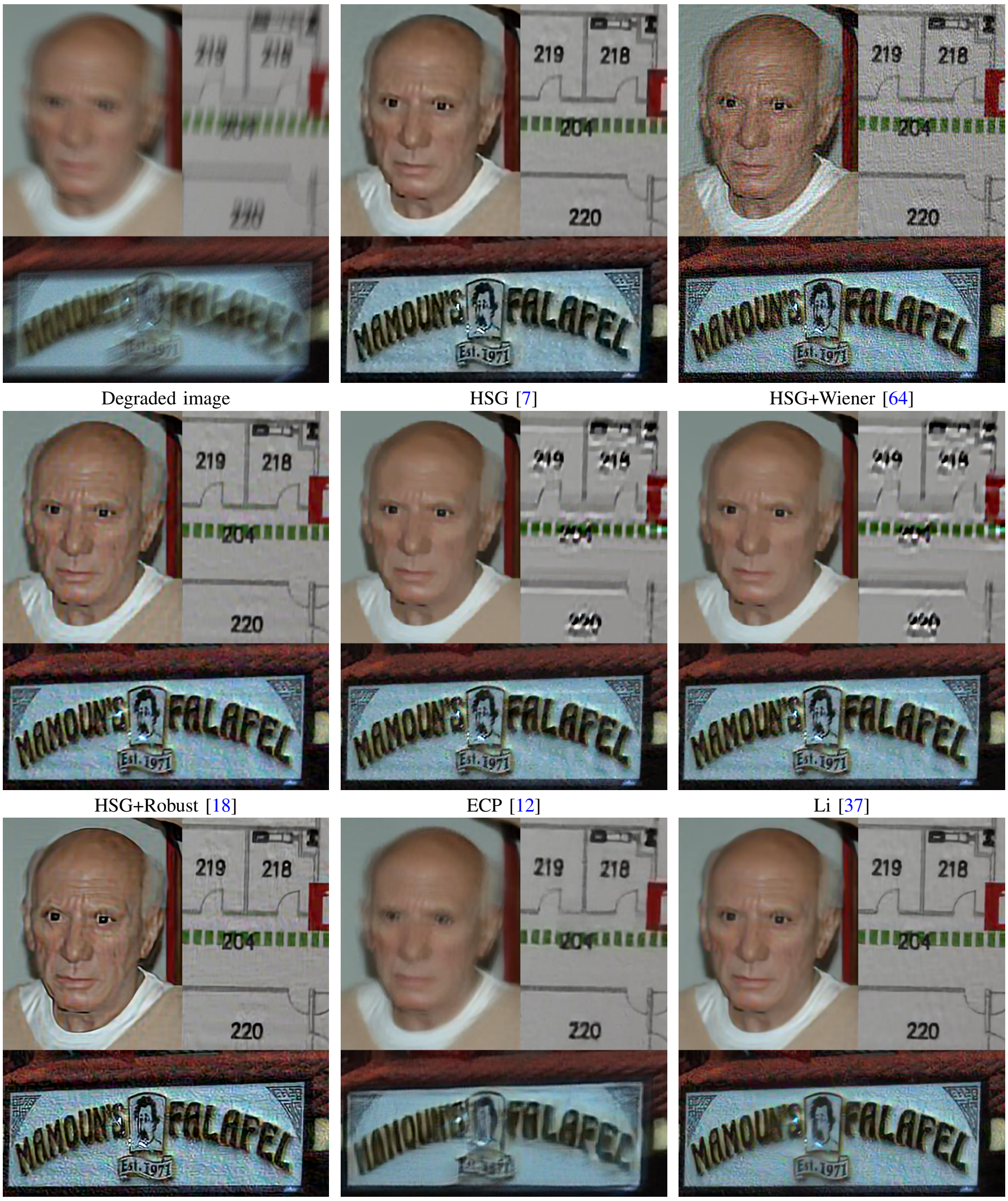
Fig. 7. Visual comparison of the proposed DDNet and competing methods on the GOPRO [44] dataset.

with other methods, more consistent results. When compared to HSG+Wiener, it is clear that the proposed method is able to remove all restoration artifacts while maintaining the details in the image. The DDNet method produces sharper images than HSG, ECP, Li, and DeblurGan-v2 methods, fewer artifacts than HSG+Robust (see, the color artifacts in Picasso's face and hair, for instance) and, although not as crisp as HSG+IRCNN, it produces less noisy images (see the roof over the restaurant sign), without artifacts in flat regions and more natural looking image (see, Picasso's cheek and T-shirt). In general, the proposed DDNet method is robust to different categories of images and blurs, and provides good, natural looking, artifacts free images.

With respect to the computational cost, the DDNet method achieves the second lowest time, behind DeblurGAN-v2 [46], but usually with a much higher restoration quality. Note that most of the time is spent in the estimation of the blur kernel, which is carried out in the CPU, while the CNN cost is of a few hundredths of a second. Compared with other combined analytical and DL methods, the DDNet method is much faster than the Li [37] and HSG+IRCNN [27] methods. Note that the Li [37] method does not use GPU computing. The proposed method only needs a fraction of the computing time of the IRCNN time with better quality results.

VII. Conclusions

We have proposed a combination of an analytical and a DL method for blind image deconvolution. The proposed DL method adapts the affine projection [50] to estimate a solution to the BID problem consistent with the degradation model. The adaptation includes the use of a (possible inaccurate) estimation of the blur and the use of the Wiener filter as an approximation of the pseudo-inverse of the blur convolution operator. The PSF estimation is carried out by an analytical approach while image deconvolution is handled by a DL model that removes noise and ringing artifacts. Spatially variant artifacts caused by an inexact PSF estimation or other degradation model inconsistencies are corrected using the spatially-variant filters produced by a DNF. As



Degraded image HSG [7] HSG+Wiener [64]

HSG+Robust [18] ECP [12] Li [37]

HSG+IRCNN [27] DeblurGAN-v2 [46] DDNet

Fig. 8. Visual comparison of the proposed DDNet and competing methods on real images.

shown by our ablation experiments, this approach is able to outperform similar CNN models twice as deep. The proposed DDNet method outperforms the rest of the compared methods in spatially invariant degraded images and is competing in spatially variant degraded ones, producing better results than all the methods except DeblurGAN-v2. The proposed method has proved to be robust, always providing one of the best results (on average) in different categories of images, including challenging faces, saturated and text images. The DDNet method is faster than all other analytical methods and the combined analytical and DL method. Most of the time is consumed by the PSF estimation, that is carried out in the CPU. In the future, we will extend the proposed approach to perform PSF estimation also using a GPU.

References

[1] P. Ruiz, X. Zhou, J. Mateos, R. Molina, and A. K. Katsaggelos, "Variational Bayesian blind image deconvolution: A review," *Digital Signal Processing*, vol. 47, pp. 116–127, 2015.

[2] X. Zhou, J. Mateos, F. Zhou, R. Molina, and A. K. Katsaggelos, "Variational Dirichlet blur kernel estimation," *IEEE Trans. Image Process.*, vol. 24, pp. 5127–5139, 2015.

[3] D. Krishnan and R. Fergus, "Fast image deconvolution using hyper-laplacian priors," in *Conference on Neural Information Processing Systems*, 2009.

[4] B. Zhang, R. Liu, H. Li, Q. Yuan, X. Fan, and Z. Luo, "Blind Image Deblurring Using Adaptive Priors," in *Internet Multimedia Computing and Service*, ser. Communications in Computer and Information Science. Springer, Singapore, Aug. 2017, pp. 13–22.

[5] D. Perrone, R. Diethelm, and P. Favaro, "Blind deconvolution via lower-bounded logarithmic image priors," in *International Conference on Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR)*, 2015.

[6] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman, "Removing camera shake from a single photograph," *ACM Trans. Graph*, vol. 25, no. 3, 2006.

[7] X. Zhou, M. Vega, F. Zhou, R. Molina, and A. K. Katsaggelos, "Fast Bayesian blind deconvolution with Huber super Gaussian priors," *Digital Signal Processing*, vol. 60, pp. 122–133, 2017.

[8] S. D. Babacan, R. Molina, M. N. Do, and A. K. Katsaggelos, "Bayesian blind deconvolution with general sparse image priors," in *Computer Vision – European Conference on Computer Vision 2012*, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 341–355.

[9] L. Chen, Q. Sun, and F. Wang, "Adaptive blind deconvolution using generalized cross-validation with generalized lp/lq norm regularization," *Neurocomputing*, vol. 399, pp. 75 – 85, 2020.

[10] L. Xu, S. Zheng, and J. Jia, "Unnatural L0 sparse representation for natural image deblurring," in *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition*, ser. CVPR '13. Washington, DC, USA: IEEE Computer Society, 2013, pp. 1107–1114.

[11] J. Pan, D. Sun, H. Pfister, and M. Yang, "Blind image deblurring using dark channel prior," in *2016 IEEE Conf. Comput. Vision and Pattern Recognition*, 2016, pp. 1628–1636.

[12] Y. Yan, W. Ren, Y. Guo, R. Wang, and X. Cao, "Image deblurring via extreme channels prior," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6978–6986.

[13] J. Kotera, V. Šmídl, and F. Šroubek, "Blind deconvolution with model discrepancies," *IEEE Transactions on Image Processing*, vol. 26, no. 5, pp. 2533–2544, May 2017.

[14] X. Chen, Y. Zhu, J. Sun, and Y. Zhang, "Robust Motion Blur Kernel Estimation by Kernel Continuity Prior," *IEEE Access*, vol. 8, pp. 46 162–46 175, 2020.

[15] C. Yang, M. Chang, H. Feng, Z. Xu, and Q. Li, "Beyond camera motion removing: How to handle outliers in deblurring," *arXiv:2002.10201 [cs, eess]*, Feb. 2020, arXiv: 2002.10201.

[16] Q. Shan, J. Jia, and A. Agarwala, "High-quality motion deblurring from a single image," *ACM Trans. Graph*, vol. 27, no. 3, 2008.

[17] N. Galatsanos, V. Mesarovic, R. Molina, A. Katsaggelos, and J. Mateos, "Hyperparameter estimation in image restoration problems with partially-known blurs," *Optical Engineering*, vol. 41, no. 8, pp. 1845–1854, 2002.

[18] H. Ji and K. Wang, "Robust image deblurring with an inaccurate blur kernel," *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 1624–1634, 2012.

[19] A. Lucas, M. Iliadis, R. Molina, and A. K. Katsaggelos, "Using deep neural networks for inverse problems in imaging," *Signal Processing Magazine*, vol. 35, no. 1, pp. 20–36, 2018.

[20] V. Jain and S. Sebastian, "Natural image denoising with convolutional networks," in *Advances in Neural Information Processing Systems 21*, D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, Eds. Curran Associates, Inc., 2009, pp. 769–776.

[21] J. Xie, L. Xu, and E. Chen, "Image Denoising and Inpainting with Deep Neural Networks," *Conference on Neural Information Processing Systems*, pp. 1–9, 2012.

[22] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 2, pp. 295–307, Feb. 2016.

[23] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, pp. 1–122, 2011.

[24] D. Geman and C. Yang, "Nonlinear image recovery with half-quadratic regularization," pp. 932–946, 1995.

[25] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo, "Fast image recovery using variable splitting and constrained optimization," *IEEE transactions on image processing*, vol. 19(9), no. 3, pp. 2345–2356, 2010.

[26] T. Meinhardt, M. Möller, C. Hazirbas, and D. Cremers, "Learning proximal operators: Using denoising networks for regularizing inverse imaging problems," in *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, 2017, pp. 1799–1808.

[27] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep cnn denoiser prior for image restoration," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3929–3938.

[28] N. Parikh and S. Boyd, "Proximal algorithms," *Found. Trends Optim.*, vol. 1, no. 3, pp. 127–239, Jan. 2014.

[29] L. Huang and Y. Xia, "Joint blur kernel estimation and cnn for blind image restoration," *Neurocomputing*, vol. 396, pp. 324 – 345, 2020.

[30] J. H. R. Chang, C. Li, B. Póczos, and B. V. K. V. Kumar, "One network to solve them all — solving linear inverse problems using deep projection models," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct. 2017, pp. 5889–5898.

[31] W. Dong, P. Wang, W. Yin, G. Shi, F. Wu, and X. Lu, "Denoising prior driven deep neural network for image restoration," *Computing Research Repository*, vol. abs/1801.06756, 2018.

[32] D. Gong, Z. Zhang, Q. Shi, A. van den Hengel, C. Shen, and Y. Zhang, "Learning deep gradient descent optimization for image deconvolution," *IEEE Transactions on Neural Networks and Learning Systems*, vol. in press, 2020.

[33] J. Sun, W. Cao, Z. Xu, and J. Ponce, "Learning a convolutional neural network for non-uniform motion blur removal," in *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on.* IEEE, 2015, pp. 769–777.

[34] R. Yan and L. Shao, "Blind image blur estimation via deep learning," *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1910–1921, Apr. 2016.

[35] C. J. Schuler, M. Hirsch, S. Harmeling, and B. Schölkopf, "Learning to deblur," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 7, pp. 1439–1451, 2016.

[36] A. Chakrabarti, "A neural approach to blind motion deblurring," in *Computer Vision - ECCV 2016 - 14th European Conference*, 2016, pp. 221–235.

[37] L. Li, J. Pan, W. Lai, C. Gao, N. Sang, and M. Yang, "Learning a discriminative prior for blind image deblurring," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6616–6625.

[38] Z. Luo, S. Chen, and Y. Qian, "A deep optimization approach for image deconvolution," 2019, arXiv:1904.07516.

[39] S.-W. Lee, J.-H. Kim, J. Jun, J.-W. Ha, and B.-T. Zhang, "Overcoming catastrophic forgetting by incremental moment matching," in *Advances in Neural Information Processing Systems 30*, 2017, pp. 4652–4662.

[40] Z. Wang, Z. Wang, Q. Li, and H. Bilen, "Image deconvolution with deep image and kernel priors," in *The IEEE International Conference on Computer Vision (ICCV) Workshops*, Oct 2019.

[41] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep image prior," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 9446–9454.

[42] Y. Li, M. Tofighi, J. Geng, V. Monga, and Y. C. Eldar, "Efficient and interpretable deep blind image deblurring via algorithm unrolling," *IEEE Transactions on Computational Imaging*, vol. 6, pp. 666–681, 2020.

[43] S. Vasu, V. R. Maligireddy, and A. N. Rajagopalan, "Non-blind deblurring: Handling kernel uncertainty with cnns," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3272–3281.

[44] S. Nah, T. H. Kim, and K. M. Lee, "Deep multi-scale convolutional neural network for dynamic scene deblurring," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 257–265.

[45] O. Kupyn, V. Budzan, M. Mykhailych, D. Mishkin, and J. Matas, "Deblurgan: Blind motion deblurring using conditional adversarial networks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8183–8192.

[46] O. Kupyn, T. Martyniuk, J. Wu, and Z. Wang, "Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 8877–8886.

[47] T. H. Kim, K. M. Lee, B. Schölkopf, and M. Hirsch, "Online video deblurring via dynamic temporal blending network," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct. 2017, pp. 4058–4067.

[48] P. Wieschollek, M. Hirsch, B. Schölkopf, and H. Lensch, "Learning blind motion deblurring," in *Proceedings IEEE International Conference on Computer Vision (ICCV)*. Piscataway, NJ, USA: IEEE, Oct. 2017, pp. 231–240.

[49] X. Jia, B. De Brabandere, T. Tuytelaars, and L. V. Gool, "Dynamic filter networks," in *Advances in Neural Information Processing Systems 29*, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, Eds. Curran Associates, Inc., 2016, pp. 667–675.

[50] C. Sonderby, J. Caballero, L. Theis, W. Shi, and F. Huszar, "Amortised MAP inference for image super-resolution," in *International Conference on Learning Representations*, 2017. [Online]. Available: https://arxiv.org/abs/1610.04490

[51] A. Albert, *Regression and the Moore-Penrose pseudoinverse*, ser. Mathematics in Science and Engineering. Burlington, MA: Elsevier, 1972. [Online]. Available: https://cds.cern.ch/record/1253778

[52] S. Lopez-Tapia, A. Lucas, R. Molina, and A. K. Katsaggelos, "Multiple-degradation video super-resolution with direct inversion of the low-resolution formation model," in *2019 27th European Signal Processing Conference (EUSIPCO)*, Sep. 2019, pp. 1–5.

[53] S. López-Tapia, A. Lucas, R. Molina, and A. Katsaggelos, "A single video super-resolution gan for multiple downsampling operators based on pseudo-inverse image formation models," *Digital Signal Processing*, 2020.

[54] "Chapter III Geometric and Analytic Properties of the Moore-Penrose Pseudoinverse," in *Regression and the Moore-Penrose Pseudoinverse*, ser. Mathematics in Science and Engineering, A. Albert, Ed. Elsevier, 1972, vol. 94, pp. 15 – 42, iSSN: 0076-5392. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0076539208629197

[55] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. C. Loy, "Esrgan: Enhanced super-resolution generative adversarial networks," in *The European Conference on Computer Vision Workshops (ECCVW)*, September 2018.

[56] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 1874–1883.

[57] X. Zhou, R. Molina, F. Zhou, and A. K. Katsaggelos, "Fast iteratively reweighted least squares for l_p regularized image deconvolution and reconstruction," in *2014 IEEE Int. Conf. Image Process. (ICIP)*, 2014, pp. 1783–1787.

[58] G. Boracchi and A. Foi, "Modeling the performance of image restoration from motion blur," *IEEE Transactions on Image Processing*, vol. 21, no. 8, pp. 3502–3517, 2012.

[59] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980

[60] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Deep laplacian pyramid networks for fast and accurate super-resolution," in *IEEE Conferene on Computer Vision and Pattern Recognition*, 2017.

[61] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman, "Understanding and evaluating blind deconvolution algorithms," in *Computer Vision and Pattern Recognition*, 2009.

[62] L. Sun, S. Cho, J. Wang, and J. Hays, "Edge-based blur kernel estimation using patch priors," in *2013 IEEE Int. Conf. on Comput. Photography (ICCP)*, 2013.

[63] W.-S. Lai, J.-B. Huang, Z. Hu, N. Ahuja, and M.-H. Yang, "A comparative study for single image blind deblurring," in *2016 IEEE Conf. Comput. Vision and Pattern Recognition (CVPR)*, 2016, pp. 1701–1709.

[64] R. Gonzalez, R. Woods, and S. Eddins, *Digital Image Processing Using Matlab*. Prentice Hall, 2003.

[65] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[66] D. Perrone and P. Favaro, "A logarithmic image prior for blind deconvolution," *International Journal of Computer Vision*, vol. 117, no. 2, pp. 159–172, 2016.

3.2 Fast and Robust Cascade Model for Multiple Degradation Single Image Super-Resolution

3.2.1 Publication details

Authors: Santiago López-Tapia, and Nicolás Pérez de la Blanca.
Title: A Fast and Robust Multi-Degradation Cascade Model for Single Image Super-Resolution.
Publication: IEEE Transactions on Image Processing, 2020.
Status: Under review.
Quality indices:

- Impact Factor (JCR 2019): 9.340
- Rank: 11/266 (Q1 and D1) in Engineering, Electrical and Electronic

3.2.2 Main Contributions

- In this work we introduce a new fast, accurate, and robust CNN cascade-based approach for Single Image Super-Resolution (SISR). This model is able to deconvolve and super-resolve the image at the same time.
- Our approximation uses a novel connection approach where each sub-module reuses the features calculated by all previous sub-modules.
- We propose introducing a new module that extracts and matches multi-scale features from the blurred LR image and a non-blurred but noisy Weiner filtered LR image. The rest of the network will use these features.
- Making use of "Privileged Information" [134], we train the deblur sub-module to approximate not the restored LR but one with a little of Gaussian blur that is easier to super-resolve.
- We split the upsampling module into two: upsampling and one final refinement module that corrects the cascade's accumulated errors. This is done using the approximation proposed in [44] to constrain the output of the upsampling to fulfill the LR image formation model.
- We perform our experiments using standard SISR image datasets with blur kernels simulating simple and complex camera movement. These experiments are performed in both non-blind and blind settings. We compare our model to the current state-of-art in terms of PSNR, SSIM and the number of operations.

Fast and Robust Cascade Model for Multiple Degradation Single Image Super-Resolution

Santiago López-Tapia, Nicolás Pérez de la Blanca
Dept. of Computer Science and Artificial Intelligence, *University of Granada*, Granada, Spain
Email: {sltapia, nicolas}@decsai.ugr.es

Abstract

Single Image Super-Resolution (SISR) is one of the low-level computer vision problems that has received increased attention in the last few years. Current approaches are primarily based on harnessing the power of deep learning models and optimization techniques to reverse the degradation model. Owing to its hardness, isotropic blurring or Gaussians with small anisotropic deformations have been mainly considered. Here, we widen this scenario by including large non-Gaussian blurs that arise in real camera movements. Our approach leverages the degradation model and proposes a new formulation of the Convolutional Neural Network (CNN) cascade model, where each network sub-module is constrained to solve a specific degradation: deblurring or upsampling. A new densely connected CNN-architecture is proposed where the output of each sub-module is restricted using some external knowledge to focus it on its specific task. As far we know this use of domain-knowledge to module-level is a novelty in SISR. To fit the finest model, a final sub-module takes care of the residual errors propagated by the previous sub-modules. We check our model with three state of the art (SOTA) datasets in SISR and compare the results with the SOTA models. The results show that our model is the only one able to manage our wider set of deformations. Furthermore, our model overcomes all current SOTA methods for a standard set of deformations. In terms of computational load, our model also improves on the two closest competitors in terms of efficiency. Although the approach is non-blind and requires an estimation of the blur kernel, it shows robustness to blur kernel estimation errors, making it a good alternative to blind models.

Index Terms

Single image super-resolution, super-resolution, multiple degradation deconvolution, convolutional neural networks, cascade model.

I. Introduction

Single Image Super-Resolution (SISR) is a fundamental low-level computer vision that has received considerable attention in recent years [1], [2], [3], [4], [5], [6], [7], [8]. It consists of recovering a high-resolution (HR) image from a given low resolution (LR) image, assuming

Santiago López-Tapia received financial support through the Spanish FPU program.

a degradation model connecting both images. Typically, the general image degradation model assumed in the SISR literature is given by,

$$y = [x \circledast k] \downarrow_s + n, \tag{1}$$

where x is the HR image, y is the LR image, n is the noise, usually additive white Gaussian noise (AWGN), $\downarrow_s$ is the downsampling operator for factor s and $x \circledast k$ represents the convolution of x with the blur kernel k.

The current models used to estimate x from y are typically grouped into three categories according to the assumed degradation model and the solver used: interpolation-based (IB), model-based or energy-function optimization (MBO) and learning-based (LB). The IB approach assumes the following simpler model as the degradation model,

$$y = x \downarrow_s + n, \tag{2}$$

and an interpolation technique is proposed as the solver. Clearly, Eq. (2) models the downsampling operation including as part of the noise any other degradation present in the image. Although these methods are the fastest, they are too simple to cope with real blur degradation effects [9], [8], [10].

Traditional MBO approaches [11], [12], [13], [14], [15] define a regularized energy function that is optimized to estimate x. The degradation model (1), together with smoothness or edge preservation conditions in solution, are the most common elements that define the energy function [15], [13]. In these models, an optimization is performed for each new LR image, which provides them with high flexibility, under the LR imaging generation conditions, to manage any degradation type. However, closed-form solutions are not feasible and the use of iterative optimization techniques makes MBO approaches computationally expensive. In addition, the hyperparameters must be hand-picked for each degradation type [16].

In recent years, LB approaches have gained significant momentum thanks to the introduction of deep learning (DL). Recently, variable splitting techniques such as half-quadratic splitting (HQS) [17] and alternating direction method of multipliers (ADMM) [18] have been used to incorporate DL in MBO to improve efficiency. In [6], [19], and [20], for instance, a denoising CNN was used to estimate the final HR image. Furthermore, in [8], the authors expand on this approximation by tasking the CNN with upsampling and denoising in a non-blind approach. Clearly, this introduces a connection between MBO and CNN-based approaches because the MBO solution eventually relies on learned modules. However, this connection assumes a new degradation model,

$$y = x \downarrow_s \circledast k^L + n, \tag{3}$$

where blurring k^L is applied on a downsampled image. Until now, no connection has been provided between this new model and the one in Eq. (1).

In contrast to MBO approaches, most LB models do not explicitly use any image degradation model to estimate x, assuming that the degradation model is well represented by Eq. (2). A large database of pairs (LR, HR) is used to learn an end-to-end mapping $f(\cdot)$ such that $x = f(y)$ [21], [22], [23], [24]. The DL models based on Convolutional Neural Networks (CNNs) have shown the highest performance compared to any other approach [2], [4], [5], [7], [25]. The early success of vanilla CNN models in SISR [2], [26] has stimulated the development of deeper designs that have achieved the current state of the art (SOTA) in SISR. As a matter of fact, most of the proposals from the first one, given in [2], have been driven by the incorporation of technical achievements in the CNN-field as for instance, deeper models, and use of residual block to improve accuracy metrics [27], [28], [29], [5], [7], [30], [31], [32]. To improve the performance and processing time, upsampling moves to the last layer of the network [26], [33], [29]. In addition, SRGAN [29] and EDSR [5] improved over those architectures and introduced residual blocks to increase the network depth. DenseSR [30] used residual dense blocks to improve the networks by combining features from different layers. In ESRGAN [32] and RCAN [31] specific blocks to improve performance are proposed, reaching a new SOTA, but at the cost of a significant increase in computing time. Although these models outperform traditional MBO in terms of performance, they lack adaptation to new deformations at test time.

Very recently, CNN-based models (CNN-BMs) assuming Eq. (1) as a degradation model have been proposed. In this scenario, Shocher et al. [34] proposed a technique for the “Zero-Shot” case that exploits the internal recurrence of information inside a single image to train a DL model. This shares the flexibility at test time, but also the high time consumption of MBO approaches. Alternatively, recent works have proposed architectures that encode information about the blur kernel k and introduce it into the network. Riegler et al. [35] used conditioned regression models to encode a Gaussian family of kernels. In SRMD [9], a Principal Component Analysis (PCA) representation of the blur kernel and a stretching strategy to concatenate it with the LR image is proposed. In [10], the latter approach was expanded and improved, with the use of spatial feature transform (SFT) layers [7], to introduce the PCA representation of the blur kernel into the network. By doing so, they are able to implement a very deep residual network with a significant increase in performance over [9]. However, PCA is not a suitable representation to encode families of blurs with different degrees of freedom, in addition to having to be recalculated to cope with new degradation types. Consequently, the CNN models used in [9] and [10] show shortcomings when blur kernels from strong camera movement are present [9].

An alternative to the above CNN-BMs that attempt to solve the SISR problem using a single CNN is to decompose the problem into sub-tasks and use a group of CNNs connected in a cascade fashion to solve it. This “divide and conquer” approach simplifies the function that each sub-module must learn, easing the learning process. If $n = 0$, Eq. (1) can be inverted using an upsampling CNN followed by a deblurring CNN. However, in practice, this approximation does not perform well because of the introduction of strong artifacts during upsampling and the increasing difficulty of deconvolving in HR space [10]. The CNN-BMs based on cascades [10], [8], [36] make the first denoise/deblur and later upsampling implicitly assuming that Eq. (1) can be approximated by Eq. (3). In [36], an unsupervised network was trained to deblur and denoise the LR image, before using an SR network to estimate the HR. However, as shown in [10] and [8], these cascade approaches underperform compared with other SOTA models. Here, we argue that cascade models can be significantly improved by modifying the connection between the sub-modules and taking advantage of the fact that each sub-module can be specialized in a task by constraining its output using domain knowledge.

In this study, we propose a fast, accurate, and robust CNN cascade-based approach for multiple degradations SISR, CAscade-Deblurring-Upsampling-Fusion (CADUF). Compared with previous cascade-based approaches, our model shows new elements regarding deblur, upsampling, and information processing. In deblurring, the LR image is complemented with the information provided by a deblurred and noisy release of itself obtained from the Weiner filter. We take advantage of the fact that artifacts and noise introduced by the Weiner filter are similar across all possible combinations of blur and images. After extracting and matching features from the two images, we use those features to filter out the noise and artifacts in the Weiner-filtered image. In upsampling, we make use of domain knowledge in the form of "Privileged Information" [37] and constrain the output of upsampling to fulfill an affine projection model. As a result, we obtain an error range that can be reduced by a simple final sub-module. Finally, we improve the communication between sub-modules by using a novel connection approach where each sub-module re-uses the features calculated by all previous sub-modules. Our experiments for non-blind SISR show that CADUF reaches SOTA results being competitive with MBO for unseen blurs but much faster. Furthermore, experiments performed in the blind setting show that our method is also robust to inaccuracies in the blur kernel estimation.

The remainder of the paper is organized as follows. In Section II, we describe the proposed CADUF model and explain each one of its primary components. Section III presents the experimental settings used in this study (dataset, degradations and training hyperparameters) as well as an ablation study of each key component and a comparison with SOTA methods. Finally, Sect. IV concludes the paper.

II. Method

In this section, we first introduce the SR problem settings and propose a new CNN architecture that addresses it by combining multiple CNN sub-modules that solve specific sub-problems. Each of these CNN sub-modules are presented in Sections II-A to II-D. Finally, in Section II-E, we describe in detail the parameters of the proposed architecture.

As previously stated in Section I, we consider that the LR image y is produced from the high-res image x in Eq. (1). Typically, in SISR, k is assumed to be an isotropic Gaussian kernel and $\downarrow_s$ bicubic interpolation [9], [10], but in real applications, different kernel types could arise. In this study, we consider that k^H can also be a more complex blurs, like motion blur. Fig. 1 shows some examples of the complexity of the kernels used during our experiments.

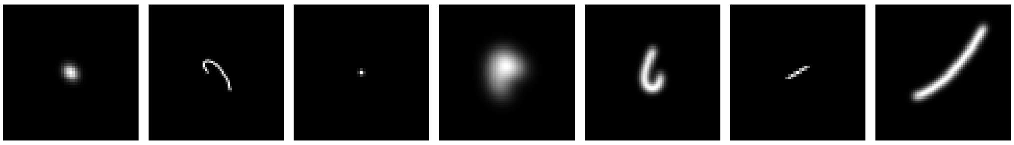


Fig. 1. Examples of the blur kernels k considered in this work. They represent a blur combination of isotropic Gaussian and motion.

We propose a CNN architecture composed of several CNN sub-modules, each of which focuses on solving a portion of the degradation process that produces the LR image. Using this approach has several advantages over using a current SR CNN:

1) By breaking the SR task into sub-tasks each addressed by a sub-module, a strong regularization is imposed over the function space that the model is able to explore. This makes learning easier, leading to improved optima and generalization.
2) Task-specific constraints in each sub-module can be introduced, to further reduce the search space (see Sections II-B and II-C).
3) Specialized architectural elements can be used for each task in each sub-module, improving model performance and efficiency. (see Section II-A).

To do this, we design a cascade based on Eq (3) as an approximation to Eq (1). This allows us to first perform denoising and deblurring before we upsample the LR image. Based on Eq. (3) and assuming that k^L is known, we propose the decomposition of the architecture in the following specialized sub-modules:

1) E_θ: Extraction of motion-corrected features. See Section II-A.
2) D_ϕ: Deblur and Denoising of the LR image, Section II-B.
3) U_ψ: Upsampling, to get an initial SR solution, Section II-C.
4) F_ω: Refinement, improvement of the initial SR solution by cleaning the errors and artifacts caused by the accumulated errors in the cascade. See Section II-D.

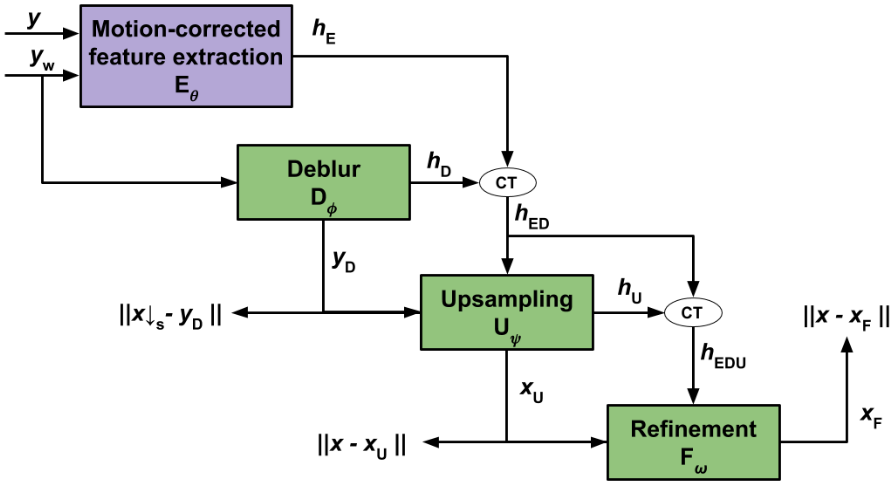


Fig. 2. Proposed architecture for solving the MDSR problem. Each sub-module is a CNN and is specialized in solving a sub-task. Note that “CT” indicates concatenation of the feature maps.

The structure of the proposed architecture is shown in Fig.2. The architecture follows a cascade approach where the features h and the result of each sub-task are needed for the next sub-module in the cascade. Compared to other cascade approaches, our sub-modules use not only the previously calculated image, but also the features calculated by all previous sub-modules, which allows the propagation and reuse of more information through the cascade. For each spatial location, our model encodes a vector of information, instead of just a value, better representing the distribution of solutions. The architecture is trained in a end-to-end fashion, being its loss function

$$\mathcal{L}_{\mathrm{CADUF}} = \alpha\mathcal{L}_{\mathrm{D}} + \beta\mathcal{L}_{\mathrm{U}} + (1-\alpha-\beta)\mathcal{L}_{\mathrm{F}} \tag{4}$$

where $\alpha, \beta > 0$, $\alpha + \beta < 1$ and $\mathcal{L}_{\mathrm{D}}$, $\mathcal{L}_{\mathrm{U}}$ and $\mathcal{L}_{\mathrm{F}}$ are specific losses for sub-modules D_ϕ, U_ψ and F_ω, respectively. These specific losses will be defined later.

Let us now describe in detail each of the proposed sub-modules of the architecture.

A. *Extraction of motion-corrected features*

Following Eq. (3), the first issue to be addressed is the deconvolution of y by the blur kernel k^L to obtain $x \downarrow_s$. Because the deformations present in the image depend on the combination of the blur kernel k^L and the image, learning a single CNN that can cope with all these variations is a challenging task. For this reason, we propose to start from an initial noisy solution that can be calculated quickly using the Wiener filter:

$$y_w = \mathcal{F}^{-1}\left(\frac{\overline{\mathcal{F}(k^L)}\mathcal{F}(y)}{\overline{\mathcal{F}(k^L)}\mathcal{F}(k^L) + \epsilon}\right), \tag{5}$$

where $\mathcal{F}$ and $\mathcal{F}^{-1}$ denote the fast Fourier transform (FFT) and inverse FFT, respectively, and $\overline{\mathcal{F}}$ denotes the complex conjugate of $\mathcal{F}$, and ϵ is the regularization term.

As seen in Section III-E, the use of y_w as the starting point significantly outperforms using only the blur image y. However, the Wiener filter increases the noise in the image and introduces artifacts that are difficult to eliminate without the information of the blur image y, see Fig 3. To correct the image with a CNN, at each spatial location, we use the features extracted from y and y_w. However, because of the blurring processes, y and y_w are not aligned with each other. In this case, we can consider y_w as an anchor that indicates the correct position.

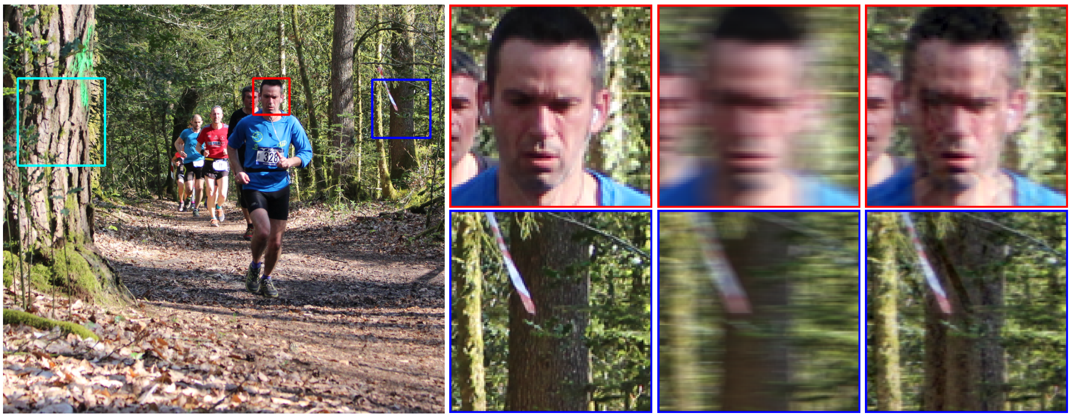


Fig. 3. Illustration of the artifacts originating from using the Wiener filter. Some produced artifacts can be compatible with natural images, being difficult to distinguish from real structures using only the filtered image.

To align the features extracted from both y and y_w, we propose the use of deformable convolution. First proposed in [38] and enhanced in [39] to handle more deformations, deformable convolutions are a modified convolution operation:

$$g^l(m) = \sum_{o=1}^{O} w_o a_{m,o} f^{l-1}(m + o + \Delta m_{m,o}), \tag{6}$$

where $g^l(m)$ denotes the feature vector in layer l at location m, o is the position of the convolutional kernel w, and $\Delta m_{m,o}$ and $a_{m,o}$ are offsets and scalar modulation, respectively. These parameters are calculated for each m location using another convolutional layer. Owing to this operation, our model can align the features of y and y_w at each spatial location, making further downstream calculations much easier.

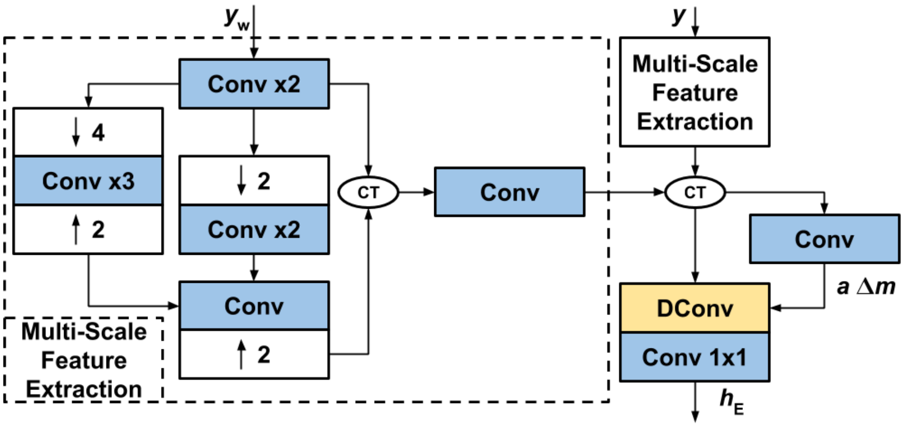


Fig. 4. Motion-corrected feature extraction sub-module. Conv is a convolution, and DConv is a deformable convolution. Each convolution is followed by a leaky-ReLU activation with 0.2 of negative slope and has a 3×3 kernel with the exception of the first and last convolutions that use a 5×5 and 1×1, respectively. Convolutions for scale 1 use 64 features while convolutions on scales $/2$ and $/4$ use 32.

Nevertheless, the calculation of parameters a and Δm is not an easy task: Movement of the camera can be quite strong, thus matching locations between y and y_w can be too far away. In these cases, a large receptive field is required. To obtain this, we extract features at scales 1, $/2$, and $/4$ using the architecture shown in Fig. 4. By doing so, we increase the receptive field to 37 pixels without the need to use a far deeper architecture.

This sub-module, E_θ, produces the features h_E that will be used by the remaining network. Although it could move the feature vectors anywhere, by propagating the loss of the next sub-modules in the cascade, it would have to learn to put then in the correct place, producing motion-corrected features.

B. LR-Anchor image estimation

According to Eq. (3), we start mapping the LR image y to an LR image y^*_D without blur or noise.

To define y^*_D, one approach can be defining it as $x\downarrow_s$. However, the difficulty of the SR task primarily depends on the blur kernel used before downsampling. Those cases with both low and high Gaussian blur led to significantly worse reconstruction. Large kernels in HR produce highly blurry LR images while small kernels in HR do not have a smooth downsampling form in LR, which introduces aliasing artifacts in the LR image. Aliased LR images can be very difficult to super-resolve because artifacts must be distinguished from real high-frequency patterns in the image.

Following the above arguments, we define y^*_{D} as the LR image obtained by downsampling x with a downsampling operator $\downarrow^{\mathrm{D}}_s$ that it is easier to invert by the upsampling network that we use. That is:

$$y^*_{\mathrm{D}} = x \downarrow^{\mathrm{D}}_s = [x \circledast k^{\mathrm{D}}] \downarrow_s, \tag{7}$$

where $\downarrow_s$ is the bicubic downsampling of factor s, and k^{D} is an isotropic Gaussian with σ 0.8 and 1.8 for scaling factors 2 and 4, respectively. These values have been found experimentally from within the range $[0.2, 4.0]$. Notice that the selection of these values can be seen as a form of "Privileged Information" [37]. The images $\{y^*_{\mathrm{D}}\}$ represent additional information available only during training and are used to estimate the optimal LR image. In this regard, y^*_{D} acts as an anchor that guides and regularizes the model training.

The features for the deblurring task are computed using a CNN-regression model that maps its input $\{h_{\mathrm{E}}, y_w\}$, the motion-corrected features, and the low-resolution Wiener deconvolved image, into y^*_{D}, the downsampled version of x. Let us denote by $\mathrm{D}_\phi(h_{\mathrm{E}}, y_w)$ the mapping defined by the CNN model. Then, $\mathrm{D}_{\hat{\phi}}(h_{\mathrm{E}}, y_w) : \{y_w\} \to \{y_{\mathrm{D}}\}$, where $\hat{\phi}$ represents the weights learned after training the model. The parameters can be optimized introducing the following loss $\mathcal{L}_{\mathrm{D}}$

$$\mathcal{L}_{\mathrm{D}} = \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}(\mathrm{D}_\phi(h_{\mathrm{E}}, y_w), y^*_{\mathrm{D}}), \tag{8}$$

where $\mathcal{L}(\cdot)$ is any loss function that estimates the differences between two images, like mean squared error (MSE). It can be observed D_ϕ computes a deblurring and denoising mapping for the degradation model defined in Eq. (3) because the low-resolution image y is mapped into an image without blur or noise, y_{D}, that is easier to super-resolve.

C. Initial high-res image estimation

After deconvolution, we are left with upsampling. This is equivalent to the problem defined in Eq. 2 when $n = 0$ and $\downarrow^{\mathrm{D}}_s = \downarrow_s$. Let us define the minimization of the upsampling process as follows:

$$\hat{\psi} = \arg\min_{\psi} \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}(\mathrm{P}_\psi(z), x), \tag{9}$$

where P_ψ is a CNN with parameters ψ, and z is the input that will be defined later. It should be noted that this approach is similar to the one used in other cascade methods [10], [8]. If the error introduced by the previous steps in the cascade is low, it is expected that it will not significantly affect this step. In this case, the simplest solution, therefore, would be to use as an upsampling sub-module a classic CNN for SISR such as SRResNet [29] to minimize Eq. 9. However, as shown by our experiments in Section III-E, this is not the best solution in practice. By minimizing Eq. 9, it has to solve two distinctive tasks: Inversing the operator $\downarrow^{\mathrm{D}}_s$ and removing the approximation error. The learning of the parameters is more difficult: Small differences can be due to approximation errors or different structures in the HR space. If not constrained, small changes in the LR image can lead to very different SR images, leading to an ill-conditioned model. This could result in the appearance of strong artifacts.

To use Eq. 9 to learn only the inverse of $\downarrow_s^{\mathrm{D}}$, we constrained the predicted image x_{U} to those solutions that satisfy,

$$x_{\mathrm{U}} \downarrow_s^{\mathrm{D}} = y_{\mathrm{D}}, \tag{10}$$

through the use of the projection introduced in [40]. Let us define A as a degradation operator such that $Ax = x \downarrow_s^{\mathrm{D}}$ and A^+ as the Moore-Penrose pseudoinverse [41] of A (refer to Section III-B to see the estimation of A^+). In our case, A is an affine projection implemented as a stride convolution, and A^+ is implemented using a convolutional transpose layer. For a given $\mathrm{P}_\psi(z)$, a new transformation $\mathrm{U}_\psi(z)$ can be defined as,

$$\mathrm{U}_\psi(z) = (I - A^+A)\mathrm{P}_\psi(z) + A^+ y_{\mathrm{D}}, \tag{11}$$

such that if no noise is present because $AA^+A = A$ and A is a full row rank matrix, that is, $AA^+ = I$, then,

$$A\mathrm{U}_\psi(z) = A(I - A^+A)\mathrm{P}_\psi(z) + AA^+ y_{\mathrm{D}} = 0 + y_{\mathrm{D}}. \tag{12}$$

Therefore, if $x_{\mathrm{U}} = \mathrm{U}_\psi(z)$, Eq. 10 is always satisfied.

Owing to the loss of information caused by approximation errors, the information of y_{D} contained in h_{D} is not sufficient to properly invert $\downarrow_s^{\mathrm{D}}$. Our model needs information of the original y, which could not be preserved in h_{D}. We add this information by concatenating to h_{D} the motion-corrected features h_{E}, obtaining h_{ED}. Note that this would not be possible without motion-correction because the features h_{D} will not align with features extracted from the blurred image y. Therefore, we define the input as $z = \{h_{\mathrm{ED}}, y_{\mathrm{D}}\}$. Using these inputs and the affine projection, we obtain a specialized SR sub-module by replacing P_ψ for U_ψ and introduced the following loss to optimized its parameters:

$$\mathcal{L}_{\mathrm{U}} = \frac{1}{N}\sum_{i=1}^{N} \mathcal{L}(\mathrm{U}_\psi(h_{\mathrm{ED}}, y_{\mathrm{D}}), x). \tag{13}$$

D. High-res image refinement

Compared to previous SISR cascade models, ours adds an improvement in the form of a last fusion block fed with the characteristics of all the blocks in the cascade (see Fig. 2) focused on removing the accumulated error. This last block is the result of distinguishing two tasks in the previous step: Inverting the downsampling operator $\downarrow_s^{\mathrm{D}}$, in addition to removing the accumulated error e introduced by the previous cascade sub-modules,

$$x = x_{\mathrm{U}} + e. \tag{14}$$

Note that e can be seen as being structured noise dependent not only on y, but also on previous sub-modules E_θ, D_ϕ, and U_ψ. Because U_ψ is constrained, e will not introduce strong artifacts when the error introduced by D_ϕ is small (as shown in Section III-E). Therefore, we argue that a last refinement sub-module F_ω, when fed with the features h_{EDU} calculated by each one of the specialized sub-modules, E_θ, D_ϕ, and U_ψ, can learn to calculate a residual that, when added to x_{U}, compensates for the accumulated errors e in the cascade. We implemented F_ω using a CNN regression model. The learning of the parameters is performed using the following loss function:

$$\mathcal{L}_{\mathrm{F}} = \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}(\mathrm{F}_{\omega}(h_{\mathrm{EDU}}, x_{\mathrm{U}}), x). \tag{15}$$

E. D_{ϕ}, U_{ψ} *and* F_{ω} *architecture*

The architecture that implements our sub-modules D_{ϕ}, U_{ψ}, and F_{ω} is shown in Fig. 5. As seen, it is a modification of the residual architecture SRResNet proposed in [42]. We remove the batch normalization layers as suggested in [5] and replace the ReLU activation with leaky ReLU with 0.2 negative slope. To further increase the receptive field, the first convolution in each residual block is a dilated convolution [43] with the dilatation parameter set to 2. The architecture is fed with features h calculated by the previous sub-modules. The first two convolutional layers transform the input features. Then, Nb residual blocks are used to produce the new set of features $\hat{h}$. To calculate the output image $\hat{z}$, we use $\hat{h}$ together with a dynamic filter network (DFN) [44]. The DFN predicts c, which is a collection of filters, one for each spatial location in the output image. Using the approximation proposed in [45], $\hat{z}$ is calculated as,

$$\hat{z}_{i,j} = r_{i,j} + \sum_{l=-p}^{p}\sum_{m=-p}^{p} c_{i,j,l,m} * z_{i+l,j+m}, \tag{16}$$

where z is the output image calculated by the previous sub-module, i, j is a spatial location, and r is a residual image calculated by a CNN using $\hat{h}$. We use a value of $p = 2$. By using this approximation, our model can filter spatially-variant artifacts present in the outputs of previous sub-modules. Then, the filtered outputs can be used to perform residual learning [46], [47] more effectively. These artifacts are caused by approximation errors in the image formation model and the calculation. Notice that a normal CNN can also learn to remove these artifacts, but it would require far more parameters because CNN filters are spatially-invariant.

In each convolution, except in the input convolutions of each sub-module and the two output convolutions, kernels of size 3×3 and 64 filters are used. The input convolutions use kernels of size 1×1 to fuse the features of the previous sub-modules. In our experiments, the number of residual blocks Nb was set to eight for D_{ϕ}, 16 for U_{ψ}, and F_{ω} uses only four. In the case of U_{ψ} and F_{ω}, both c and r are upsampled by scaling factor s using sub-pixel convolutions [33]. The sub-pixel convolution is located after the last convolution for c and before in the case of r. The first two convolutions of D_{ϕ} are skipped because E_{θ} already provides a suitable set of features.

III. Experiments

A. k^{*L} *estimation*

To train our models, for each k in Eq. (1), we need to have k^{*L} for Eq. (3) such as $k^{*L} = \arg\min_{k^L} ||[x \circledast k] \downarrow_s - x \downarrow_s \circledast k^L||^2_F$,, that is, k^{*L} minimizes the difference between both degradation models. To calculate k^{*L}, we use a neural network with one hidden layer with 2048 neurons. This network L_{ζ} with parameters ζ is trained by minimizing $||x \circledast k] \downarrow_s - x \downarrow_s \circledast \mathrm{L}_{\zeta}(k)||^2_F$. We use the Adam optimizer for 60 epochs with $\mathrm{lr} = 10^{-4}$ and weight decay set to 10^{-4}.

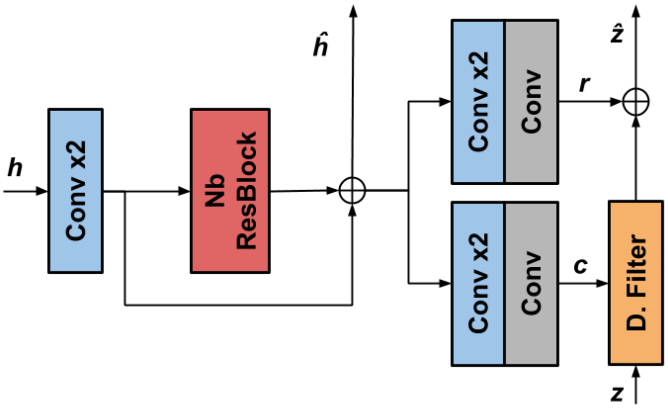


Fig. 5. Architecture used for sub-modules D_ϕ, U_ψ, and F_ω. Conv indicates a convolutional layer and D. Filter the correlation of kernels c with an input image z. We use Nb residual blocks [42], each composed of two convolutions after removing the batch normalization layers, as suggested in [5]. Each convolution uses a 3×3 kernel size, with the exception of the first layer that uses 1×1 kernels to fuse the previous sub-module features. With the exception of the output convolutions, all are followed by a leaky ReLU activation with a negative slope of 0.2. We skip the first two convolutions for D_ϕ because E_θ already provides a suitable set of features. In the case of U_ψ and F_ω, sub-pixel convolutions [33] of scaling factor s are used to produce c and r with the correct output size (see the text for further details).

TABLE I

KEY COMPONENT ANALYSIS OF THE PROPOSED MODEL FOR SCALE FACTOR 4 ON THE GAUSSIANSM TESTING DATASET (IMAGES FROM BSD100 [48], URBAN100 [3] AND MANGA109 [49] DATASETS). BEST AND SECOND-BEST RESULTS APPEAR IN BOLD AND UNDERLINED, RESPECTIVELY.

1	2	3	4	5	6	7	8	9	10
Model	$\mathrm{P}_\psi(y)$	$\mathrm{P}_\psi(y, \mathrm{PCA}(k))$	$\mathrm{P}_\psi(y_w)$	$\mathrm{P}_\psi\mathrm{D}_\phi(y_w)$-no-PI	$\mathrm{P}_\psi\mathrm{D}_\phi(y_w)$	$\mathrm{F}_\omega\mathrm{U}_\psi\mathrm{D}_\phi(y_w)$	$\mathrm{F}_\omega\mathrm{U}_\psi\mathrm{D}_\phi(y, y_w)$	CNN-Cascade	CADUF
PSNR/SSIM	26.63-0.773	27.06-0.780	27.09-0.785	27.24-0.789	27.37-0.792	27.51-0.795	27.53-0.796	27.25-0.788	**27.90-0.806**

B. A^+ calculation

To use the affine projection proposed in Section II-C, we implement the A^+ operator using a CNN with two convolutional layers with 32 filters followed by leaky ReLU ($\alpha = 0.2$) activation and a final sub-pixel convolution. The sizes of the kernels are 7×7, 5×5, and 3×3, respectively. Following the approach in [40], we train this network by minimizing the following loss function with a stochastic gradient descent, that is,

$$\begin{aligned}\hat{\mu} =& \arg\min_{\mu} \mathbb{E}_x \| Ax - AA^+_\mu(Ax) \|_2^2 \\ &+ \mathbb{E}_y \| A^+_\mu(y) - A^+_\mu(AA^+_\mu(y)) \|_2^2,\end{aligned} \tag{17}$$

where A is the degradation operator and A^+_μ is a CNN of the parameters μ. We stop the optimization when the loss value is less than 10^{-7}. During the training of our main network, we keep μ fixed.

C. Datasets

The training dataset is formed by images from the DIV2K [50] and Flickr2K [51] datasets. The ground-truth of our training consists of 3450 high-quality 2K images. During training, random patches cropped from the images of the dataset were extracted. The size of each HR patch is

$s48 \times s48$, where s is the scaling factor. The LR images were synthesized according to Eq. (1). For testing, we used images from standard SR test image datasets BSD100 [48], Urban100 [3], and Manga109 [49]. To ensure that a wide variety of degradations are present in the test, we generated three LR images for each of the HR images.

The blur kernels used to generate the LR images cause a uniform blur and are a combination of an isotropic Gaussian blur kernel and a motion blur kernel. We simulate two different motion blur kernels representing simple and complex camera movements. In the simple case, only simple linear movement is considered. Meanwhile, the motion blur kernels for the complex case simulate more complex camera movements with curved trajectories. They are generated using the method in [52] with $T = 0.8$ and anxiety $10^R/1000$, where R is a random number from a uniform $[0, 1]$ distribution. Furthermore, in this scenario, AWGN of standard deviation $\sigma = 0.01$ is added to the LR image. The combination of Gaussian blur and the motion blur kernels of each scenario produces two sets of blur kernels that we use in our experiments. Fig.1 in Columns (1,3,6) and (2,4,5,7) show blur kernel examples of the simple and complex case respectively. We refer to these two sets of kernels, GaussianSM and GaussianCM, respectively.

In the case of GaussianSM, the σ of the Gaussian blur is sampled randomly during training from the range [0.2, 2.0] and [0.2, 4.0] for scaling factors 2 and 4, respectively. The motion blur kernels of GaussianSM are sampled with angles in the $[0, 180)$ range and length in the range $[1, 9]$ and $[1, 15]$ for scaling factors 2 and 4, respectively. GaussianCM consists of 640 blur kernels with sizes between 11×11 and 45×45 pixels that are combined with a Gaussian blur with σ in the range [0.2, 1.0] and [0.2, 2.0] for factors 2 and 4, respectively. We used 540 of these kernels for training and 100 for testing.

D. Model training

We train our models using the Adam optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, weight decay of 10^{-4}, and a batch size of 64. For each epoch, we sample 3000 batches. We define L as the Charbonnier loss [53], as we found it to be more stable than MSE. The Charbonnier loss between two images u and v is $\gamma(u, v) = \sum_i \sum_j \sqrt{(u_{i,j} - v_{i,j})^2 + \varepsilon^2}$, where $\varepsilon = 10^{-3}$. We apply data augmentation by randomly performing horizontal and vertical flips, 90-degree rotations, and random scaling with a factor between [0.5, 1.25]. The parameter of the Wiener filter is set as the minimum between 0.001 and the estimated standard deviation of the noise in the images. All of our models use the RGB images as input and do not require any additional preprocessing step. The implementation was performed using Pytorch [54][1].

Our training method consists of two distinctive phases: A) An initialization phase where we train our model using much higher $\alpha = 0.6$ and $\beta = 0.3$, forcing the model to first converge to good solutions for the intermediary problems of an optimal LR image and initial high-res image estimation. Although it is possible to train the proposed model without this initial step, we have found that this initialization helps stabilize the model and makes it converge earlier. We trained for 20 epochs with $\mathrm{lr} = 10^{-4}$.

[1] The code and trained models shown in this work will be made available upon acceptance of the paper at https://github.com/vipgugr/CADUF.

B) Main training phase. We set $\alpha = 0.1$ and $\beta = 0.1$ and train our model for 120 epochs. The learning rate was set to 10^{-4} for the first 90 epochs and then set to 10^{-5} for the remaining 30 epochs.

E. Ablation study

In this section, we conducted experiments to determine the contribution of each component of the proposed CADUF. All models have 28 residual blocks and are trained as described in Section III-D. The experiments were conducted for scale factor 4 and using GaussianSM training and testing sets (see Section III-C),with the difference that the blurs used were sampled with σ in the range of [0.2, 3.0) and length of $[1, 11]$. This was done to ease the training on the simpler models because using strong blurs makes it unstable. Table I contains the results for this study in terms of the PSNR and SSIM.

The base model $\mathrm{P}_{\psi}(y)$ is a single CNN model that uses the architecture described in Section II-E (see Fig 5), where h is the features extracted from y with two convolutional layers and $z = y$. This model shows the performance that can be expected from a classic SR CNN model. We improve the model performance by progressively adding each main component of the CADUF.

The first improvement of 0.46dB-0.12(SSIM) is achieved by introducing the k^L blur information into the network using the Weiner-filtered image y_w instead of y ($\mathrm{P}_{\psi}(y_w)$) (Columns 2 and 4 in Table.I). Facilitating the blur information eases the network task because the blur no longer needs to be identified. Alternatively, Zhang et al. [9] proposed using PCA to encode the information of the blur kernel k and use it alongside the blur image y. We also test this approximation, making the model $\mathrm{P}_{\psi}(y, \mathrm{PCA}(k))$ (Column 3 in Table.I). Despite the noise and artifacts introduced by the Weiner filter and the approximation error of using Eq. (3), $\mathrm{P}_{\psi}(y_w)$ slightly outperforms $\mathrm{P}_{\psi}(y, \mathrm{PCA}(k))$. Moreover, the use of PCA requires its training with the possible blurs k, which harms the generalization capabilities of the model to unseen blurs.

The second improvement is the decomposition of $\mathrm{P}_{\psi}(y_w)$ into $\mathrm{P}_{\psi}\mathrm{D}_{\phi}(y_w)$ (Columns 4 and 6 in Table.I). By specializing part of the model in the denoising and deconvolution of the LR image, we can see an increase in performance of 0.28dB-0.007(SSIM) despite the use of the same number of residual blocks (a total of 28 residual blocks, eight residual blocks for D_{ϕ} and 20 for P). We also tested the contribution of using privileged information (PI) [37] during training. By training D_{ϕ} to map the input to $x \downarrow_s$ instead of $x \downarrow_s^{\mathrm{D}}$, we obtain a new model that does not make use of PI. We call this model $\mathrm{P}_{\psi}\mathrm{D}_{\phi}(y_w)$-no-PI. As seen in Columns 5 and 6 of Table I, the use of PI is a key element of the proposed CADUF because omitting it significantly deteriorates the performance (0.13dB-0.003(SSIM)).

Next, we use the approximation proposed in Section II-C to separate P_{ψ} into $\mathrm{F}_{\omega}\mathrm{U}_{\psi}$, obtaining the model $\mathrm{F}_{\omega}\mathrm{U}_{\psi}\mathrm{D}_{\phi}(y_w)$ (Column 7-Table I). Using the same number of residual blocks (28), $\mathrm{F}_{\omega}\mathrm{U}_{\psi}\mathrm{D}_{\phi}(y_w)$ outperforms $\mathrm{P}_{\psi}\mathrm{D}_{\phi}(y_w)$ (Column 6-Table I) by 0.14dB-0.003(SSIM). This is in line with our hypothesis that the decomposition of the task of P_{ψ} into two sub-tasks, upsampling, and error correction, simplifies the optimization and allows for better minima. It can also be observed that, owing to the introduction of prior information and the use of constraints, the concatenation

TABLE II

COMPARISON WITH NON-BLIND SOTA ON THE BSD100 [48], URBAN100 [3] AND MANGA109 [49] DATASETS FOR THE BLUR KERNELS IN GAUSSIANSM AND GAUSSIANCM DATASETS. TO ENSURE FAIRNESS IN THE COMPARISON, THE METHODS INDICATED WITH "*" WERE RETRAINED. THE AVERAGE NUMBER OF MACS PER IMAGE OF EACH METHOD IS ALSO REPORTED. BEST AND SECOND-BEST RESULTS ARE BOLD AND UNDERLINED, RESPECTIVELY.

Scale	Model	PSNR-SSIM						MACs
		GaussianSM			GaussianCM			
		BSD100	Urban100	Manga109	BSD100	Urban100	Manga109	
×2	Bicubic	26.21-0.692	23.58-0.685	25.54-0.809	23.24-0.543	20.68-0.529	21.28-0.655	$2.02 * 10^{7}$
	ZSSR [34]	26.55-0.717	23.81-0.711	25.76-0.810	24.11-0.556	21.79/0.591	21.92-0.664	$4.70 * 10^{13}$
	RCAN [56]*	30.85-0.874	29.22-0.882	34.27-0.956	25.13-0.640	22.61-0.639	24.40-0.775	$2.46 * 10^{12}$
	IRCNN [6]+RCAN [56]	27.08-0.759	24.66-0.751	27.92-0.879	26.23-0.712	24.30-0.732	27.01-0.846	$2.70 * 10^{12}$
	DeblurGAN+RCAN [56]	26.21-0.702	23.57-0.694	25.54-0.815	23.99-0.573	21.55-0.568	22.86-0.706	$2.47 * 10^{12}$
	SRMDNF [9]*	30.83-0.873	29.42-0.884	34.94-0.957	25.99-0.695	24.01-0.712	26.36-0.825	$\mathbf{2.60 * 10^{11}}$
	SFTMD [10]*	31.57-0.889	30.62-0.907	36.79-0.969	27.26-0.742	25.66-0.774	28.97-0.880	$9.71 * 10^{11}$
	DPSR [8]	28.37-0.793	27.11-0.815	33.19-0.925	27.23-0.740	25.89-0.790	30.11-0.895	$4.86 * 10^{12}$
	CADUF	**31.92-0.893**	**31.11-0.914**	**37.70-0.972**	**27.92-0.761**	**26.64-0.805**	**30.73-0.906**	$6.56 * 10^{11}$
×4	Bicubic	24.19-0.582	21.51-0.564	22.52-0.698	23.05-0.533	20.40-0.511	20.99-0.642	$1.68 * 10^{7}$
	ZSSR [34]	24.76-0.589	21.83-0.573	22.88-0.709	23.32-0.545	20.90-0.520	21.45-0.652	$2.14 * 10^{13}$
	RCAN [56]*	26.92-0.713	24.86-0.736	28.80-0.877	24.80-0.610	22.35-0.612	24.13-0.756	$6.38 * 10^{11}$
	IRCNN [6]+RCAN [56]	24.71-0.596	22.18-0.587	23.60-0.714	24.19-0.567	21.87-0.565	23.21-0.691	$6.98 * 10^{12}$
	DeblurGAN+RCAN [56]	23.48-0.550	21.34-0.538	22.31-0.663	22.11-0.498	20.59-0.494	21.35-0.618	$6.40 * 10^{11}$
	SRMDNF [9]*	26.82-0.708	24.76-0.729	28.53-0.870	25.31-0.632	23.10-0.649	25.45-0.793	$\mathbf{6.71 * 10^{10}}$
	SFTMD [10]*	27.12-0.721	25.25-0.752	29.42-0.889	25.67-0.647	23.62-0.674	26.46-0.821	$2.82 * 10^{11}$
	DPSR [8]	25.06-0.641	23.47-0.686	27.35-0.842	24.79-0.629	22.51-0.656	24.95-0.804	$1.33 * 10^{12}$
	CADUF	**27.41-0.729**	**25.64-0.768**	**30.05-0.900**	**25.73-0.652**	**23.82-0.689**	**26.78-0.835**	$2.03 * 10^{11}$

of D_ϕ and U_ψ does not output complex artifacts that are difficult to filter out by F_ω. As shown in [55], CNN architectures introduce deep priors, which in some applications explain a large portion of the model efficiency. However, in our case, the ablation study shows that it is the specific training, with the introduction of $\mathrm{F}_\omega\mathrm{U}_\psi$ and privileged information, that is the primary factor responsible for the model efficiency.

We also tested the proposed mechanism to connect all the sub-modules of the CADUF. To do that, we develop a new model by feeding each sub-module only with the image z calculated by the previous one (not the features h) and omitting the reuse of z with dynamic filtering. By doing this, we obtain a classic CNN-cascade model similar to those used in the literature [36], [10], a concatenation of a deblur and upsampling CNN. This model, which we call CNN-Cascade (Column 9 in Table.I), obtains results far worse than $\mathrm{F}_\omega\mathrm{U}_\psi\mathrm{D}_\phi(y_w)$ and even $\mathrm{P}_\psi\mathrm{D}_\phi(y_w)$ (0.26dB-0.007(SSIM) and 0.12dB-0.004(SSIM), respectively). As we argue in Section II, the difference in performance is because the use of the features of previous sub-modules allows for better information propagation through the cascade.

Finally, we evaluate the contribution of our proposed motion-corrected feature extraction sub-module, E_θ, appending the information extracted from y and y_w together. Overall, it constitutes the full CADUF model (Column 10 in Table.I). Compared to the concatenation of y and y_w ($\mathrm{F}_\omega\mathrm{U}_\psi\mathrm{D}_\phi(y, y_w)$) (Column 8 in Table.I), the use of E_θ significantly increases the performance (0.37dB-0.010(SSIM)) with little overhead computation.

F. SOTA Comparison

In this section, we compare our CADUF model against the SOTA methods in SISR dealing with multiple degradations. RCAN [56] is a very deep CNN developed for SISR that does not use any information about the degradation but achieves a very high score. In contrast, SRMDNF [9] and SFTMD [10] incorporate information of the kernel k into the network by encoding it using PCA, but obtaining lower performance than RCAN [56]. We also show the results of combining SOTA deconvolution methods with an SISR CNN in a cascade. Specifically, we compared two combinations, both using deconvolution followed by upsampling: IRCNN [6]+RCAN [56] and DeblurGAN+RCAN [56]. Finally, we also compare the MBO approaches ZSSR [34] and DPSR [8].

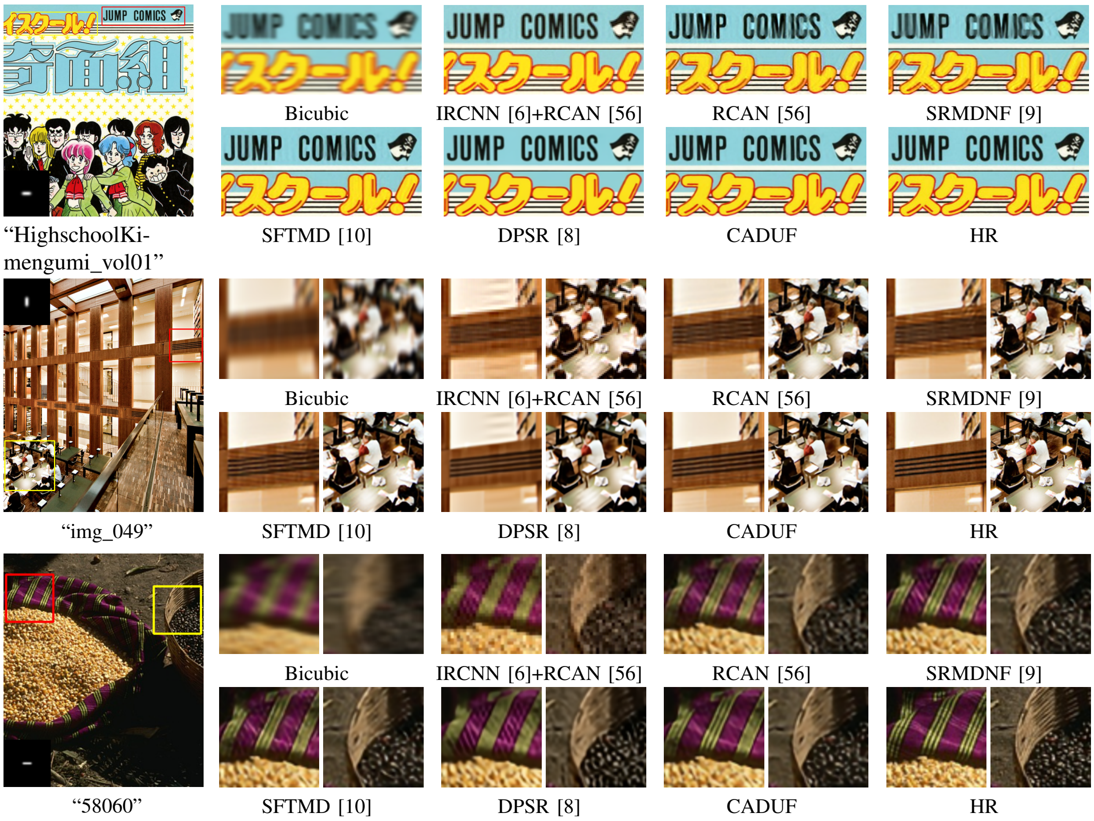
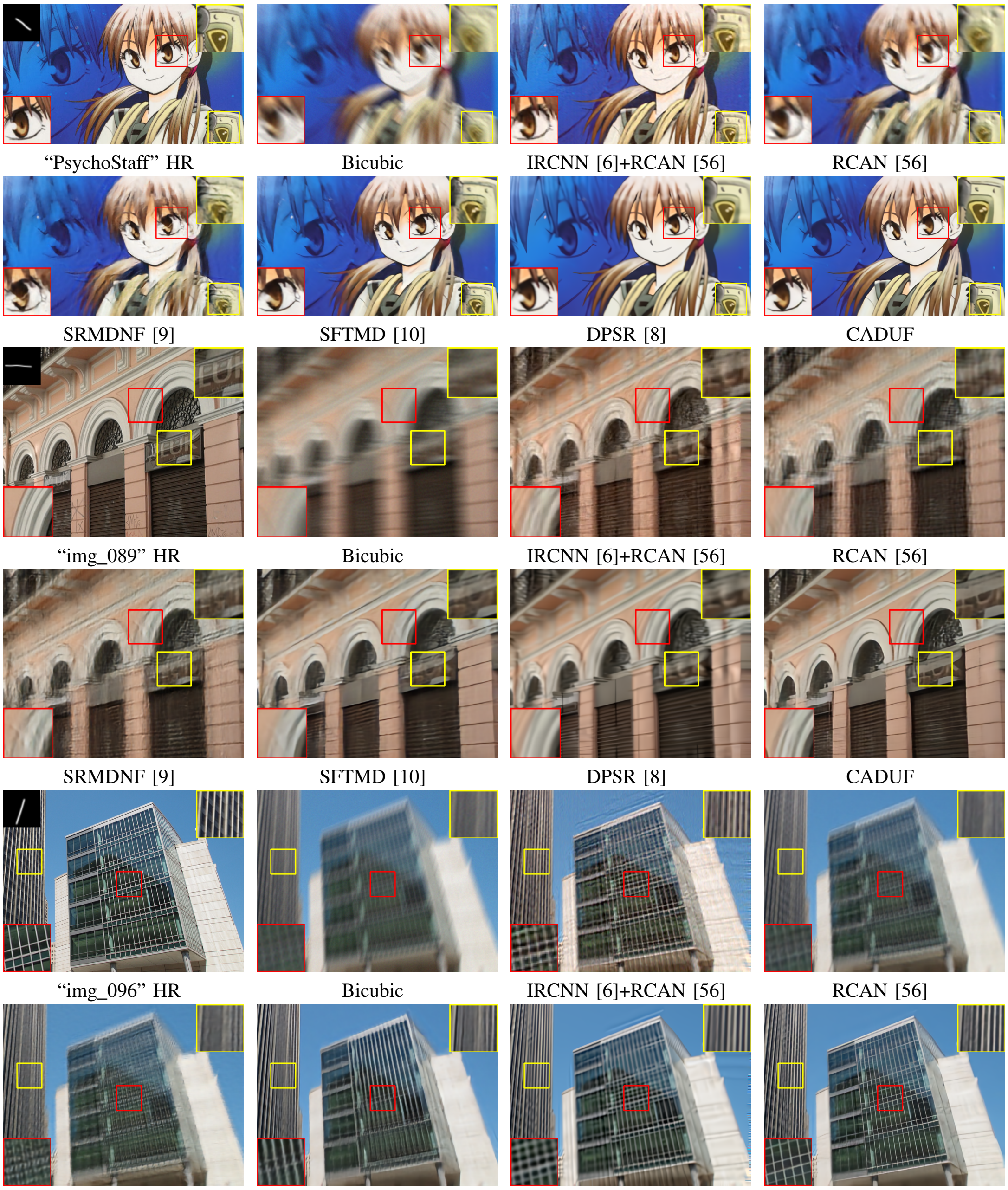


Fig. 6. Comparison of IRCNN [6]+RCAN [56], RCAN [56], SRMDNF [9], SFTMD [10], DPSR [8] and CADUF on GaussianSM dataset for factor 2 on image "HighschoolKimengumi_vol01" from Managa109 [49] dataset, image "img_049" from Urban100 [3] dataset and image "58060" from BSD100 [48] dataset.

We test all models using the test datasets GaussianSM and GaussianCM described in Section III-C. To ensure fairness in the comparison, we not only train our model for each setting using the appropriate training set, but also do the same for competing models that require it. Therefore, we trained RCAN [56], SRMDNF [9], and SFTMD [10]. The training and evaluation of the competing methods was performed using the code provided by the original authors.

Table II shows the results of our CADUF model and competing models in the non-blind scenario for GaussianSM and GaussianCM kernels datasets and scaling factors 2 (x2) and 4 (x4). We evaluate the performance of the methods using the PSNR and SSIM metrics. Furthermore,



“PsychoStaff” HR Bicubic IRCNN [6]+RCAN [56] RCAN [56]

SRMDNF [9] SFTMD [10] DPSR [8] CADUF

“img_089” HR Bicubic IRCNN [6]+RCAN [56] RCAN [56]

SRMDNF [9] SFTMD [10] DPSR [8] CADUF

“img_096” HR Bicubic IRCNN [6]+RCAN [56] RCAN [56]

SRMDNF [9] SFTMD [10] DPSR [8] CADUF

Fig. 7. Visual comparison of IRCNN [6]+RCAN [56], RCAN [56], SRMDNF [9], SFTMD [10], DPSR [8] and CADUF on GaussianCM dataset for factor 2 on image “PsychoStaff” from Managa109 [49] dataset and images “img_089” and “img_096” from Urban100 [3] dataset.

we also include a study of the time complexity of each method using the average number of multiplication and additions (MACs) per image as a metric. The smaller the number of MACs, the faster the algorithm performance.

As seen in Table II, compared to other LB methods, CADUF performs significantly better in all datasets and scenarios. In GaussianSM, CADUF obtains an average improvement on the datasets of 2.13dB-0.02(SSIM) in x2 and 0.84dB-0.02(SSIM) in x4 over RCAN [56] and 0.58dB-

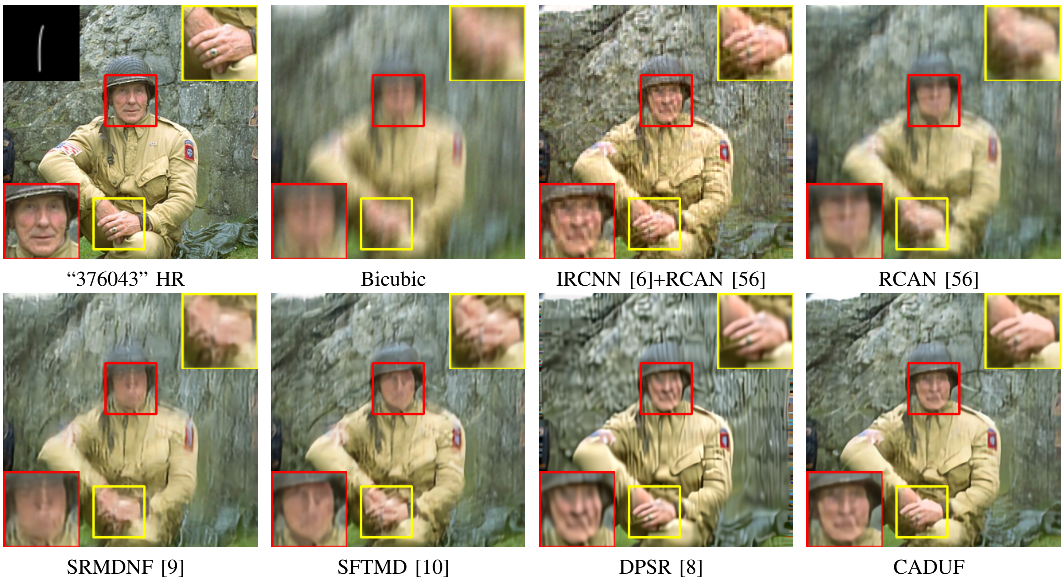
TABLE III
COMPARISON WITH BLIND SOTA ON THE BSD100 [48], URBAN100 [3] AND MANGA109 [49] DATASETS FOR THE BLUR KERNELS IN THE GAUSSIANCM DATASET. TO ENSURE FAIRNESS IN THE COMPARISON, THE METHODS INDICATED WITH "*" WERE RETRAINED. BEST AND SECOND-BEST RESULTS ARE BOLD AND UNDERLINED, RESPECTIVELY.

Scale	Model	PSNR-SSIM		
		BSD100	Urban100	Manga109
×2	Bicubic	23.24-0.543	20.68-0.529	21.28-0.655
	RCAN [56]*	25.13-0.640	22.61-0.639	24.40-0.775
	DeblurGAN+RCAN [56]	23.99-0.573	21.55-0.568	22.86-0.706
	IKC [10]+SFTMD [10]*	18.27-0.451	16.29-0.456	17.50/0.585
	DPSR [8]	23.56-0.678	20.32-0.652	23.57-0.825
	Our Model	**25.65-0.698**	**23.27-0.674**	**25.06-0.847**

0.01(SSIM) in x2 and 0.44dB-0.01(SSIM) in x4 over SFTMD [10], being significantly faster. The performance difference increases for the GaussianCM dataset. In this case, the figures show improvements of 4.38dB-0.14(SSIM) in x2 and 1.68dB-0.06(SSIM) in x4 over RCAN [56] and 1.14dB-0.03(SSIM) in x2 and 0.19dB-0.01(SSIM) in x4 over SFTMD [10].

The difference between CADUF and the other best performing methods in GaussianSM is more apparent if we perform a closer inspection of the produced images in Fig. 6. As shown, when uniform motion blur is present in the image, CADUF is able to recover fine details in the image as well as the other best performing method, SFTMD [10], but without introducing ghosting and other artifacts. Despite being a blind method, RCAN [9] is able to compete with the remaining methods in GaussianSM, obtaining similar results to other non-blind methods such as SRMDNF [9]. However, its performance drops significantly for the more complex blurs in GaussianCM, showing that even a CNN as complex as RCAN [9] is not able to cope with the degradation variance of a more complex realistic scenario without the introduction of specialized elements. In this case, although SRMDNF [9] and SFTMD [10] do not suffer as much as RCAN [9], they show a more significant drop in performance than CADUF and DPSR [8]. We believe that this is because both rely on PCA to encode blur information. Since the variability of the blur is much higher, it is more difficult to accurately depict new blur kernels at test time. Furthermore, both models rely on standard convolutional layers that are not well suited for recovering blurs caused by strong camera motions. In contrast, the use of the motion-corrected feature extraction sub-module and the Wiener filter allows our model to not only adapt to blur introduced by more complex camera motions but also to better generalize and be more robust to blurs not seen in training. These features are not shared by the other two cascade models IRCNN [6]+RCAN [56] and DeblurGAN+RCAN [56]. Both produce worse results than RCAN [56] and cannot compete with the remaining CNN methods in terms of performance and time consumption. The accumulated errors in the cascade and the strong artifacts that appear due to it are the source of this poor performance.

Compared to MBO methods (ZSSR [34] and DPSR [8]), we can see that CADUF requires orders of magnitude fewer MACs because it does not need to solve an expensive optimization problem for each new test image. Although our model's ability to resolve unseen blurs is lower, it generalizes well to new degradations close to those observed in training, that is, a similar type of noise and blur (motion, defocus, downsampling). We can see that our model significantly outperforms both ZSSR [34] and DPSR [8], with the latter being the closest in terms of performance of the other two. Fig. 7[2] shows a visual comparison of SOTA methods and CADUF for testing images of the GaussianCM dataset. As seen, compared to the other two best-performing models, SFTMD [10] and DPSR [8], CADUF produces significantly better images, closer to the HR image and with far fewer artifacts. Especially remarkable is the case of image "img096" of the Urban100 [3] dataset, where CADUF is able to reconstruct the building windows on the center of the image far better than the other methods. Despite the good performance shown by CADUF, when very strong blurs are present, it would fail to properly recover the HR image. One example of these cases is shown in Fig. 8. The CADUF method is unable to recover the face of the soldier and produces some easily seen artifacts in the background wall and the soldier helmet. However, CADUF still produces better results than the other SOTA methods, introducing significantly fewer artifacts and better recovering the hands of the soldier and more details of the face. When making a comparative visual study of our model against the SOTA, we found that it produces competing results when the blur does not include too much motion and outperforms other models when strong motion is present.


"376043" HR

Bicubic

IRCNN [6]+RCAN [56]

RCAN [56]

SRMDNF [9]

SFTMD [10]

DPSR [8]

CADUF

Fig. 8. Failure case of CADUF for factor 2 on image "376043" from the BSD100 [48] dataset with degradation from the GaussianCM dataset.

Finally, in Table III, we compare our method against competing methods in the blind setting in the GaussianCM dataset. We estimate the LR blur kernel k^{*L} for algorithms that require it (DPSR [8] and ours) using the method in [57]. In our model case, we fine-tuned it for 100 epochs with lr=10^{-5}. We report results for factor 2 because the loss of information coupled with errors during the degradation estimation makes it difficult to analyze the differences between the methods for factor 4. As seen, the proposed model has a significantly less drop in performance compared to DPSR [8] when an approximation of k^{*L} is used. In this case, our model benefits of the

[2] More examples can be downloaded at https://github.com/vipgugr/CADUF.

possibility to fine-tune and adapt to the errors in the blur kernel estimation. Thanks to this fact, it still significantly outperforms all other methods (2.18dB-0.02(SSIM) and 0.61dB-0.06(SSIM) over DPSR [8] and RCAN [56], respectively).

IV. Conclusions

In this study, a new cascade CNN model for SISR capable of dealing with multiple degradations, has been proposed. The model exhibits a new approach for searching for a solution with several differentiating keys. First, cascade sub-modules are explicitly specialized in specific tasks: motion-corrected feature extraction, deblur, and upsampling. Second, a new way of regularizing the deblur and upsampling sub-modules is introduced from domain knowledge. As a consequence of the upsampling restriction using affine projection [40], a final sub-module that corrects the accumulated cascade error can be introduced to further increase the model performance. As far we know, the use of domain knowledge to constrain the training of modules in a cascade represents a novelty in SISR. Finally, the model uses an improved method of connecting the sub-modules in the cascade where each sub-module re-uses the features calculated by all previous sub-modules. As a result, our model is fast, robust, and accurate. The experiments were performed using two new SISR datasets, containing degradations with simple and complex camera movement. These experiments show that CADUF significantly outperforms CNN-BMs with architectures of the same depth. Furthermore, CADUF outperforms the remaining compared SOTA methods for non-blind and blind SISR while being significantly faster. Although CADUF is a CNN-BM, it can be generalized to unseen blurs during training, competing in this regard with MBO models.

References

[1] C.-Y. Yang, C. Ma, and M.-H. Yang, “Single-image super-resolution: A benchmark,” in *ECCV*, 2014.

[2] C. Dong, C. C. Loy, K. He, and X. Tang, “Learning a deep convolutional network for image super-resolution,” in *European Conference on Computer Vision*, pp. 184–199, Springer, 2014.

[3] J. Huang, A. Singh, and N. Ahuja, “Single image super-resolution from transformed self-exemplars,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5197–5206, June 2015.

[4] J. Kim, J. K. Lee, and K. M. Lee, “Accurate image super-resolution using very deep convolutional networks,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1646–1654, June 2016.

[5] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, “Enhanced deep residual networks for single image super-resolution,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, vol. 1, p. 3, 2017.

[6] K. Zhang, W. Zuo, S. Gu, and L. Zhang, “Learning deep CNN denoiser prior for image restoration,” in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2808–2817, July 2017.

[7] X. Wang, K. Yu, C. Dong, and C. Change Loy, “Recovering realistic texture in image super-resolution by deep spatial feature transform,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 606–615, 2018.

[8] K. Zhang, W. Zuo, and L. Zhang, “Deep plug-and-play super-resolution for arbitrary blur kernels,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[9] K. Zhang, W. Zuo, and L. Zhang, “Learning a single convolutional super-resolution network for multiple degradations,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[10] J. Gu, H. Lu, W. Zuo, and C. Dong, “Blind super-resolution with iterative kernel correction,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[11] A. K. Katsaggelos, R. Molina, and J. Mateos, “Super resolution of images and video,” *Synthesis Lectures on Image, Video, and Multimedia Processing*, vol. 1, no. 1, pp. 1–134, 2007.

[12] Y. He, K.-H. Yap, L. Chen, and L.-P. Chau, "A soft map framework for blind super-resolution image reconstruction," *Image and Vision Computing*, vol. 27, no. 4, pp. 364 – 373, 2009.

[13] Y. Tai, S. Liu, M. S. Brown, and S. Lin, "Super resolution using edge prior and single image detail synthesis," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 2400–2407, June 2010.

[14] S. P. Belekos, N. P. Galatsanos, and A. K. Katsaggelos, "Maximum a posteriori video super-resolution using a new multichannel image prior," *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1451–1464, 2010.

[15] S. D. Babacan, R. Molina, and A. K. Katsaggelos, "Variational Bayesian super resolution," *IEEE Transactions on Image Processing*, vol. 20, no. 4, pp. 984–999, 2011.

[16] Y. Romano, M. Elad, and P. Milanfar, "The little engine that could: Regularization by denoising (red)," *SIAM J. Imaging Sciences*, vol. 10, pp. 1804–1844, 2017.

[17] D. Geman and C. Yang, "Nonlinear image recovery with half-quadratic regularization," 1995.

[18] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, pp. 1–122, 2011.

[19] T. Meinhardt, M. Möller, C. Hazirbas, and D. Cremers, "Learning proximal operators: Using denoising networks for regularizing inverse imaging problems," *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 1799–1808, 2017.

[20] S. A. Bigdeli, M. Zwicker, P. Favaro, and M. Jin, "Deep mean-shift priors for image restoration," in *Advances in Neural Information Processing Systems*, pp. 763–772, 2017.

[21] J. Yang, Z. Wang, Z. Lin, S. Cohen, and T. Huang, "Coupled dictionary training for image super-resolution," *IEEE Transactions on Image Processing*, vol. 21, pp. 3467–3478, Aug 2012.

[22] K. Zhang, X. Gao, D. Tao, and X. Li, "Single image super-resolution with multiscale similarity learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, pp. 1648–1659, Oct 2013.

[23] D. Trinh, M. Luong, F. Dibos, J. Rocchisani, C. Pham, and T. Q. Nguyen, "Novel example-based method for super-resolution and denoising of medical images," *IEEE Transactions on Image Processing*, vol. 23, pp. 1882–1895, April 2014.

[24] B. C. S. Hui Jung Lee, Dong-Yoon Choi, "Learning-based superresolution algorithm using quantized pattern and bimodal postprocessing for text images," *Journal of Electronic Imaging*, vol. 24, no. 6, pp. 1 – 8 – 8, 2015.

[25] Y. Zhang, Y. Zhang, J. Zhang, D. Xu, Y. Fu, Y. Wang, X. Ji, and Q. Dai, "Collaborative representation cascade for single-image super-resolution," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, pp. 845–860, May 2019.

[26] C. Dong, C. C. Loy, and X. Tang, "Accelerating the super-resolution convolutional neural network," in *Proceedings of European Conference on Computer Vision*, 2016.

[27] S. Anwar, S. Khan, and N. Barnes, "A deep journey into super-resolution: A survey," *ACM Comput. Surv.*, vol. 53, May 2020.

[28] J. Kim, J. Kwon Lee, and K. Mu Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1646–1654, 2016.

[29] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," *arXiv preprint arXiv:1609.04802*, 2016.

[30] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2472–2481, June 2018.

[31] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *ECCV*, 2018.

[32] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, C. C. Loy, Y. Qiao, and X. Tang, "Esrgan: Enhanced super-resolution generative adversarial networks," in *ECCV Workshops*, 2018.

[33] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1874–1883, 2016.

[34] M. I. Assaf Shocher, Nadav Cohen, ""zero-shot" super-resolution using deep internal learning," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[35] G. Riegler, S. Schulter, M. Rüther, and H. Bischof, "Conditioned regression models for non-blind single image super-resolution," in *IEEE International Conference on Computer Vision*, pp. 522–530, Dec 2015.

[36] Y. Yuan, S. Liu, J. Zhang, Y. Zhang, C. Dong, and L. Lin, "Unsupervised image super-resolution using cycle-in-cycle generative adversarial networks," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 814–81409, 2018.

[37] V. Vladimir and I. Rauf, "Learning using privileged information: Similarity control and knowledge transfer," *Journal of Machine Learning Research*, pp. 2023–2049, 2015.

[38] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," *arXiv preprint arXiv:1703.06211*, 2017.

[39] X. Zhu, H. Hu, S. Lin, and J. Dai, "Deformable convnets v2: More deformable, better results," *arXiv preprint arXiv:1811.11168*, 2018.

[40] C. Sonderby, J. Caballero, L. Theis, W. Shi, and F. Huszar, "Amortised MAP inference for image super-resolution," in *International Conference on Learning Representations*, 2017.

[41] A. Albert, *Regression and the Moore-Penrose pseudoinverse*. Mathematics in Science and Engineering, Burlington, MA: Elsevier, 1972.

[42] E. Pérez-Pellitero, M. S. M. Sajjadi, M. Hirsch, and B. Schölkopf, "Photorealistic video super resolution," in *Workshop and Challenge on Perceptual Image Restoration and Manipulation (PIRM) at the 15th European Conference on Computer Vision (ECCV)*, 2018.

[43] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *International Conference on Learning Representations (ICLR)*, May 2016.

[44] X. Jia, B. De Brabandere, T. Tuytelaars, and L. V. Gool, "Dynamic filter networks," in *Advances in Neural Information Processing Systems 29* (D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, eds.), pp. 667–675, Curran Associates, Inc., 2016.

[45] Y. Jo, S. W. Oh, J. Kang, and S. J. Kim, "Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3224–3232, 2018.

[46] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising," *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3142–3155, 2017.

[47] D. Li and Z. Wang, "Video super-resolution via motion compensation and deep residual learning," *IEEE Transactions on Computational Imaging*, 2017.

[48] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, vol. 2, pp. 416–423 vol.2, July 2001.

[49] Y. Matsui, K. Ito, Y. Aramaki, A. Fujimoto, T. Ogawa, T. Yamasaki, and K. Aizawa, "Sketch-based manga retrieval using manga109 dataset," *Multimedia Tools and Applications*, vol. 76, pp. 21811–21838, Oct 2017.

[50] E. Agustsson and R. Timofte, "Ntire 2017 challenge on single image super-resolution: Dataset and study," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1122–1131, July 2017.

[51] R. Timofte, E. Agustsson, L. V. Gool, M. Yang, L. Zhang, B. Lim, S. Son, H. Kim, S. Nah, K. M. Lee, X. Wang, Y. Tian, K. Yu, Y. Zhang, S. Wu, C. Dong, L. Lin, Y. Qiao, C. C. Loy, W. Bae, J. Yoo, Y. Han, J. C. Ye, J. Choi, M. Kim, Y. Fan, J. Yu, W. Han, D. Liu, H. Yu, Z. Wang, H. Shi, X. Wang, T. S. Huang, Y. Chen, K. Zhang, W. Zuo, Z. Tang, L. Luo, S. Li, M. Fu, L. Cao, W. Heng, G. Bui, T. Le, Y. Duan, D. Tao, R. Wang, X. Lin, J. Pang, J. Xu, Y. Zhao, X. Xu, J. Pan, D. Sun, Y. Zhang, X. Song, Y. Dai, X. Qin, X. Huynh, T. Guo, H. S. Mousavi, T. H. Vu, V. Monga, C. Cruz, K. Egiazarian, V. Katkovnik, R. Mehta, A. K. Jain, A. Agarwalla, C. V. S. Praveen, R. Zhou, H. Wen, C. Zhu, Z. Xia, Z. Wang, and Q. Guo, "Ntire 2017 challenge on single image super-resolution: Methods and results," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1110–1121, July 2017.

[52] G. Boracchi and A. Foi, "Modeling the performance of image restoration from motion blur," *IEEE Transactions on Image Processing*, vol. 21, no. 8, pp. 3502–3517, 2012.

[53] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Fast and accurate image super-resolution with deep laplacian pyramid networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[54] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai,

and S. Chintala, “Pytorch: An imperative style, high-performance deep learning library,” in *Advances in Neural Information Processing Systems 32* (H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, eds.), pp. 8024–8035, Curran Associates, Inc., 2019.

[55] D. Ulyanov, A. Vedaldi, and V. S. Lempitsky, “Deep image prior,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[56] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, “Image super-resolution using very deep residual channel attention networks,” *CoRR*, vol. abs/1807.02758, 2018.

[57] X. Zhou, M. Vega, F. Zhou, R. Molina, and A. K. Katsaggelos, “Fast Bayesian blind deconvolution with Huber super Gaussian priors,” *Digital Signal Processing*, vol. 60, pp. 122–133, 2017.

Chapter 4

Threat Detection in Passive Millimeter-Wave Images

4.1 Using Machine Learning to Detect and Localize Concealed Objects in Passive Millimeter-Wave Images

4.1.1 Publication details

Authors: Santiago López-Tapia, Rafael Molina, and Nicolás Pérez de la Blanca.
Title: Using Machine Learning to Detect and Localize Concealed Objects in Passive Millimeter-Wave Images.
Publication: Engineering Applications of Artificial Intelligence, vol. 67, 81-90, 2018.
Status: Published.
Quality indices:

- Impact Factor (JCR 2019): 4.201
- Rank: 33/136 (Q1) in Computer Science, Artificial Intelligence

4.1.2 Main Contributions

- First, we introduce a new preprocessing algorithm for PMMWIs to improve the performance of ML based models for threat detection.
- Second, we perform an extensive experimental comparison of the combination of different sets of features and ML models for PMMWI threat detection. As a result of this study, we propose a new ML-based approach for PMMWI threat detection.
- Finally, we introduce a new and comprehensive database of PMMWIs. This database contains 3309 images (463 without threat, 2144 with one threat and 702 with two threats) obtained from 33 people of different complexion and 12 different hidden objects simulating threats. To the best of our knowledge, this database is the largest and possesses the greatest variety of object types and sizes ever used for threat detection in PMMWIs.

Using Machine Learning to Detect and Localize Concealed Objects in Passive Millimeter-wave Images

Santiago López-Tapia, Rafael Molina, Nicolás Pérez de la Blanca
Dept. of Computer Science and Artificial Intelligence, *University of Granada*, Granada, Spain
Email: {sltapia, rms, nicolas}@decsai.ugr.es

Abstract

The detection and location of objects concealed under clothing is a very challenging task that has crucial applications in security. In this domain, passive millimeter-wave images (PMMWIs) can be used. However, the quality of the acquired images, and the unknown position, shape, and size of hidden objects render this task difficult. In this paper, we propose a machine learning-based solution to this detection/localization problem. Our method outperforms currently used approaches. The effect of non–stationary noise on different classification algorithms is analyzed and discussed, and a detailed experimental comparative study of classification techniques is presented using a new and comprehensive PMMWI database. The low computational testing cost of this solution allows for its use in real-time applications.

Index Terms

Threat detection, machine learning, passive millimeter-wave imaging.

I. Introduction

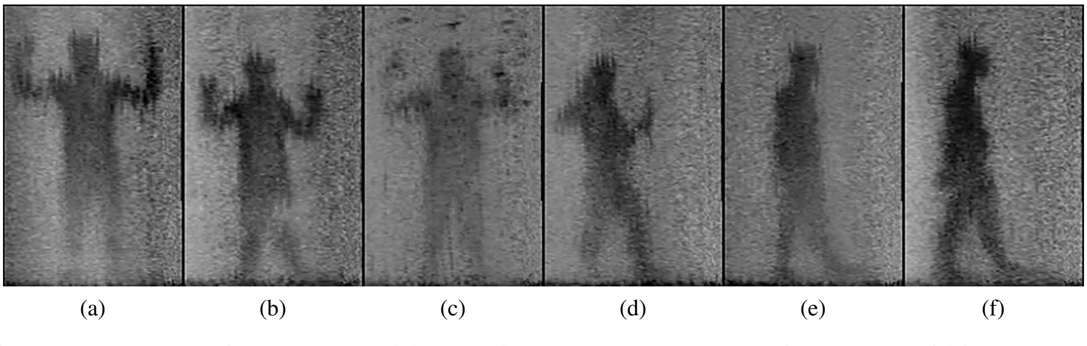

(a) (b) (c) (d) (e) (f)

Fig. 1: Examples of PMMWI. Hidden objects correspond to whiter areas within the body. Unfortunately, not all whiter areas correspond to hidden objects.

Millimeter and submillimeter waves are very high-frequency electromagnetic radiations with wavelengths in the ranges 30–300 GHz and 0.3–3 THz, respectively. Images in both ranges can be obtained by employing wave scanners. Depending on wavelength, two types of scanners can be distinguished: active, which direct waves from 350 GHz to 800 GHz to a subject, and collect and interpret the reflected energy; and passive scanners, which create images using background radiation and that emitted by human bodies or objects in the range 30–350 GHz.

Active scanners provide images with higher signal-to-noise ratios (SNR), but problems related to privacy intrusion have prevented their use in many applications. PMMWIs (see Figure 1)acquired using passive scanners are currently being used in anti-theft and threat detection systems ([1]) in places like airports and warehouses. Unfortunately, passive millimeter sensors (and consequently, their images) suffer from the following problems: low SNR, low resolution, and space-variant signal intensity. These factors degrade the performance of detection systems as they tend to produce a very large number of false positive detections and miss real threats. Detection systems based on passive scanners must deal with the unknown position, shape, size, and transmission properties of the hidden objects.

In this paper, we use artificial intelligence techniques to propose a new detection approach capable of dealing with the poor quality of PMMWIs. Our approach is based on feature extraction and classification algorithms, in contrast to current methods that only aim to enhance the SNR of images and apply simple detection algorithms.

Despite their enormous potential for security applications, the artificial intelligence community has so far shown little interest in PMMWIs. This is likely because of the absence of large databases of PMMWIs to work with. For this work, we took pictures of people of different complexions wearing 12 objects on 10 parts of the body: forearm, chest, stomach, thigh, ankle (front), waist (side), armpit (side), arm, ankle (lateral) thigh (lateral), and two images without any object. Images of people wearing two objects in different locations were also taken. Real hidden threats were simulated using objects of different sizes with millimeter wave responses similar to real threats. A cutter, gel, a clay bar, a simulated gun, sugar, frozen peas, cologne, a bag with metal pieces, flour, a bottle of water, and a hydrogen peroxide bottle were used. The dataset consisted of 463 pictures of people with no objects, 2,144 pictures containing one object, and 702 pictures containing two objects. More details are provided in Appendix. The passive scanner used in this work provided 125×195-pixel images. The sizes of the hidden objects of interest ranged from 35×39 to 10×10 pixels, corresponding to roughly $2752.39 cm^2$ to $201.64 cm^2$ object areas, respectively[1]. In the section detailing the experiments, a discussion of how the size, location, and composition of the threat influence the detection is provided.

This paper contains two major contributions. First, it provides a new real time solution to the detection/location of hidden objects in PMMWIs using a machine learning approach. Second, it introduces a new and comprehensive database of PMMWIs that encourages research on this challenging problem. In the remainder of the paper, the terms hidden object and threat are used interchangeably.

The rest of the paper is organized as follows: In section II, the current literature on PMMWI acquisition and threat detection is reviewed. Section III describes the image enhancement technique we propose to improve the quality of acquired images. Section IV describes the image patches extracted from the images in the dataset. Section V discusses meaningful features extracted

[1] The image database used in this paper can be downloaded at `http://decsai.ugr.es/pi/pmmwi/testdata.html` upon acceptance of the paper.

from each patch to solve the threat detection problem, and these features are compared in the experimental section. Once features have been extracted from each patch, a formal definition of our solution using a patch detection function and an image classification function is presented in section VI. These two functions are described in detail in section VII. An exhaustive experimental validation of the proposed classifiers is presented in section VIII. Conclusions are drawn in section IX. Appendix contains a complete description of the created dataset.

II. Related Work

The seminar work [2] described the phenomenology that defines the performance of PMMW imaging systems, explained the technological advances that have made these systems a reality, and presented some of the applications for which these sensors can be used (see also [3]). [4] examined trends in millimeter wave imaging technologies, focusing mainly on applications and technical parameter variations for security surveillance and non-destructive inspections.

As we have indicated, the use of very high frequencies in active sensors compromises the privacy of the people being scanned. Furthermore, due to their very narrow frequency range detection, passive terahertz sensors (0.3-0.35 THz) must be calibrated to detect specific materials making them blind to others. This favors the interest in and the wide range application of PMMW sensors. However, these sensors require the use of robust and efficient algorithms to detect concealed objects.

Image processing techniques have been utilized on PMMWIs. Denoising, deconvolution, and enhancement techniques have been applied to these images, (see [5], [6], [7], [8], [9], [10]). In this paper, we develop an image denoising technique tailored to the sensors used during the acquisition process. Note that the use of compressive sensing and super–resolution techniques on these images has also been explored (see [11], [12]).

Few studies have been devoted to the development of robust algorithms for the automatic detection of hidden objects when this type of image is used. In [13], an MMW imager which employed a 1D scanned focal-plane array operating at 0.35 THz and produced a real-time head-to-toe video output is utilized. K-means was used to segment MMWIs into three regions: background, body, and threats. However, the method may not detect a threat when its associated region is not connected to the body. To solve this problem, the authors used Active Shape Models (ASM) inside the body. However, this approach does not guarantee adequate body segmentation. In [14], Gaussian mixture models were used to characterize background, body, and threat regions and segment the images. Although the reported results were better than those in [13], this method also produces an unconnected body segmentation. In [15], the method was extended to detect and track metallic objects in a sequence of MMWIs.

In [16], for a passive terahertz imaging system, noise was first removed from the image using anisotropic diffusion. Following this, the boundaries of the concealed objects were detected. To model the distribution of the temperature inside the image a mixture of Gaussian densities was used. Curves were then evolved along the isocontours of the image to identify the concealed objects. In [17], the authors applied noise elimination and then image segmentation using local binary fitting (LBF). Two noise removal algorithms were used: non–local means (NLM) and iterative steering kernel regression (ISKR). Although this method's detection rate was around 90%, its computing time made it impractical for real-time applications. Furthermore, its performance significantly decreased when used on noisy or low quality images. In [18], a graph-cut algorithm was proposed to segment threats, but its evaluation was inaccurate. In [19], a method to detect and recognize threats in outdoor PMMWIs is proposed. The threat detection was carried out through global and local segmentation: at each level (global and local), a Gaussian mixture model whose parameters were initialized using vector quantization (for a different initialization approach see [20])and optimized through expectation maximization was used. This method was able to detect threats, but was only tested on a small set of images and 2 types of threats. In [21], the same segmentation process as in [19] was used, with the difference being the initialization of the Gaussian mixture model using k-means clustering. Shape features from the detected object were extracted and compared with the true features with reasonable accuracy. In [22], a mean-standard deviation-based segmentation technique was used and to detect and classify simple shape objects in MMWI images, and a probability density function for this classification was proposed. Furthermore, to improve the quality of the images the authors proposed the use of a neuronal network. In [23], a highly time efficient two-step algorithm, based on denoising and mathematical morphology, was proposed. On noisy or low contrast images, it achieved an acceptable detection rate but at the cost of a high false positive detection rate. In [24], singular value decomposition and discrete wavelet transform was used to ease the detection in MMWIs when thresholding was utilized. After this, a neuronal network was used for target identification. In [25], the authors used principal component analysis and a two-layer classification algorithm to distinguish between threats and normal objects.

Most of the reviewed techniques were oriented to obtain good quality images which were then used in tandem with simple methods to detect threats. In some cases, the methods were oriented to detect only particular threat types evaluated on small datasets. The approach we follow here is completely different, as it is entirely based on machine learning from the spatial statistical information of the gray level of the image and does not requires of geometric segmentation techniques. We base our approach on image patch classification aiming for the best accuracy/false positive trade-off. The approach is highly efficient; it can achieve a 100% true detection rate with a relatively small false positive detection rate. It can also be used to dissuade, for instance, robbery either by customers or workers. In this case a 100% true detection rate may not be achieved if *good* customers or workers are not to be disturbed. However, a trade-off between accuracy and false positive detection can always be achieved by tuning the model parameters. The proposed approach achieved very good detection scores for medium-low signal-to-noise ratio images. Preliminary results were presented at [26], here we provide a formal definition of the problem, new material on image pre-processing techniques, additional experiments, and a deeper discussion.

III. Image Enhancement

A natural question to consider is whether image filtering (smoothing, contrast enhancement, etc.) helps the detection process. Figures 2(c-d) show the result of applying mean, median, figs. 2(e-f), and bilateral, figs. 2(g-h), filters to observed passive images, figs. 2(a-b).

We observed that to assign to each pixel an estimation of the most frequent value in its neighborhood was a better smoothing criterion. That is, for each pixel i, let $\mathbf{y}_i^{\mathrm{B}}$ denote a block

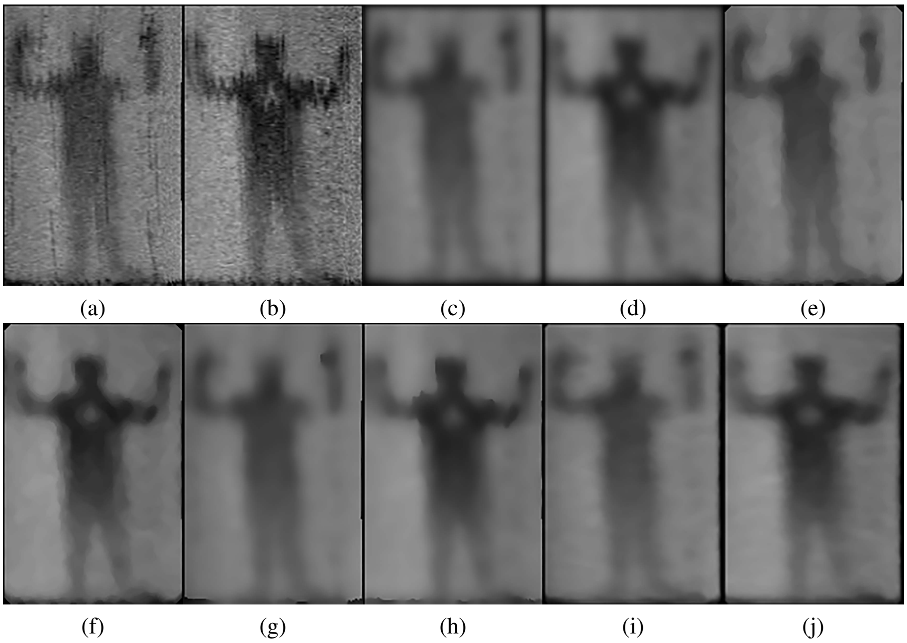


Fig. 2: Results of applying different image filtering techniques to (a) and (b). See text for details.

around y_i in the observed image $\mathbf{y}$ and $z_i(1), \ldots, z_i(K)$ denote K independent samples with replacement from $\mathbf{y}_i^{\mathrm{B}}$. We then define:

$$q_i = \frac{\sum_{k=1}^{K} z_i(k)}{K}, \tag{1}$$

assign $\mathbf{y} = \mathbf{q}$ and repeat the process L times. The final processed image is:

$$x_i = \text{median}(\mathbf{y}_i^{B}) \tag{2}$$

We found that $B = 5 \times 5$ and $L = 5$ produced satisfactorily processed images (see Figure 2(i-j)). Note that additional contrast improvement can be observed in these images. In the experimental section, we analyze how image processing improved the performance of the classifier.

IV. Patch extraction

As mentioned in the Introduction, passive scanners provide very low-resolution images (in our case, 125×195 pixel images). Because of the small size of the threat regions and the possibility of having several threats in the same image, threat detection was carried out using image patches. A patch is a rectangular image piece centered on a pixel.

Due to the variability of the threat sizes described above, we used patches at three scales on each pixel. These corresponded to patch sizes of 39×39, 19×19, and 9×9 pixels, respectively. Only pixels whose three patches were fully contained in the image (active pixels) were considered. We used $\mathbf{x}_j^P$ to denote the 39×39 patch centered on pixel j.

Let P_I be a mapping that selects from an image $\mathcal{I}_i$ in the image dataset $\mathcal{D} = \{\mathcal{I}_i, i = 1, \cdots, N_I\}$ a subset of 39×39 patches $\mathcal{P}_I^i$, specifically, $P_I(\mathcal{I}_i) = \mathcal{P}_I^i$. From each image, we extracted one 39×39 patch every 2×2 pixels, obtaining a total of 3476 patches per image. From the full dataset, we obtained $11,502,084$ patches (see Appendix). Patches that fully covered a threat were labeled "positive"$(+1)$; all others were labeled "negative." (-1). Patches that partially overlapped a threat were excluded from the training dataset. We acquired $392,494$ positive instances and $9,026,123$ negative instances. As the number of negatives was much higher (approximately 23 times), and considering that most of the negative patches were very similar, we used a subset of these for training. We kept one negative sample from every 2×2 image block, meaning that we retained $1/4$ of the negative patches we acquired. The final number of negative samples $(2,256,530)$ was approximately five times the number of positive ones. These patches constituted the patch dataset $\mathcal{T} = \cup_{i=1}^{N_I}\mathcal{P}_I^i$, associated with the image dataset $\mathcal{D}$. Feature vectors were then extracted from each patch.

V. Feature extraction

All classifiers to be used in this work needed to be fed with feature vectors. The pixel values in our 39x39 patches can be seen as a high-dimensional feature vector where local spatial information is hidden in the correlation between subsets of elements. In order to obtain an adequate vector of more useful characteristics for each patch, where: a) spatial information is explicitly represented; and b) individual characteristics are as uncorrelated as possible, two multi-resolution feature extraction techniques were utilized. Although the literature on feature extraction is extensive, (for example, [27]), we focused on features expected to have good threat detection capabilities.

We used Haar filters ([28]) and local binary patterns (LBP) codes ([29]) to create the feature vectors. The corresponding feature vectors are denoted by $\mathbf{x}_j^F$, with $F = LBP$ and $F = Haar$, respectively. It is well-known that both filter banks extract good neighboring features from an image; both features can be computed very efficiently.

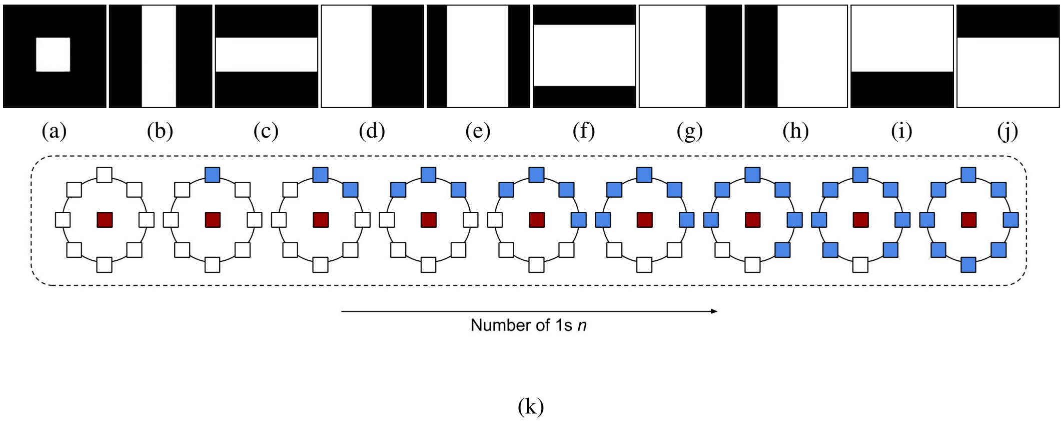


Fig. 3: Examples of Haar filters (a)-(j) and invariant to rotation LBP patterns for an 8 point neighborhood (k).

A. Haar filters

Haar filters ([28]) compute correlations of different binary patterns with an image region. The pixel image values in a white pattern are added and the difference between these values and the sum of pixel image values in the corresponding black pattern is calculated (see Figures 3(a-j)).

These patterns are expected to obtain very positive values at patches containing a threat as body regions have lower pixel values than threat regions. However, their responses will be close to zero on pure background or body regions. Notice, for instance, that the filter shown in Figure 3(a) shares the pattern of the hidden object in the row 1, column 2 image in Figure 9. Similarly, the filters in figs. 3(b-c) are similar to the hidden object areas in the row 1, column 1 and row 1, column 3 images in Figure 9, respectively. Although the gray levels of hidden objects and background are similar, hidden objects can be recognized as they are attached to the body, a darker region. We have used 115 filters on each patch resulting in a 3×115 feature vector per active pixel, which will be denoted by $\mathbf{x}_j^{Haar}$.

B. Local Binary Patterns

Local binary patterns (LBP) ([29]) capture image local structures by detecting gray level changes around each pixel. For every pixel in a patch, a binary vector is computed by checking if its value is greater than the value of a fixed number n of pixels located at a given radius r around it. This binary vector is coded as an integer number. The LBP feature vector of a patch is then calculated as the histogram of these numbers obtained for all pixels in the patch. Different radii can be used and the resulting feature vectors can be concatenated resulting in a larger number of features. Regions with an inside strong contrast due to the presence of boundaries between a hidden object and body or background are highlighted. In this experiment, we used the invariant-to-rotation LBP extension proposed in [30](see Figure 3(k)). For each of the three patches described in the previous section and centered on an active pixel, the histogram of all LBP configurations was obtained using a radius $r \in \{1, .., 4\}$, and several points $n = r * 8$ were built. The feature vector on each patch was obtained by concatenating the histograms of its three associated regions. In this case, there are 261 components and the feature vector will be denoted by $\mathbf{x}_j^{LBP}$.

In the following section we formally define the statistical learning framework we used to detect threats in PMMWIs. Its basic building blocks will be the image patches and the features extracted from them.

VI. MODEL

Given our image dataset $\mathcal{D}$ (see section IV), our goal was to learn a labeling function $f_{\mathcal{D}} : \mathcal{D} \rightarrow \{0, 1\}^s$, where s is the image size, with the lowest generalization error. This function assigns to each image pixel a binary value $\{0, 1\}$ indicating whether the pixel is part of a threat (1) or not (0). Thus, our problem was defined as an object localization problem using machine learning. We defined the function $f_{\mathcal{D}}$ in two steps. First, we detected potential threat regions (patches). Second, we determined which pixels were considered to belong to real threats. The full image dataset $\mathcal{D}$ was used to learn the function $f_{\mathcal{D}}$, and its generalization error was estimated using cross validation.

We used the patch dataset $\mathcal{T}$ defined in section IV to learn a patch detection function $f_{\mathcal{P}} : \mathcal{T} \rightarrow [0, 1]$. We assumed, without loss of generality, that the range of this function was the interval $[0, 1]$ (note that normalization can always be used). A binary classifier was built by splitting the range of $f_{\mathcal{P}}$ using a threshold to be learned.

Since the patch dataset $\mathcal{T}$ (see section IV) has five times more negative than positive patches, the set of negative patches used to learn an ensemble of classifiers was partitioned. Let $\mathcal{T}_P, \mathcal{T}_N$ define the subsets of positive and negative labeled patches respectively $\mathcal{T} = \mathcal{T}_P \cup \mathcal{T}_N$. Let us denote by n_P and n_N their cardinals, where $n_P << n_N$ and $n_C = n_N/n_P$. Let $\mathcal{T}_N = \cup_{i=1}^{n_C} \mathcal{T}_N^i$ be a random decomposition of the set $\mathcal{T}_N$ in n_C disjointed subsets. We solved n_C learning problems $f_{\mathcal{P}}^k$, $k = 1, \cdots, n_C$, associated to the training sets defined by $\{\mathcal{T}_N^k \cup \mathcal{T}_P\}$ respectively. We repeated the same procedure t times, obtaining an ensemble of $t \times n_C$ functions $f_{\mathcal{P}}^k$, which use the positive and subsets of the negative patches. Following this, the patch detection function was defined as $f_{\mathcal{P}}(\mathbf{x}^P) = \frac{1}{t \times n_C} \sum_k f_{\mathcal{P}}^k(\mathbf{x}^P)$.

We associated to the patch detection function $f_{\mathcal{P}}$ a detection threshold, $thr \in [0, 1]$. We set $f_{\mathcal{P}}(\mathbf{x}^P) = 0$ if $f_{\mathcal{P}}(\mathbf{x}^P) < thr$ and initially did not change $f_{\mathcal{P}}(\mathbf{x}^P)$ if $f_{\mathcal{P}}(\mathbf{x}^P) \geq thr$. To calibrate the threshold value, we used the ROC curve obtained from $f_{\mathcal{P}}$ when used as a binary classifier. Although $f_{\mathcal{P}}(\mathbf{x}^P) \geq thr$ declares $\mathbf{x}^P$ a potential threat patch (not all with the same detection function value), some of these potential threat patches are false positives that appear in the neighborhood of a true threat due to the contamination of the highest patch detection value to surrounding patches. We eliminated these contaminated patches using non-maximum suppression on overlapping patches. On each image, we rejected a detected patch $\mathbf{x}_j^P$ ($f_{\mathcal{P}}(\mathbf{x}_j^P) \geq thr$) if it had an intersection-over-union overlap larger than a learned threshold with a higher scoring patch $\mathbf{x}_l^P$. That is, if $f_{\mathcal{P}}(\mathbf{x}_l^P) > f_{\mathcal{P}}(\mathbf{x}_j^P) \geq thr$, we set $f_{\mathcal{P}}(\mathbf{x}_j^P) = 0$. This process is depicted in Figure 4.

Due to the strong no-stationary noise presence in the image, this approach can produce a high rate of false positive patches even when the thr value is properly tuned. Our solution to this problem is given in section VIII, where a calibration curve, to estimate the best rate between accuracy and false detection rates, is proposed.

After running the non-maximum suppression algorithm, we assigned to each image pixel, k, the maximum of all $f_{\mathcal{P}}(\mathbf{x}^P)$, for $k \in \mathbf{x}^P$. After this final processing, we acquired an image size vector of values in the interval [0,1]. This vector represents our estimation of the labelling function $f_{\mathcal{D}}$ on each image.

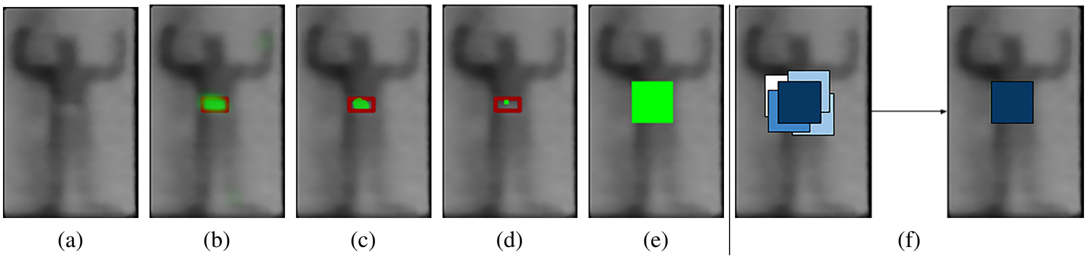


Fig. 4: Detection process: (a) observed image; (b) shows, using green intensities, the initial pixel probabilities (only at patch central pixels) assigned by the patch detection function together with the red bounding box surrounding the threat; (c) shows the active pixels after thresholding by thr; (d) shows the final estimation of the threat center; (e) shows the corresponding patch after non-maximum suppression on (c); (f) shows the details of the non-maximum suppression process.

VII. Learning the patch detection function $f_{\mathcal{P}}$ and the image classification function $f_{\mathcal{D}}$

We considered six patch detection functions $f_{\mathcal{P}}$. These functions represent different strategies to measure the presence of a threat in a patch: logistic regression (LR), quadratic logistic regression (QLR), support vector machine (SVM), random forest (RF), extreme random trees (ERT), and adaboost (ADA). LR and QLR were the linear approaches used as our baseline in the comparative study, and SVM was the kernel-based approach. These three methods search for the best function from a fixed set of possible functions. ADA, RF, and ERT build a classifier from an ensemble of simpler classifiers. RF and ERT use different subsets of the training dataset for each member of the ensemble, and ADA weights the training dataset with different sets of values. In all cases, the patch detection function range is normalized to $[0, 1]$.

For each classifier M$\in$ $\{$LR, QLR, SVM, RF, ERT$\}$, the corresponding image classification function $f_{\mathcal{D}}^{M}$ was estimated and the corresponding training and error estimations were calculated using five-fold cross-validation. This cross–validation partition was performed on the images and on every fold. Each fold contained approximately 600 images, all of which had the same proportion of images with no threat, one threat, and two threats. For adaboost, we used asymmetric boosting ([31]) instead of a committee of classifiers.

TABLE I: Hyperparameter grid for the classifiers; see text for details.

Classifier	Parameter	Range	Classifier	Parameter	Range
LR	$\mathrm{C_{LR}}$	$[0, 5]$	SVM	$\mathrm{C_{SVM}}$	$[10^{-1}, 10^{5}]$
QLR	$\mathrm{C_{QLR}}$	$[0, 5]$		γ_{RBF}	$[10^{-5}, 10]$
RF	NT	$[100, 300]$	ERF	NT	$[100, 300]$
	MNF	[8, 30]		MNF	[8, 30]
	MEL	$[1, 50]$		MEL	[1, 50]

Hyperparameter estimation was carried out using five-fold cross-validation on every training fold. To reduce the time required to estimate each the hyperparameters of each model, a smaller subset of patches was selected. A uniform sample of patches, using an additional factor reduction of 3, was used. This required selecting one location for every 6×6 image block. Finally, for each threat in an image, we added at least one patch that contained it to the training dataset. This guaranteed the inclusion of all the threats in the database. The hyperparameter ranges for each patch detection function are shown in Table I. They are the regularization strength parameters $\mathrm{C_{LR}}$, $\mathrm{C_{QLR}}$, and $\mathrm{C_{SVM}}$ for LR, QLR, and SVM, with RBF kernel with gamma parameter γ_{RBF}, respectively. The estimated hyperparameters of the tree-based models RF y ERT are the number of trees (NT), the maximum number of features (MNF) to consider for splitting, and minimum number of examples per leaf (MEL).

The LR the regularization parameter ($\mathrm{C_{LR}}$)for quadratic penalty (weight-decay) was determined using an adaptive search. QLR uses linear and quadratic functions on the features, hence, the LASSO penalty function was utilized to select the relevant features which were then used to learn an LR model with a quadratic penalty. For SVM, RF, and ERT, a grid search was used to estimate the hyperparameters, and feature vectors were normalized by mean and variance before training.

To remove the possible bias introduced by the reduced number of patches used to learn the hyperparameters, we selected the three best sets of hyperparameters using the area value under the ROC curve for each classifier. These three sets were then compared to the complete set of training patches and the one with the largest AUC value was selected.

VIII. Experimental Validation

For a given image, let us denote by $\mathcal{H}_{\mathcal{S}}(\mathbf{h})$ and $\mathcal{P}_{\mathcal{S}}(\mathbf{x}^{\mathcal{P}})$ the support regions (sets of pixels) associated with a hidden object $\mathbf{h}$ and patch $\mathbf{x}^P$, respectively. Given an existing real hidden threat $\mathbf{h}$ in an image, we considered in the experiments that a patch $\mathbf{x}^P$ was correctly classified as containing a hidden threat when $f_{\mathcal{P}}(\mathbf{x}^P) > thr$ and $\mathcal{H}_{\mathcal{S}}(\mathbf{h}) \cap \mathcal{P}_{\mathcal{S}}(\mathbf{x}^P) > 0.5 * size(\mathcal{H}_{\mathcal{S}}(\mathbf{h}))$. A hidden object in the image was said to be correctly detected when there was at least one patch satisfying both conditions. Finally, an image was correctly classified as positive (containing a hidden threat) when, at least, a hidden object was detected.

We analyze two scenarios: preprocessed images and raw images. Our preprocessing technique was described in section III, and Tables II and III contain a summary of the results for both scenarios. The AUC column represents the area under the ROC curve for the test images when classified as with or without a hidden object. The third column (TP) shows the true positive percentage of detected hidden objects computed by five-fold cross-validation; fold mean and standard deviation are included. The fourth column shows the average number of false positive (FP) patches per image and their standard deviations. Note the high FP deviation values, which are due to the non–uniform quality of the images. The fifth and sixth columns contain the threshold and overlap parameters used for non-maximum suppression.

TABLE II: For the preprocessed images using the preprocessing method presented in III, this table shows a summary of the performance (AUC) of the classifiers when Haar and LBP features were used. Threshold (Thr) and Overlapping values (Ovl) are also included. See text for details.

	Haar features (preprocessed)				
Class	AUC	TP$\times 10^2$	FP	Thr.	Ovl.
LR	0.51	84±1.8	10.5±10.7	0.6	0.5
QLR	0.57	91±1.1	8.7±9.1	0.7	0.5
SVM	0.52	92±3.0	6.5±6.6	0.75	0.3
RF	**0.75**	**94±0.9**	**4.0±3.8**	**0.7**	**0.3**
ERT	0.74	94±0.6	5.0±5.0	0.7	0.5
ADA	0.57	93±1.1	6.4±6.2	0.5	0.3

LBP features (preprocessed)				
AUC	TP$\times 10^2$	FP	Thr.	Ovl.
0.57	92±0.8	10.2±10.5	0.70	0.5
0.57	92±1.0	10.2±10.5	0.70	0.5
0.55	93±1.4	8.8±8.9	0.60	0.4
0.61	92±0.5	6.6±6.5	0.55	0.3
0.63	90±1.2	6.1±6.0	0.55	0.3
0.59	92±1.2	10.0±10.2	0.50	0.5

As Table II indicates, the best results are obtained by RF with Haar features extracted from the preprocessed images using the algorithm proposed in Section III. Note that we also ran experiments with mean, median, and bilateral smoothing algorithms but obtained lower scores. Almost all classifiers obtained similar TP values, RF and ERT were the best ones when used with Haar features.

TABLE III: For raw images, this table shows a summary of the performance (AUC) of the classifiers when Haar and LBP features were used. Threshold (Thr) and Overlapping values (Ovl) are also included. See text for details.

	Haar features (raw)					**LBP features (raw)**				
Class.	AUC	TP×10^2	FP	Thr.	Ovl.	AUC	TP×10^2	FP	Thr.	Ovl.
LR	0.51	82±1.5	10.2±10.4	0.6	0.5	0.54	88±1.0	10.4±10.8	0.68	0.5
QLR	0.55	90±1.0	8.2± 8.7	0.73	0.5	0.56	89±1.0	10.6±10.9	0.7	0.5
SVM	0.50	94±1.0	6.7±6.6	0.75	0.3	0.52	93±6.0	11.5±12.1	0.7	0.4
RF	0.72	94±1.0	4.0±3.7	0.7	0.3	0.60	86±2.0	6.1±6.1	0.58	0.3
ERT	0.73	94±1.0	4.0±3.8	0.68	0.3	0.59	84±1.0	6.0±6.0	0.57	0.3
ADA	0.63	94±0.6	5.8±5.7	0.50	0.3	0.55	90±1.0	10.3±10.5	0.50	0.5

Table III shows the scores for the raw image scenario. In this case, RF and ERT again obtained the best results for Haar and LBP features. The AUC values were lower, and the best value was obtained by the ERT classifier. However, in terms of TP and FP values, the average results were similar for both scenarios. That is, the proposed preprocessing method only slightly helped the detection process, which shows that our machine learning-based classification approach can deal with poor-quality images.

In Table II it can also be observed that TP values for Haar and LBP features are very similar. However, when considering FP values, Haar features clearly outperform LBP features. As we have indicated in the Introduction, the average FP per image is a key figure to minimize. When analyzed in percentage terms, all models performed reasonably well. This means that FP values were always below 10% when the threshold was fixed to obtain a 100% TP detection rate (0% FN). However, a 10% FP value means that a very large number of patches were incorrectly considered to contain a threat, which means that the model became useless. Consequently, a better compromise between TP and FP scores must be reached by selecting a higher thr_{M} value which, however, will reduce the TP percentage. Figure 5a shows the ROC curve for the best model combination, RF+Haar features on the preprocessed images. Figure 5b shows the true positive and true negative rate curves for image classification. Their intersection point is slightly above 68% and defines the accuracy of the system to classify new images when $\text{FP} = 1 - \text{TP}$, i.e., both false errors are equal. Although this score might be considered low, it is important to note that the high slope of both curves (positive rates, negatives rates) at the intersection point indicates that it is possible to improve the true positive rate with a small threshold increment, but only if the increase in FPs is affordable. Figure 6 shows the lack of contrast in some FN images and, therefore, the difficulty in detecting some hidden objects without increasing the FP rate.

The success of tree-based methods can be explained by the fact that both RF and ERT minimized mean square error using ensemble voting, which is an effective approach to reduce noise in images. Regarding LBP, the figures in both tables show a high influence of noise on the quality of this feature type. Although LBP features yielded better detection results than Haar features when used with parametric classifiers (LR, QLR, SVM), their FP scores were too high to make them competitive. These results clearly demonstrate the influence of noise on the behavior of classifiers

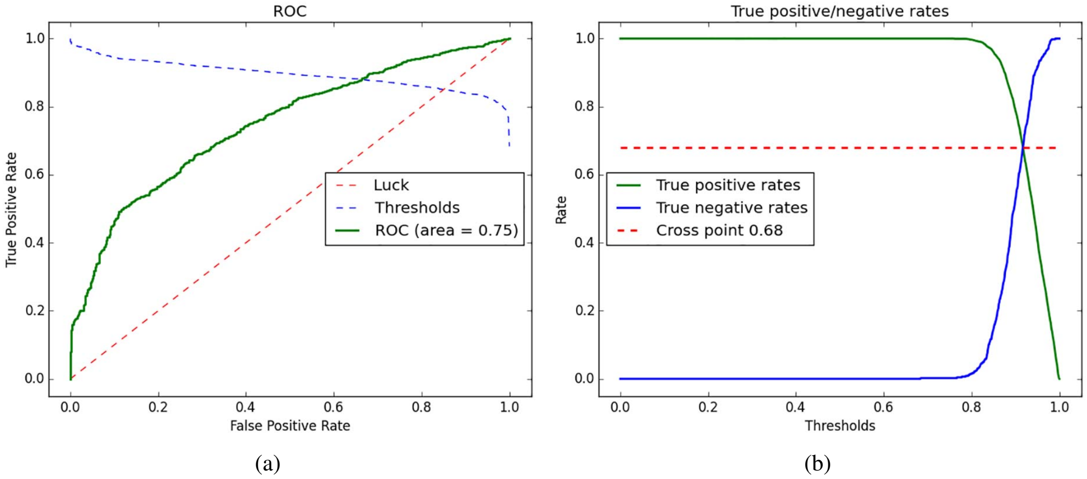


Fig. 5: (a) Shows the ROC when classifying new images. (b) The curves show the accuracy of the model on new images for a range of detection probability thresholds. The cross point is at the 68% of accuracy for both classes.

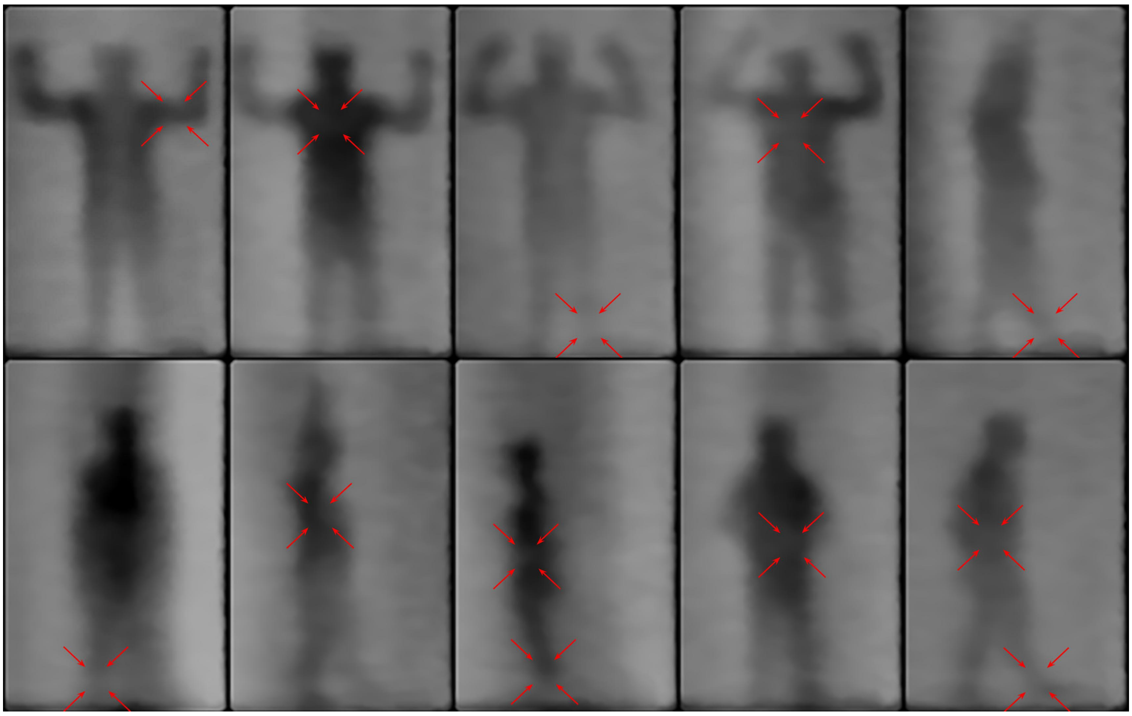
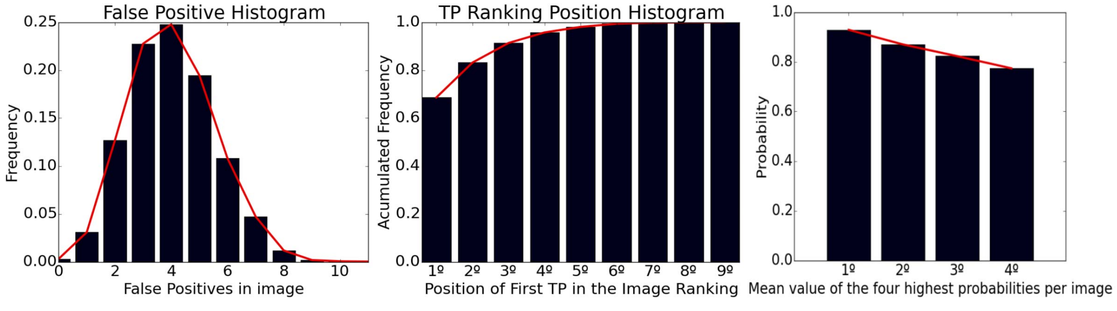


Fig. 6: Examples of hidden objects that RF could not detect when the classifier was calibrated for the same FP and FN rates. Red arrows indicate object locations.

and the importance of preprocessing when LBP features are used.

Figure 7 shows the analysis of the behavior of our winning combination, Haar+RF. In Figure 7a, the histogram of the average number of FPs per image is shown. We observed that the mode of the average of FPs per image was 4. Figure 7b shows the histogram of the first TP position on each image in the list of detected locations, ordered by decreasing probability. This histogram indicates that the clear majority of TPs were among the first two detected positive patches on



(a) (b) (c)

Fig. 7: Performance histograms of the best model, Haar+RF. See text for details.

each image, which validates the use of a high threshold for object detection. Figure 7c shows the histogram of the mean value of the four highest probability values per image, and the correct behavior of the classifier assigning the highest probabilities to regions that overlapped with hidden objects.

Tables IV(a-b) show how Haar+RF performed depending on threat location and type. Variations in TP and FP were highly dependent on the position of the object. When objects were in the body area (chest, stomach, waist, armpit) detection was simpler. However, when they were mainly surrounded by the background (arm, thigh, ankle), they were more difficult to detect. Table IV(b) shows a clear ranking of the objects depending on their PMMW responses. While the FP values were similar, the TP scores showed a large variation. Objects with higher densities (such as metallic objects) were easier to detect.

We also investigated whether models tailored to body regions improve the quality of the image classification function. For this, we used our best $f_{\mathcal{P}}$ function (RF+Haar) to fit four new models, one for each of the following body regions: arms, legs, chest, and ankles. Thus, only patches from one of these regions were used by each model. For test samples, the image localization of the patch determined the model to be used, and a partition of the image in the four body parts was prefixed. In terms of AUC, values there was no significant improvement. However, a 5% improvement on average FP was observed.

In the experiments, we used an Intel Xeon E5-2630 v3 at 2.40 GHz with 8 cores and 128 GB of RAM. Computation times during testing are shown in Table V. We observed that for both feature types, the methods performed approximately in real time. However, when using Haar features the classifiers performed faster. Finally, for RF and ERT, we used parallel implementations running on 16 threads.

The approach presented in this paper cannot be fairly compared with the methods described in section II. The methods reported in section II were tested on small sets of higher quality images, with a reduced number of threat locations, and with reduced variability on the threat characteristics. Nonetheless, we compared our best method, RF+Haar, against the methods described in [14] and [23] using datasets of different sizes. The best performing method was [14], which, for a dataset of 200 images, obtained a 13% TP and 1.92 mean FP. Our worst result was 47% TP and 1.5 mean FP, and was obtained with 30 raw images. We believe that our method performed better because it is more generally applicable and less adapted to the quality of the acquired images.

TABLE IV: The results of the best method (RF with Haar features). (a): Based on the threat location (front (F) and lateral (L)). (b) Based on object type (see Appendix). See text for details.

(a)

Loc.	AUC	TP×10^2	FP
Forearm	0.73	97±11.2	4.4±4.1
Chest	0.82	95±11.6	4.4±4.1
Stomach	0.86	96±11.6	4.3±4.1
Thigh	0.53	85±14.0	4.7±4.5
Ankle-F	0.65	88±8.3	4.4±4.2
Thigh-L	0.75	97±9.2	3.1±2.9
Waist-L	0.85	95±8.3	3.0±2.8
Armpit-L	0.86	99±12.0	3.0±2.7
Arm	0.55	85±14.4	5.3±4.9
Ankle-L	0.65	92±16.5	3.8±3.5

(b)

Threat	AUC	TP×10^2	FP
1	0.72	94±7.8	4.5±4.3
2	0.83	95±10.6	4.5±4.3
3	0.68	90±9.3	4.5±4.3
4	0.72	94±10.7	4.5±4.2
5	0.72	93±7.6	4.5±4.3
6	0.81	96±8.7	4.5±4.3
7	0.81	96±2.9	4.5±4.3
8	0.82	95±8.0	4.5±4.3
9	0.79	95±16.4	4.5±4.3
10	0.64	90±10.8	4.5±4.3
11	0.79	94±8.1	4.5±4.3
12	0.83	96±12.7	4.5±4.3

TABLE V: Testing Total (all the images) and per image (P.I.) times in seconds for each method.

	Haar		LBP	
Classifier	Total	P.I.	Total	P.I.
LR	302	0.09	4939	1.49
QLR	341	0.1	4939	1.49
SVM	1277	0.38	6907	1.87

	Haar		LBP	
Classifier	Total	P.I.	Total	P.I.
RF	**740**	**0.22**	**5497**	**1.66**
ERT	1202	0.36	5507	1.66
ADA	1965	0.59	7006	2.11

IX. Conclusions

This paper was devoted to the study of hidden object detection in PMMWIs. The main difficulty in this task arises from the low SNR and non-stationary noise that populates an image. Simple thresholding methods can be used but are most effective with high-quality images. In this study, a machine learning approach to the detection task was developed. This approach deals with the poor quality of passive images and outperforms state-of-the-art threat detection methods for PMMWIs.

Given the lack of publicly available PMMWI datasets, we created one that, to the best of our knowledge, is the largest, and possesses the greatest variety of object types and sizes ever used for this purpose problem[1].

Our proposed method is based on a committee of classifiers defined on two highly unbalanced classes of image patches, and performed well on all experiments. We compared different approaches to estimate image classification functions, and found using tree sets to be the most effective, reaching an average 94% TP score with a distribution of the number of false positives in the range of one to seven. The influence of the image quality and the extracted features were also analyzed. Our filtering method slightly helps the detection process; Haar filter banks, very well adapted to our task, performed very well for all classifiers.

The results indicate that large objects with reduced or zero emissions are simpler to detect. The easiest threat locations to detect were those where objects were exposed to the camera in larger areas. Threats in ankles, arms, and thighs were more difficult to detect.

Finally, a comparison between our detection model and other approaches in the literature indicated that our method is less reliant on the quality of the observed images. Furthermore, when a large image training set is available, our method performs very well, which makes a prediction of excellent performance for a wide range of millimeter-based detection systems realistic.

Appendix

A comprehensive dataset of PMMWIs was created. It consisted of 3,309 125×195 images of 33 people of different complexions. The hidden objects were in the range 35×39 to 10×10 pixels which, in our images, corresponded roughly to $2752.39cm^2$ and $201.64cm^2$, respectively. Smaller hidden objects were not considered relevant.

Threats were simulated by bags containing substances with different millimeter wave responses (see Figure 8). Note that objects of different sizes were used. We took pictures of each person wearing 12 objects on 10 parts of the body: forearm, chest, stomach, thigh, ankle (front), waist (side), armpit (side), arm, ankle (lateral) thigh (lateral), and 20 images without any object. Images of people wearing two objects in different locations were also taken. Some images were eliminated because of poor quality. The final dataset consisted of 463 pictures of people with no objects, 2,144 containing people wearing one object, and 702 containing two objects.

Figure 9 shows PMMWIs of subjects with simulated threats on different parts of the body and the corresponding color images taken by a camera located on the scanner. Although objects are visible and not hidden under the clothing, this was irrelevant for the PMMW sensors. Color images were taken at the same time. Threats are marked on the color images by the smallest bounding boxes containing them. To transfer object-bounding boxes from colored to PMMW images, a homography was estimated using a planar calibration pattern. Colored and PMMW images were then registered. The bounding boxes were used to assess the performance of the image classification functions.

Acknowledgements

Wavecamm (http://wavecamm.com) provided the PMMWIs used in the paper This work was supported by the Spanish MINECO under project TIN2013-43880-R,FEDER funds, and by Project TIC-1692 (Junta de Andalucía). Santiago López-Tapia received financial support through the Spanish FPU program.

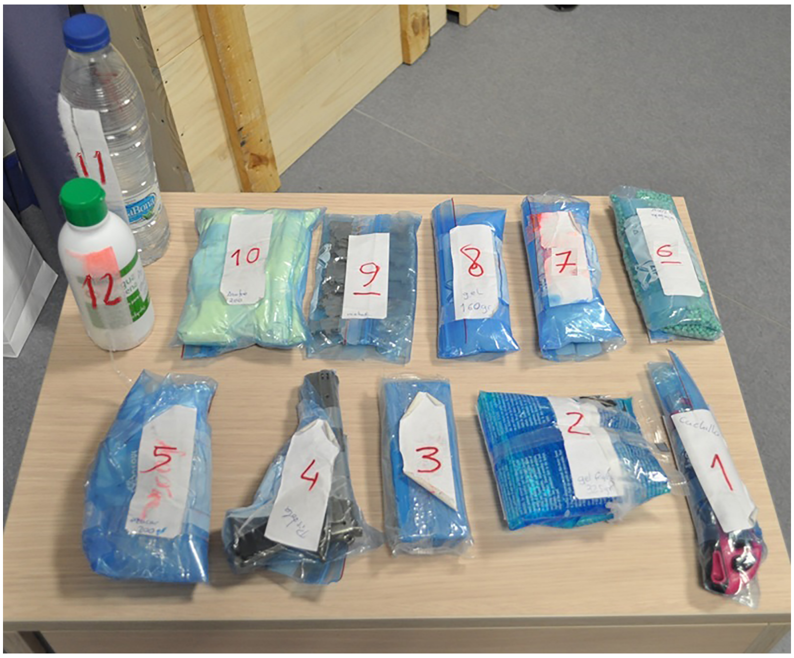


Fig. 8: Simulated threats in the dataset: a cutter (1), 325g of gel (2), a 200g clay bar (3), a simulated gun (4), 200g of sugar (5), 200g of frozen peas (6), 150ml of cologne (7), 160g of gel (8) , a bag with metal pieces (9), 200g of flour (10), a 50cl water bottle (11), and a 250ml hydrogen peroxide bottle (12).

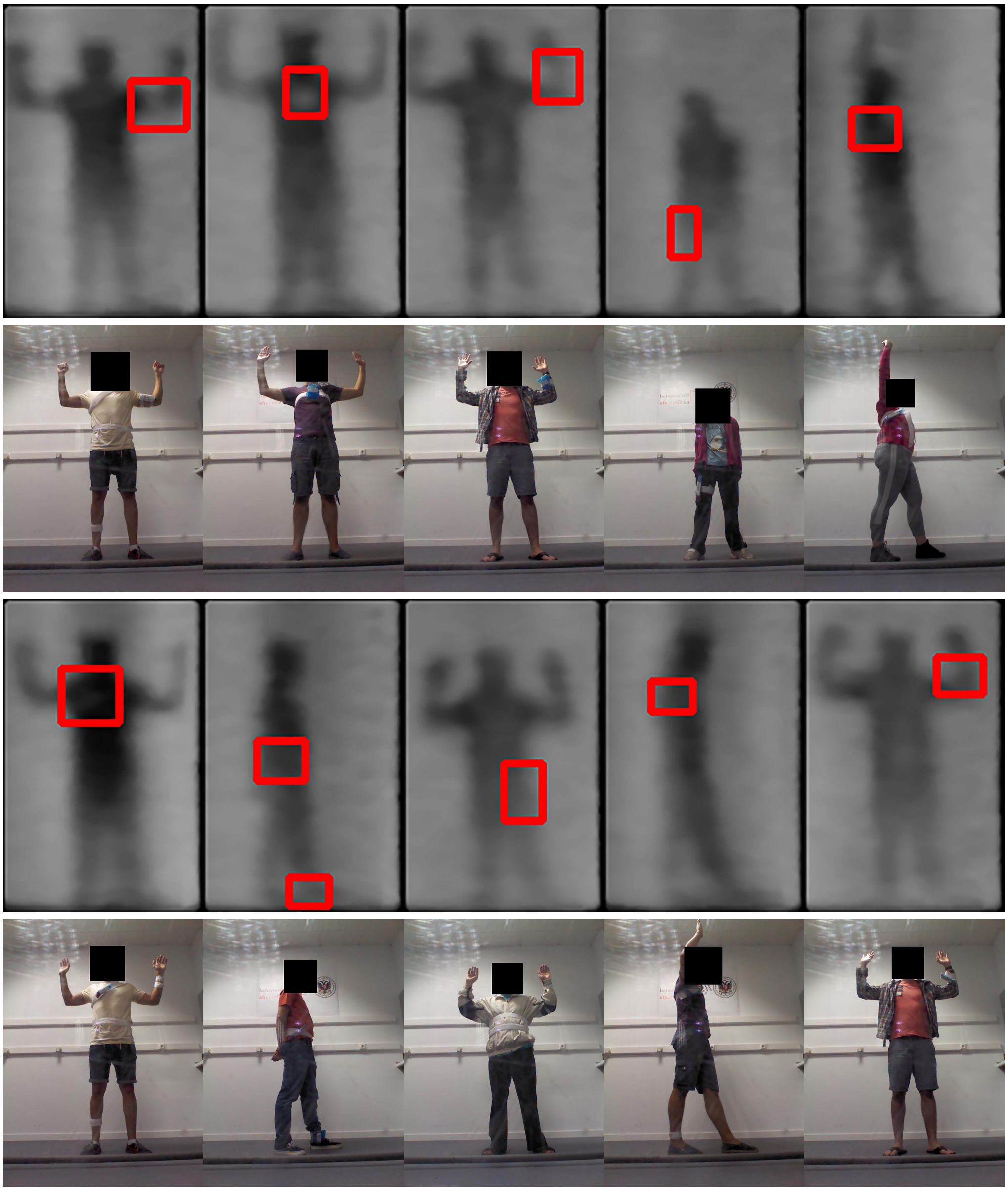


Fig. 9: First row: Examples of PMMWIs. Red boxes indicate the locations of hidden objects. Second row: Corresponding visual images of the PMMWIs' examples. (Best viewed on a high-resolution color screen).

References

[1] N. E. Alexander, C. Callejero Andrés, and R. Gonzalo, "Multispectral mm-wave imaging: materials and images," in *SPIE*, vol. 6948, pp. 694803–694812, 2008.

[2] L. Yujiri, M. Shoucri, and P. Moffa, "Passive millimeter wave imaging," *IEEE Microwave Magazine*, vol. 4, no. 3, pp. 39–50, 2003.

[3] L. Yujiri, "Passive millimeter wave imaging," in *IEEE MTT-S International Microwave Symposium Digest*, pp. 98–101, June 2006.

[4] S. Oka, H. Togo, N. Kukutsu, and T. Nagatsuma, "Latest Trends in Millimeter-Wave Imaging Technology," *Progress In Electromagnetics Research Letters*, vol. 1, pp. 197–204, 2008.

[5] B. Han, J. Xiong, L. Li, J. Yang, and Z. Wang, "Research on millimeter-wave image denoising method based on contourlet and compressed sensing," in *2nd International Conference on Signal Processing Systems*, vol. 2, pp. 471–475, July 2010.

[6] J. Mateos, A. López, M. Vega, R. Molina, and A. Katsaggelos, "Multiframe blind deconvolution of passive millimeter wave images using variational dirichlet blur kernel estimation," in *IEEE International Conference on Image Processing*, pp. 2678–2682, Sept 2016.

[7] T. Liu, Z. Chen, S. Liu, Z. Zhang, and J. Shu, "Blind image restoration with sparse priori regularization for passive millimeter-wave images," *Journal of Visual Communication and Image Representation*, vol. 40, pp. 58–66, 2016.

[8] H. Fang, Y. Shi, D. Pan, and G. Zhou, "Iteratively reweighted blind deconvolution for passive millimeter-wave images," *Signal Processing*, vol. 138, pp. 182–194, 2017.

[9] J. Yang, J. Wang, and L. Li, "A new algorithm for passive millimeter-wave image enhancement," in *2nd International Conference on Signal Processing Systems*, vol. 3, pp. 507–511, July 2010.

[10] W. Yu, X. Chen, S. Dong, and W. Shao, "Study on image enhancement algorithm applied to passive millimeter-wave imaging based on wavelet transformation," in *International Conference on Electrical and Control Engineering*, pp. 856–859, Sept 2011.

[11] S. D. Babacan, M. Luessi, L. Spinoulas, A. K. Katsaggelos, N. Gopalsami, T. Elmer, R. Ahern, S. Liao, and A. Raptis, "Compressive passive millimeter-wave imaging," in *18th IEEE International Conference on Image Processing*, pp. 2705–2708, Sept 2011.

[12] W. Saafin, S. Villena, M. Vega, R. Molina, and A. Katsaggelos, "Compressive sensing super resolution from multiple observations with application to passive millimeter wave images," *Digital Signal Processing*, vol. 50, pp. 180–190, 2016.

[13] C. Haworth, B. Gonzalez, M. Tomsin, R. Appleby, P. Coward, A. Harvey, K. Lebart, Y. Petillot, and E. Trucco, "Image analysis for object detection in millimetre-wave images," in *Passive Millimetre-wave and Terahertz Imaging and Technology*, vol. 5619, pp. 117–128, December 2004.

[14] C. Haworth, Y. Petillot, and E. Trucco, "Image processing techniques for metallic object detection with millimetre-wave images," *Pattern Recognition Letters*, vol. 27, no. 15, pp. 1843–1851, 2006.

[15] C. D. Haworth, Y. De Saint-Pern, D. Clark, E. Trucco, and Y. R. Petillot, "Detection and tracking of multiple metallic objects in millimetre-wave images," *International Journal of Computer Vision*, vol. 71, no. 2, pp. 183–196, 2007.

[16] X. Shen, C. R. Dietlein, E. Grossman, Z. Popovic, and F. G. Meyer, "Detection and segmentation of concealed objects in terahertz images," *IEEE Transactions on Image Processing*, vol. 17, pp. 2465–2475, Dec 2008.

[17] O. Martínez, L. Ferraz, X. Binefa, I. Gómez, and C. Dorronsoro, "Concealed object detection and segmentation over millimetric waves images," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*, pp. 31–37, June 2010.

[18] M. Sarkis and L. Mani, "Reconstruction of passive millimeter-wave images with graph cuts," in *19th IEEE International Conference on Image Processing*, pp. 2053–2056, Sept 2012.

[19] S. Yeom, D.-S. Lee, Y. Jang, M.-K. Lee, and S.-W. Jung, "Real-time concealed-object detection and recognition with passive millimeter wave imaging," *Optics Express*, vol. 20, pp. 9371–9381, Apr 2012.

[20] W. Yu, X. Chen, and L. Wu, "Segmentation of concealed objects in passive millimeter-wave images based on the gaussian mixture model," *Journal of Infrared, Millimeter, and Terahertz Waves*, vol. 36, no. 4, pp. 400–421, 2015.

[21] S. Yeom, D.-S. Lee, and J.-Y. Son, "Shape feature analysis of concealed objects with passive millimeter wave imaging," *Progress In Electromagnetics Research Letters*, vol. 57, pp. 131–137, 2015.

[22] S. Agarwal, A. S. Bisht, D. Singh, and N. P. Pathak, "A novel neural network based image reconstruction model with scale and rotation invariance for target identification and classification for active millimetre wave imaging," *Journal of Infrared, Millimeter, and Terahertz Waves*, vol. 35, no. 12, pp. 1045–1067, 2014.

[23] I. Gómez, N. Pérez de la Blanca, R. Molina, and A. Katsaggelos, "Fast millimetre wave threat detection algorithm," in *23rd European Signal Processing Conference*, pp. 599–603, Aug 2015.

[24] B. Kumar, P. Sharma, R. Upadhyay, D. Singh, and K. P. Singh, "Optimization of image processing techniques to detect and reconstruct the image of concealed blade for mmw imaging system," in *IEEE International Geoscience and Remote Sensing Symposium*, pp. 76–79, July 2016.

[25] H. Mohammadzade, B. Ghojogh, S. Faezi, and M. Shabany, "Critical object recognition in millimeter-wave images with robustness to rotation and scale," *Journal of the Optical Society of America A*, vol. 34, pp. 846–855, Jun 2017.

[26] S. López-Tapia, R. Molina, and N. Pérez de la Blanca, "Detection and localization of objects in passive millimeter wave images," in *24th European Signal Processing Conference*, pp. 2101–2105, Aug 2016.

[27] M. Nixon and A. S. Aguado, *Feature Extraction & Image Processing*. Academic Press, second ed., 2012.

[28] C. Papageorgiou, M. Oren, and T. Poggio, "A general framework for object detection," in *6th International Conference on Computer Vision*, pp. 555–562, Jan 1998.

[29] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern Recognition*, vol. 29, pp. 51–59, Jan. 1996.

[30] T. Ahonen, J. Matas, C. He, and M. Pietikäinen, "Rotation invariant image description with local binary pattern histogram fourier features," in *16th Scandinavian Conference on Image Analysis*, pp. 61–70, June 2009.

[31] P. Viola and M. Jones, "Fast and robust classification using asymmetric adaboost and a detector cascade," in *Advances in Neural Information Processing Systems*, vol. 14, pp. 1311–1318, 2001.

4.2 Deep CNNs for Object Detection using Passive Millimeter Sensors

4.2.1 Publication details

Authors: Santiago López-Tapia, Rafael Molina, and Nicolás Pérez de la Blanca.
Title: Deep CNNs for Object Detection using Passive Millimeter Sensors.
Publication: IEEE Transactions on Circuits and Systems for Video Technology, vol. 29, no. 9, 2580-2589, September 2019.
Status: Published.
Quality indices:

- Impact Factor (JCR 2019): 4.133
- Rank: 50/266 (Q1) in Engineering, Electrical and Electronic Intelligence

4.2.2 Main Contributions

- We perform a comprehensive experimental study using the database introduced in Section 4.1 and analyze the application of DL based models to PMMWI threat detection using two approaches: Detection using a Local Approach (DLA) and Segmentation using a Global Approach (SGA). As a result of this study, we propose a new DL-based SGA for fast an accurate threat detection on PMMWIs.
- We perform experiments and show that DL-based models can deal with the non-stationary noise in PMMWIs and, in contrast to other ML models, do not benefit from preprocessing the images.

Deep CNNs for Object Detection using Passive Millimeter Sensors

Santiago López-Tapia, Rafael Molina, Nicolás Pérez de la Blanca
Dept. of Computer Science and Artificial Intelligence, *University of Granada*, Granada, Spain
Email: {sltapia, rms, nicolas}@decsai.ugr.es

Abstract

Passive Millimeter Wave Images (PMMWIs) can be used to detect and localize objects concealed under clothing. Unfortunately, the quality of the acquired images and the unknown position, shape, and size of the hidden objects render these tasks challenging. In this paper we discuss a deep learning approach to this detection/localization problem. The effect of the non stationary acquisition noise on different architectures is analyzed and discussed. A comparison with shallow architectures is also presented. The achieved detection accuracy defines a new state of the art in object detection on PMMWIs. The low computational training and testing costs of the solution allow its use in real time applications.

Index Terms

Millimeter wave imaging, deep learning, object detection, security, classification.

I. Introduction

Millimeter images are obtained by sensors capturing electromagnetic radiation in the 0.001-0.01 m wavelength range. Two types of sensors can be distinguished: active, which direct the waves to the subject and then collect and interpret the reflected energy; and passive, which create images using the radiation emitted by objects, in general, and human bodies in particular.

In contrast to alternatives like backscatter X-ray, passive millimeter systems are safe, see the interesting discussion in [1]. Furthermore, they fully respect the privacy of their users. Unfortunately, millimeter sensors, and consequently their images, suffer from, among others, the following problems: low signal to noise ratio, low resolution, which can be increased by increasing the sampling rate but at the cost of decreasing the signal to noise ratio, and in-homogeneous signal intensity.

PMMWIs (see Fig. 1) are currently being utilized as theft and threat detection systems [2] in places like airports and warehouses. Since April 2009, many PMMWI systems have been installed in large USA and European airports. These detection systems have to deal with the unknown position, shape, size and transmission properties of the hidden objects. Unfortunately, their false

Research funded by the Spanish Ministry of Economy and Competitiveness (MINECO) under projects TIN2013-43880-R and DPI2016-77869-C2-2-R, FEDER Funds, and project TIC-1692 (Junta de Andalucía). Santiago López-Tapia received financial support through the Spanish FPU program. We would like to acknowledge the use of the real PMMWIs provided by Wavecamm (http://wavecamm.com).

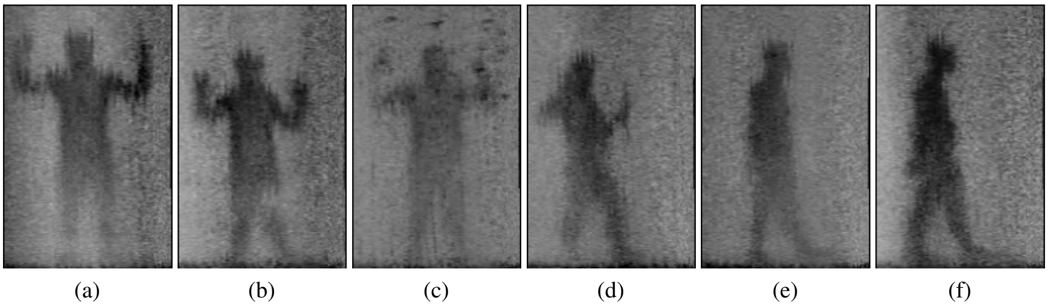


Fig. 1: PMMWI examples. Hidden objects correspond to whiter areas within the body. Unfortunately, not all whiter areas correspond to hidden objects.

positive ratios are very high. Systems based on PMMWIs should be able to detect concealed objects, incur in a very low number of false positive detections, and work in real time. It is worth mentioning here that the scientific community has, so far, shown little interest on the challenging tasks involved in threat detection using PMMWIs. This is likely because of the absence of large databases of PMMWIs to work with.

Sensor modelling and image processing techniques have been used on PMMWIs. In [3], the main concepts related to millimeter images are introduced, see also [4]. In [5], trends in millimeter wave imaging technologies are examined, focusing mainly on applications and technical parameter variations for security surveillance and nondestructive inspections. Image processing techniques have also been utilized on PMMWIs. The use of compressive sensing and superresolution techniques on these images is explored in [6], [7]. Denoising, deconvolution, and enhancement techniques have also been applied to this kind of images, see for instance [8]–[11]. In this paper, we develop an image classification approach to detect and localize objects in a context where the image noise is non stationary and very high.

II. Related Works

K-means is used in [12] to segment PMMWIs into three regions: background, body and threats. Unfortunately, the method detects unconnected areas. To solve this problem, the authors use Active Shape Models (ASM) inside the body. However, this approach does not guarantee an adequate segmentation. In [13], Gaussian mixture models are used to characterize background, body, and threat regions and segment the image. Although the reported results are better than those in [12], this method also produces an unconnected body segmentation. In [14], the authors apply noise elimination and then image segmentation using Local Binary Fitting (LBF). The authors use two algorithms for noise removal: Non Local Means (NLM) and Iterative Steering Kernel Regression (ISKR). Although its detection rate is around 90%, its computing time makes it impractical for real-time applications. Furthermore, its performance decreases significantly when used on noisy or low quality images. In [15], a fast two-step algorithm, based on denoising and mathematical morphology, was proposed. On noisy or low contrast images it achieves an acceptable detection rate but at the cost of a high false positive detection rate. In [16], a method to detect and recognize threats in outdoor PMMWIs is proposed. The threat detection is carried out through global and local segmentation: at each level (global and local), a Gaussian mixture model, whose parameters are initialized using vector quantization, is used and optimized through expectation maximization. For a different initialization approach see [17]. Finally, a Bayesian decision rule decides which cluster each pixel belongs to. The recognition process of the threat type consists of upscaling, principal component analysis (PCA), size normalization, extraction of a geometric-based feature vector composed of shape descriptors, and a decision rule, where the class is decided by minimum Euclidean distance between normalized feature vectors. The method was able to detect threats, but it was tested exclusively on a small set of images and with only two types of threats. In [18], the same segmentation process as in [16] was used, the difference being the initialization of the Gaussian mixture model using k-means clustering. Shape features from the detected object (area, perimeter, major and minor axes of the basic rectangle, rectangularity, compactness, and eccentricity) are extracted and compared to the true features. The method shows good accuracy. Notice that all the above approaches aim at segmenting the concealed objects, furthermore they were evaluated on a small set of images. In [19], [20], we introduced the UGR-PM^{2}WI database described in appendix B, a new and comprehensive dataset of PMMWIs. A comparative study of the performance of different shallow classifiers on this database was presented. Although the proposed classifiers, and in particular Random Forest, outperform previous approaches, it was shown that the image noise (see Fig. 1) has a strong influence on this kind of classifiers, making very difficult to do better than a 68% average detection score.

All the above approaches can be grouped according to the spatial context used in the feature extraction process: a) local spatial context, used in object detection/localization tasks and b) full image context, mainly used in image segmentation tasks. In what follows, these approaches will be denoted Detection using a Local Approach (DLA) and Segmentation using a Global Approach (SGA), respectively.

In the Deep Learning (DL) methodology we are going to propose, deep SGA and DLA approaches will be analyzed and compared. Both approaches will be cast under the same framework, the one provided by Convolutional Neural Network (CNN) architectures used for supervised classification problems. The difference between both approaches will be characterized by the class of functions computed by the deep CNN architectures.

The rest of the paper is organized as follows. Section III provides a short introduction to the used CNN architectures. In section IV, we describe how the threat detection problem can be tackled using deep classifiers. The detection and training procedures are also discussed in this section. In section V, we describe the experimental set-up and show the obtained results. Also, a comparison with the best shallow architectures is also presented here. In section VI, we analyze the performance of the different learning architectures, identifying the most relevant factors providing high detection score. Conclusions are drawn in section VII. Appendix A contains a description of the denoising technique applied to the observed image, the importance of its use is described in section VI. Appendix B contains a complete description of the dataset used in the experiments. [1].

III. Deep architectures

Nowadays, Feature Learning Models (FLM) compete with advantage against the standard classification methodology where a feature extraction process is carried out to obtain the information

[1] The dataset is available at http://decsai.ugr.es/pi/pmmwi/

to feed a classifier. FLMs learn the best set of features by minimizing the training error. Deep CNN models have shown the importance of using deep architectures as a mechanism to disentangle independent features coded in the data through a very complex non-linear mapping of the sample data. CNNs are currently the most successful DL models for high level image related tasks like classification, segmentation, detection, parsing, etc. CNNs define the state of the art performance in many of them [21]–[27].

Although our threat detection problem could be naively approached by the direct application of well-known image processing techniques, the problem is much more challenging. As shown in this paper, the presence of non-stationary noise in the images renders extremely difficult the detection task. The use of function classes as complex as those defined by DL architectures is required.

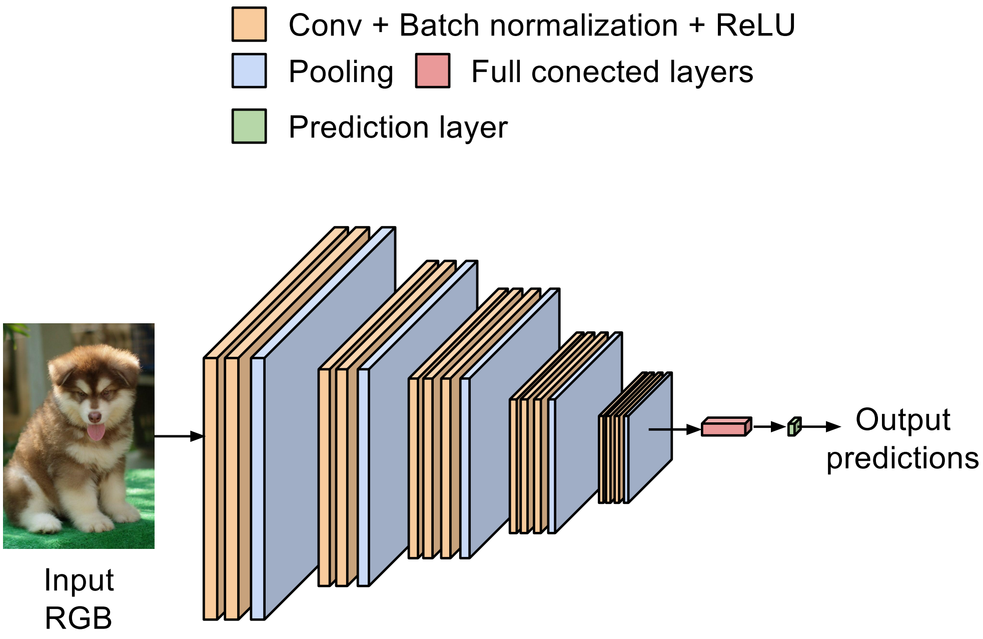


Fig. 2: Example of a CNN architecture.

A. Used architectures

The LeNeT architecture introduced by Lecun [28] can be considered as the base model from which more sophisticated CNN architectures have been designed. A CNN model is a feedforward neural net defined as a deep composition of functions or layers, where each layer computes a transformation from an accepted list of transformations: convolution, pooling, non-linear activation, full connection, regularization and normalization. The design of a specific model consists of two phases. Firstly, we have to fix the architecture, that is, to define the number, the order, and the characteristics of the layers. Secondly, we must learn the free parameters associated to the convolution and fully connected layers. In the last layer, a final vector is obtained which is used to feed a classifier. We use a softmax classifier, as it provides a posterior probability estimation for each class. A simple CNN architecture is shown in Fig. 2.

For our problem, we have analyzed five different CNN architectures. Three of them, with increasing level of complexity, for the DLA approach and two for the SGA approach. The learning protocols for DLA and SGA are different. DLA models are fed with image patches while SGA ones utilize full images.

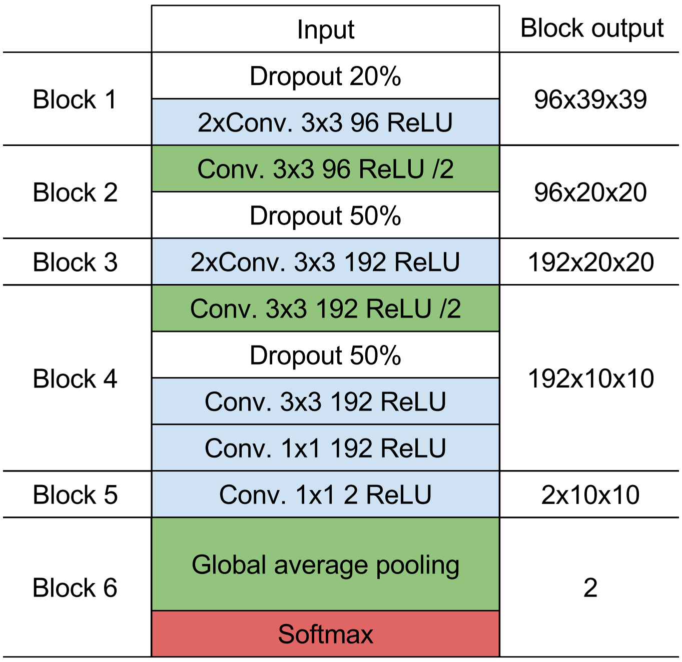
Let us start by describing the three architectures which utilize image patches. The simplest model is an adaptation of the LeNeT [29] architecture. We use the same type and number of layers as the original, 2 convolutional layers, 2 pooling layers and 2 fully connected layers, but the number of filters in the first and second convolutional layers has been changed from $\{20{,}50\}$ to $\{32{,}64\}$ filters, respectively. Also the original non-linear sigmoidal function is changed to a ReLU [30] activation function, $f(x) = max(0, x)$. Batch normalization [31] has also been introduced before each ReLU transformation to decorrelate the outputs of the convolutional layers and, finally, regularization with dropout [32] has been added to the last fully connected layers. All these changes adjust the original architecture [29] to our threat detection problem and produce better classification figures. Fig. 3 shows a summary of the this low deep CNN architecture (LD-CNN). This model has 12 layers and 1,621,598 free parameters. The central column of the figure describes the order and functionality of each layer (top-down).

	Input	Block output
Block 1	Conv. 5x5 32 ReLU	32x39x39
Block 2	Max-pooling 2x2	64x20x20
	Conv. 5x5 64 ReLU	
Block 3	Max-pooling 2x2	64x10x10
	Conv. 5x5 64 ReLU	
	Dropout 50%	
Block 4	Full connected 500 ReLU	500
	Dropout 50%	
Block 5	Full connected 2 Softmax	2

Fig. 3: Description of the LD-CNN architecture. The total number of parameters is 1,621,598. By Dropout x% we mean that a x% of the pre-activation units are assigned to zero.

A very recent architecture named All-CNN-C [33] implements pooling and fully connected layers with convolutional layers. This architecture allows us to train models similar to LD-CNNs but with larger number of layers. The main advantage of this architecture is that it only depends on convolution operations which are very well adapted to GPU hardware. A max-pooling layer can be seen as a convolution with kernel size equal to the kernel size of the pooling followed by a non-linear transformation and a downsampling operation. This means that each max-pooling layer can be replaced by three convolution layers. In addition, as a bonus, the pooling parameters can be included as parameters to optimize. A fully connected layer can also be replaced by convolutional layers with a 1×1 kernel and the same number of kernels as output units has the fully connected layer. Therefore, all fully connected layers are replaced by convolution layers.

We adopt this All-CNN-C [33] architecture to design our medium deep CNN model (MD-CNN) shown in Fig. 4. Now we have an architecture with almost twice the number of layers the LD-CNN has, but in terms of effective transformations it is only slightly deeper than LD-CNN. Two convolutional layers implement transformations equivalent to max-pooling and the layers in block 5 implement transformations equivalent to a fully connected layer. The total number of free parameters is 1,371,458. Notice that, although MD-CNN is more complex and powerful than LD-CNN, the number of parameters to be estimated is lower.



	Input	Block output
Block 1	Dropout 20%	96x39x39
	2xConv. 3x3 96 ReLU	
Block 2	Conv. 3x3 96 ReLU /2	96x20x20
	Dropout 50%	
Block 3	2xConv. 3x3 192 ReLU	192x20x20
Block 4	Conv. 3x3 192 ReLU /2	192x10x10
	Dropout 50%	
	Conv. 3x3 192 ReLU	
	Conv. 1x1 192 ReLU	
Block 5	Conv. 1x1 2 ReLU	2x10x10
Block 6	Global average pooling	2
	Softmax	

Fig. 4: Description of the MD-CNN architecture. The total number of parameters is 1,371,458. See dropout x% meaning in Fig. 3. In this model a convolution layer with stride 2 is equivalent to a pooling and sub-sampling stage.

Training very deep networks is usually difficult since the optimization procedure frequently gets trapped in very poor local minima. The Residual Network approach [34] overcomes this problem by fitting on each new layer a residual function f of the form $out = f(input) + input$ instead of using the standard function defined by $out = f(input)$. Fig. 5 shows a standard residual unit as described in [34]. In order to perform the sum operation $(f(input) + input)$ when input and output do not have the same size, we project the input into the output space using another convolutional layer (the one in green in Fig. 5) before the element-wise sum, X is the number of output feature maps and Y the number of input feature maps of the unit. Adapting the architecture proposed in [34], a very deep architecture with 20 layers, most of them residual, has been created (DR-CNN). Fig. 6 shows the proposed architecture. The three final layers are fully connected. In this case, a more adequate activation function PReLU [35] $(f(x) = max(-\epsilon x, x),\ 0 < \epsilon << 1)$, which allows activation for negative values, is used. The main effect of the PReLU function is to improve the behaviour of the gradient estimation in the bottom layers. The number of free parameters is 2,847,138.

Once we have described the three CNN architectures used for the DLA approach, we study

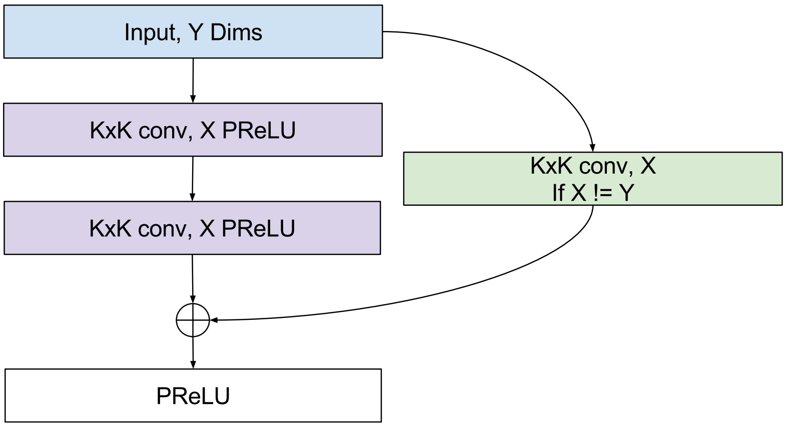
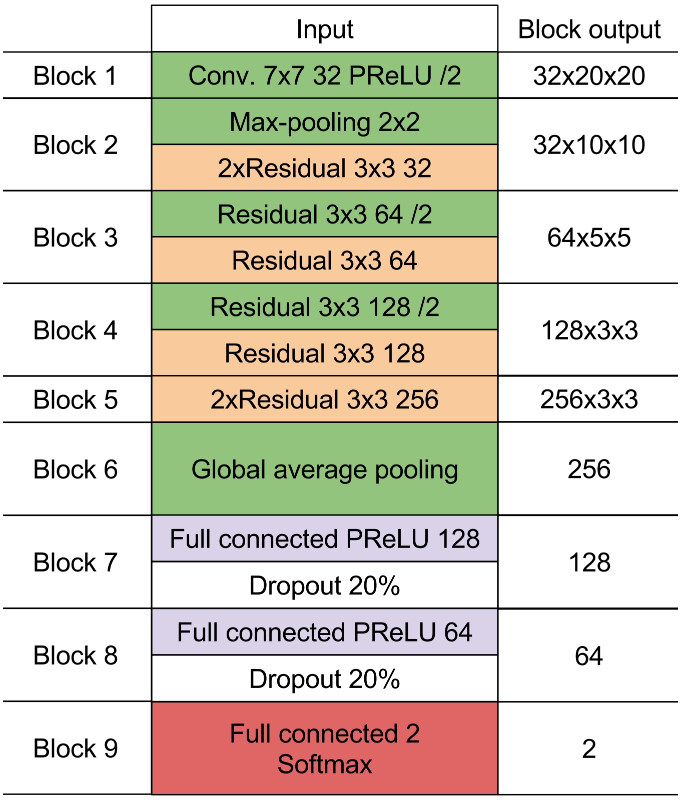


Fig. 5: Residual unit used in Deep-Residual-CNN. The input is transformed throughout two convolutional layers with PReLUs to learn the residual. See in text how the sum operation is performed.



	Input	Block output
Block 1	Conv. 7x7 32 PReLU /2	32x20x20
Block 2	Max-pooling 2x2	32x10x10
	2xResidual 3x3 32	
Block 3	Residual 3x3 64 /2	64x5x5
	Residual 3x3 64	
Block 4	Residual 3x3 128 /2	128x3x3
	Residual 3x3 128	
Block 5	2xResidual 3x3 256	256x3x3
Block 6	Global average pooling	256
Block 7	Full connected PReLU 128	128
	Dropout 20%	
Block 8	Full connected PReLU 64	64
	Dropout 20%	
Block 9	Full connected 2 Softmax	2

Fig. 6: Description of the Deep-Residual-CNN (DR-CNN) architecture. The residual units correspond to the standard residual unit described in [34], see Fig. 5. The total number of parameters is 2,847,138. See dropout x% meaning in Fig. 3.

now the two SGA methods. Our problem can be considered a segmentation task where the regions of interest are defined by the grouping of high-level gray pixels using context conditions. When the main goal is to learn features encoding the location of regions of interest, the use of CNNs has proven to be a successful approach [36]. A typical segmentation architecture is shown in

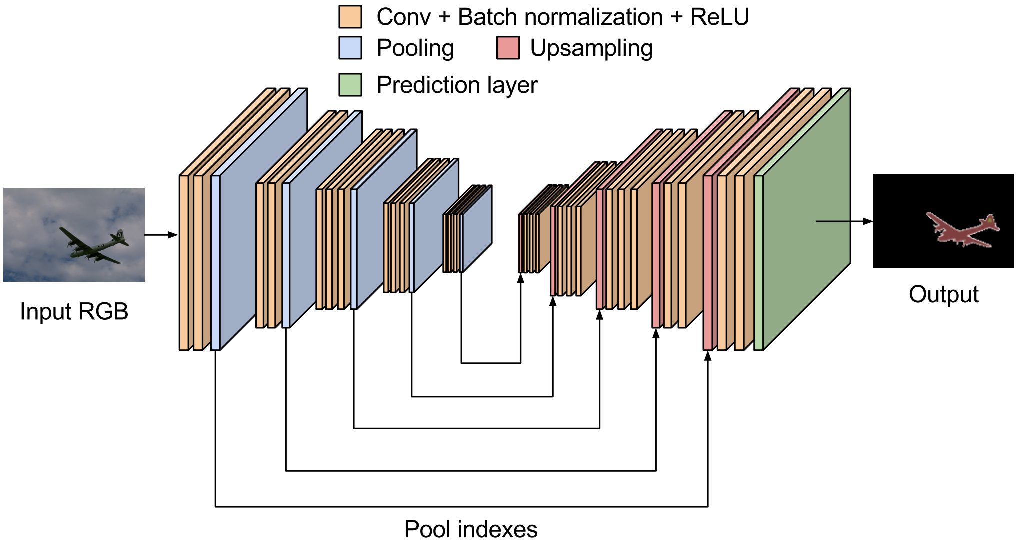


Fig. 7: Segnet architecture as described in [36].

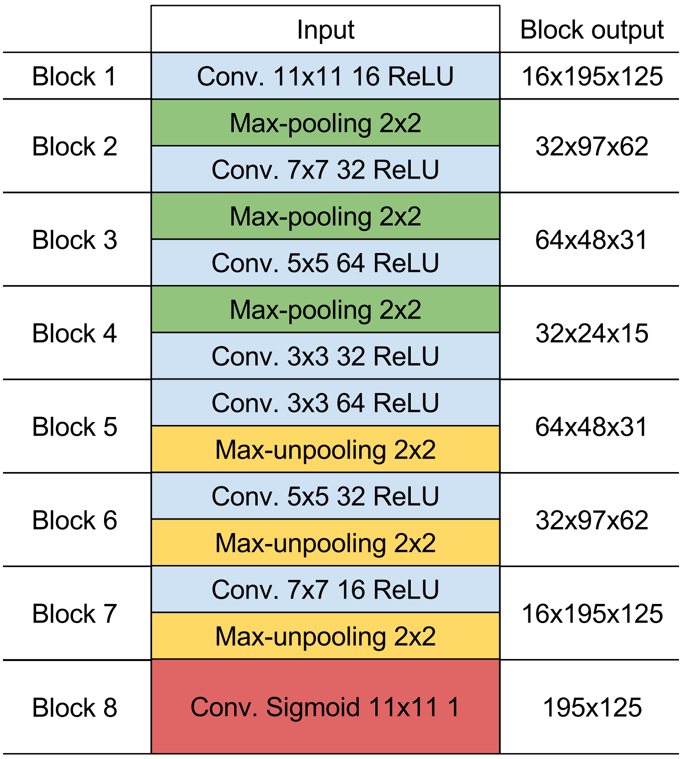
Fig. 7. The feature learning process takes place on the left side of the architecture (decreasing in size layers). The smallest layer on the left represents the learned features which encode threat locations. Since the loss function used to learn the architecture must be evaluated on all image pixels, the layers on the right side (increasing in size layers) decode the estimated label, in our case, a probability of containing a threat from the learned features. In summary, the SGA approach uses architectures that take as input the whole image I and output for each pixel a probability distribution P on the label set.

This approach has been used recently in image segmentation tasks [36] on natural images. The original model, Segnet, has 29,458,886 parameters where the encoding layers are designed by the combination of convolution, ReLU and max-pooling layers. In the decoding part (right side of Fig. 7), the model carries out the spatial propagation of the high level features using an inverse process named max-unpooling. The combination of convolutional, max-unpooling, and ReLU layers defines the spatial propagation of the high level features found by the encoder until the size of the input image has been matched. The max-unpooling mapping only assigns values to the units whose indexes were local maxima in the corresponding max-pooling operation of the encoding part.

The Segnet architecture was designed to solve a much more complex (in terms of class number) problem than ours. We adapt this approach to our task by designing two new architectures but keeping the encoder-decoder approach. The first model, shown in Fig. 8, has 193,827 parameters. We refer to it as SCNN. The probability distribution for each pixel is obtained using a sigmoidal activation on the output of the last convolutional layer to map the values to the $[0, 1]$ interval.

It is important to note here that autoencoders and, more recently, Generative Adversarial Networks are generative deep networks architectures also using an encoding-decoding approach [37]. The main goal of these architectures is to learn features and parameters to represent the generative distribution of the samples. Although this learning problem is clearly much more difficult and complex than ours, the use of deep generative models on our data is an area worth to explore.

The second model makes use of multiscale information. Multiscale architectures have been used to improve the performance of CNNs [38]. The use of multiscale features for PMMWI classification with shallow classifiers [19], [20] has also provided better classification results. This insight has led us to modify the SCNN in order to incorporate information at three different scales (s, $s/2$ and $s/4$). We process each scale to produce k feature maps using an SCNN on each scale. We then combine the $3 * k$ feature maps with 2 convolutional layers after resizing the output of the lower scales. Finally, for each pixel the probabilities are computed from regularized convolutional layer and a sigmoidal activation. To eliminate possible redundancies between scales, we apply spatial dropout (instead of dropping a single unit, we drop an entire feature map) before the scale combination. We call this model SCNN-MS, it has 355,515 parameters. Its architecture can be seen in Fig. 9.



	Input	Block output
Block 1	Conv. 11x11 16 ReLU	16x195x125
Block 2	Max-pooling 2x2	32x97x62
	Conv. 7x7 32 ReLU	
Block 3	Max-pooling 2x2	64x48x31
	Conv. 5x5 64 ReLU	
Block 4	Max-pooling 2x2	32x24x15
	Conv. 3x3 32 ReLU	
Block 5	Conv. 3x3 64 ReLU	64x48x31
	Max-unpooling 2x2	
Block 6	Conv. 5x5 32 ReLU	32x97x62
	Max-unpooling 2x2	
Block 7	Conv. 7x7 16 ReLU	16x195x125
	Max-unpooling 2x2	
Block 8	Conv. Sigmoid 11x11 1	195x125

Fig. 8: Description of the SCNN architecture. The total number of parameters is 193,827. The max-unpooling layer is the same used in the segnet architecture [36].

IV. The approach

Given an image, the goal is to classify it as Class +1 (positive) when it contains at least one concealed object or Class 0 (negative) otherwise. We also need to localize where the concealed object is on the human body.

Let $\mathcal{D} = \{\mathcal{I}_i, i = 1, \cdots, N_I\}$ be the image dataset. Our goal is to learn from $\mathcal{D}$ a function $f_{\mathcal{D}}(\mathcal{I}_i) = \mathcal{X}_i$, with the lowest generalization error. This function assigns to each image the set of pixels, $\mathcal{X}_i$, where threats are located or an empty set if no threat is present. This problem, as

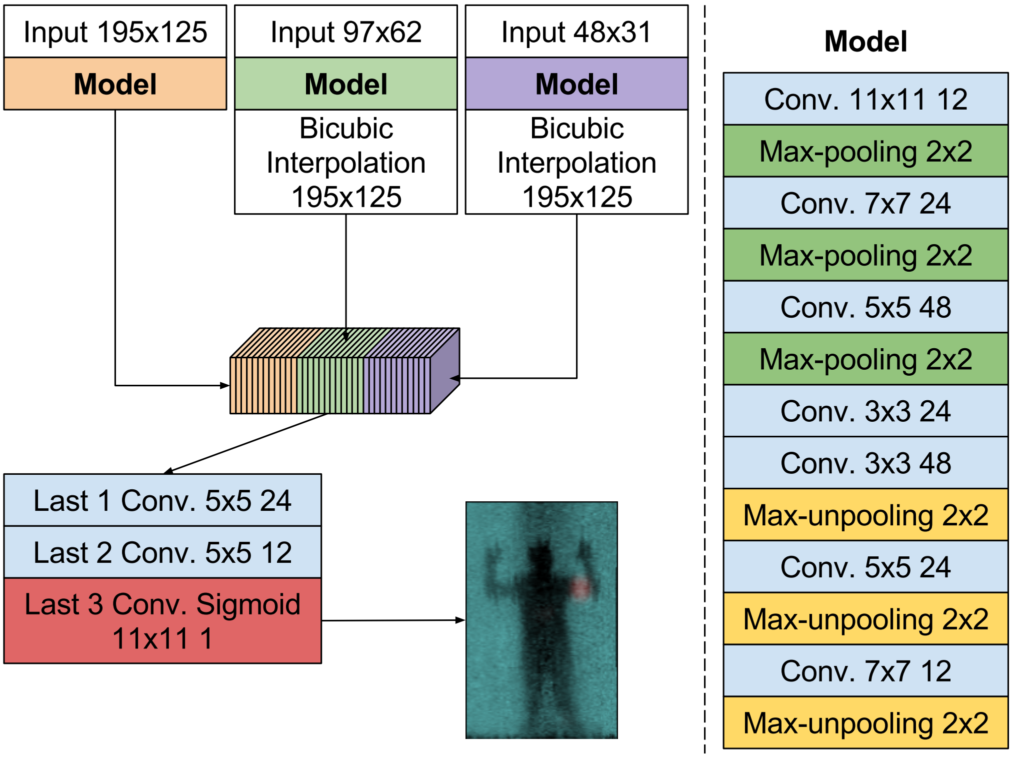


Fig. 9: Description of the SCNN-MS architecture. The total number of parameters is 355,515. The red area contains pixels with high probability of being threat.

we have already indicated, can be cast as either an object detection/localization problem using local image patches or a segmentation one when the whole image is used, both are approached here using machine learning techniques. We define the function $f_{\mathcal{D}}$ as the composition of two functions; one detecting potential threat regions and the other deciding which of the regions should be considered a real threat. The full dataset $\mathcal{D}$ is used to learn the function, $f_{\mathcal{D}}$, and its generalization error is estimated by cross validation.

A. The training set $\mathcal{T}$

We start by splitting the set of images $\mathcal{D}$ in five folds to estimate the generalization error of $f_{\mathcal{P}}$ by five fold cross-validation. Four folds are used for training and one for validation. Each fold has the same number of images with one, two o more threats. Due to the small size of the threat regions and the possibility of having several threats in the same image, a possible approach to learn the detection function is to use image patches. A patch is a rectangular image piece centered on a pixel. By $\mathbf{x}_j^P$ we denote the patch $\mathbf{x}^P$ centered on pixel j. Let P_I be a mapping which selects a subset of patches $\mathcal{P}_I^i$ from the image $\mathcal{I}_i$, that is $P_I(\mathcal{I}_i) = \mathcal{P}_I^i$. Each one of the patches is binary labelled, $(+1)$ or (-1), depending on whether it contains a threat or not. The patch dataset associated to each image fold is defined by $\mathcal{T}_j = \cup_{k=1}^{N_I} \mathcal{P}_I^k$ $j = 1, \cdots, 5$. On each cross-validation iteration, the union of four of the patch datasets is used to learn the corresponding detection function $f_{\mathcal{P}_j} : \mathcal{T}_j \to [0, 1]$ and the other is used for validation. A binary classifier is built by splitting the $f_{\mathcal{P}_j}$ range using a calibrated threshold.

Taking into account the object sizes described in appendix B, three different patch scales can be identified as representatives of the object sizes: 39×39, 19×19, and 9×9 pixels, respectively. From each image we only extract the 39×39 patches that are fully included inside the image. The use of the largest size patches forces the model to learn threats in different spatial contexts. Later on we show that the joint use of the three scales is relevant when the whole image is considered as input.

The full set of training patches extracted from four image folds is huge (see appendix B). Then, to reduce the computational cost of the training process, we extract per image one patch every 2×2 pixels, obtaining a total of $3,476$ patches per image. In total we have $11,502,084$ patches of size 39×39. We have found no loss in classification performance by this training dataset reduction. The 39×39 patches covering a full object are positively labelled (1), the rest are negatively labelled (-1). Patches that partially overlap an object are not included in the training dataset. We have $372,494$ positive instances and $9,351,130$ negative ones. Since the number of negatives is around 25 times the number of positives and taking into account that most of the negative patches are very similar, we have used for training a subset of them. Specifically, we keep only one negative sample every 2×2 block, so we retain $1/4$ of the negative patches we have. The final number of negative samples, $2,337,782$, is then around six times the number of positive ones.

When the complete image is considered as input, the learning process is simply a standard five fold cross-validation.

B. Learning $f_{\mathcal{P}_j}$

To label the 39×39 image patches as threat or non-threat, we train the [LD,MD,HD]-CNN architectures. To deal with the imbalanced training set, a sampling strategy is used. We partition the set of negative patches and learn an ensemble of classifiers. Let $\mathcal{T}_P, \mathcal{T}_N$ be the subsets of positive and negative labelled patches, respectively, and $\mathcal{T} = \mathcal{T}_P \cup \mathcal{T}_N$. Let us denote by n_P and n_N their cardinality, being $n_P << n_N$ with $n_C = n_N/n_P$. Let $\mathcal{T}_N = \cup_{k=1}^{n_C} \mathcal{T}_N^k$ be a random decomposition of $\mathcal{T}_N$ in n_C disjoint subsets. We solve n_C learning problems, $f_{\mathcal{P}}^k,\ k = 1, \cdots, n_C$, associated to the training sets defined by $\{\mathcal{T}_N^k \cup \mathcal{T}_P\}$, respectively. We repeat the same procedure t times obtaining an ensemble of $t \times n_C$ functions $f_{\mathcal{P}}^k$ which use the positive and subsets of the negative patches. Then, the patch classifier is defined using a threshold on the range of the weighted average of the $f_{\mathcal{P}}^k$, this is $f_{\mathcal{P}} = \sum_k w_k f_{\mathcal{P}}^k$ with weights $w_k > 0$ and $\sum_k w_k = 1$, alternatively as a weighted majority voting on the set of classifier provided by $\{f_{\mathcal{P}}^k\}$. In our experiments we have used the majority voting rule to decide whether a patch is a potential threat, and its detection score is measured by $\frac{1}{t \times n_C} \sum_k f_{\mathcal{P}}^k(\mathbf{x}_P)$. In the experiments we use $n_C = 6$ and $t = 1$. This means that the $\mathcal{T}_N$ subset is split in six random partitions and each partition together with the $\mathcal{T}_P$ subset is used to fit a new model.

The segmentation approaches represented by SCNN and SCNN-MS architectures use full images as input. In these cases, the label is defined by a binary matrix of the same size as the input one, with ones in pixels covered by the threats and zeros in the rest.

C. Initialization and loss function

In all cases, the initialization of the weights of the layers is done by random sampling from a Gaussian distribution with $\mathrm{mean} = 0$ and $\mathrm{std} = \sqrt{\frac{2}{\#\mathrm{input\_unit}}}$ [35] , where $\#\mathrm{input\_unit}$ represents the number of input units to each layer.

The fitting loss function used by all our models is the negative log likelihood given by

$$L(\mathbf{x}, \mathbf{y}; \theta) = -\sum_{j} y_j \mathrm{log} p(c_j|\mathbf{x}), \tag{1}$$

where $\mathbf{y}$ represents the target distribution and $p(c_j|\mathbf{x})$ denotes, for all the models, the probability of class j defined by the softmax function of the output layer.

D. Hyperparameters

The LD-CNN, MD-CNN and DR-CNN models were trained using Stochastic Gradient Descend (SGD) with minibatches of size 64, variable learning rate, and 0.9 momentum. The variable learning rate policy follows a triangular scheme [39] that consists of varying the learning rate between a minimum and a maximum value following a triangular pattern with the training iterations. The triangular learning rate parameters range from 0.01 to 0.04 for the LD-CNN and MD-CNN models and from 0.01 to 0.03 for the DR-CNN model. All these models have been trained using a total of 4 epochs, two of them increasing uniformly the learning rate value with the iterations from the minimum until reaching the maximum and two more going back, that is, we decrease the learning rate similarly from the maximum to the minimum value. We apply batch normalization before each nonlinear transformation. Also we perform local brightness and contrast normalization on each image [40], that is, we remove the mean and divide by the standard deviation of its elements.

SCNN and SCNN-MS were also trained using SGD and 0.9 momentum, additionally we had to use the Nadam [41] learning rate estimation for the method to converge. In these cases, minibatches of size 16 were used. We fix Nadam hyper-parameters to the standard values except the learning rate, which was set to 0.02. In SCNN-MS the learning rate was set to 0.002 after the first 16 epochs. SCNN was trained for 16 epochs, while SCNN-MS used 20 epochs. In all cases, batch normalization before each nonlinear transformation and before the first layer was also applied.

E. Regularization

All the models were regularized during training using weight decay with a 0.005 fixed value. Dropout regularization was also applied to the LD-CNN, MD-CNN and DR-CNN models. For the LD-CNN model a 50% dropout layer before every full connected layer was used. As suggested in [33], for the MD-CNN model a 20% dropout layer on the input data and 50% after every subsampling phase was introduced (see Fig. 4). In the DR-CNN models we only apply a 20% dropout to the output of the layers 15 and 18 (see Fig. 6).

To improve the convergence of the fitting process for our SCNN and SCNN-MS models, we corrupt the input image with random Gaussian noise of 0 mean and 0.05 standard deviation. Also random Gaussian noise of 0 mean and 0.0005 standard deviation was applied to the output of each layer (except for the last one). Spatial dropout with 50% intensity was applied to SCNN-MS after concatenating the three scales feature maps.

F. Detection and validation

The ROC curve associated to each trained model is used to calibrate its detection threshold, $\mathrm{thr}_{\mathrm{M}} \in [0, 1]$. $\mathrm{Pr}_{\mathrm{M}}(S) \geq \mathrm{thr}_{\mathrm{M}}$, M∈ {LD-CNN, MD-CNN, DR-CNN, SCNN, SCNN-MS}, will declare a potential threat in S. Here S represents a patch or the whole image.

The LD-CNN, MD-CNN, and DR-CNN models provide probabilities on overlapping patches. To fuse this information a non-maximum suppression step is carried out. We reject a patch $\mathbf{x}_j^P$ if it has an intersection-over-union overlap larger than a learned threshold with a higher scoring patch $\mathbf{x}_i^P$. That is, if $\mathrm{Pr}_{\mathrm{M}}(\mathbf{x}_i^P) > \mathrm{Pr}_{\mathrm{M}}(\mathbf{x}_j^P) \geq \mathrm{thr}_{\mathrm{M}}$, then the patch $\mathbf{x}_j^P$ is removed (see Fig. 10).

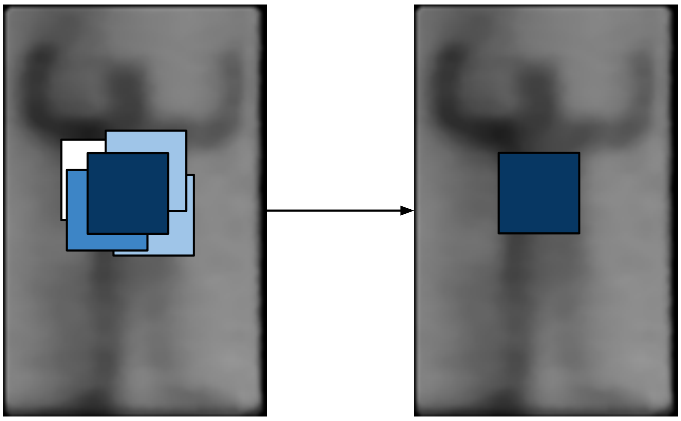


Fig. 10: The non-maxima suppression process. Blue rectangles indicate regions whose probabilities are above the minimum probability threshold (darker regions correspond to higher probability). Since these regions overlap more than the overlapping threshold, we only maintain the one with highest probability (in the image, the darkest rectangle).

Let us denote by $\mathcal{H}_{\mathcal{S}}$ and $\mathcal{P}_{\mathcal{S}}$ the support region (set of pixels) in the image associated to a hidden object and a patch respectively. In the experiments, we consider that a patch $\mathbf{x}^P$ is correctly classified as positive when $f_{\mathcal{P}}(\mathbf{x}^P) > \mathrm{thr}_{\mathrm{M}}$ and $\mathcal{H}_{\mathcal{S}} \cap \mathcal{P}_{\mathcal{S}}(\mathbf{x}^P) > 0.5 * \mathrm{size}(\mathcal{H}_{\mathcal{S}})$. A hidden object in the image is correctly detected when there is at least one patch which satisfies both conditions. Finally, an image is correctly classified as positive when, at least, a hidden object has been detected.

For SGA methods, the above approach is applied to the full image. We consider in each image the same patch structure used for patch classification.

V. Experimental setup

Our experimental setup has been designed to answer the following questions: a) what is the influence of the image noise on the models?; b) how relevant is the depth to the detection?; c) is it important to use multiscale information?; d) should deep learning models be used to solve this task?.

We ran experiments with all the architectures in two scenarios, raw and filtered images (see appendix A). This allows us to evaluate when smoothing the image helps the detection process. AUC, TP, and FP figures of merit are calculated as the average of the five values obtained from each cross-validation set. Means and standard deviations are included. AUC represents the area under the ROC curve when the validation images are classified as containing or not containing hidden objects. TP indicates the percentage of detected hidden objects. FP represents the average number of false positives per image. The threshold and overlap parameters used when non-maxima suppression was applied are also reported (see subsection IV-F).

We performed all our experiments using Caffe [42] and Keras [43] with Theano [44]. We used an Intel Xeon E5-2630 v3 at 2.40GHz with 8 cores and 256GB of Ram and 1 GPU Nvidia Titan X (Pascal) with 12GB of Ram.

VI. Discussion

Table I shows the obtained classification results with CNN architectures. It also shows how the best shallow architecture: Random Forest concatenating three scales of Haar features (RF-HAAR-MS) [19], [20] performs. We have also evaluated the role of regularization on the learning process for the SCNN and SCNN-MS models (adding noise and spatial dropout). The obtained results with no regularization, not included here, show an important decrease in their figures of merit when regularization is not used.

A comparison among the different classifiers trained from image patches, [LD,MD,DR]-CNN and RF-HAAR-MS, shows that an important improvement is obtained when very deep models are used (DR-CNN). The LD-CNN model has a much lower score than the shallow RF-HAAR-MS. This indicates that a simple architecture with a low number of layers is unable to deal with the image noise to extract good features. The MD-CNN performs slightly better than RF-HAAR-MS pointing out that increasing the number of layers introduces robustness against non-stationary noise. The DR-CNN model obtains a better score, with an AUC value of almost 79%. This shows that deep enough models are able to extract robust and informative features on noisy scenarios. Architectures using preprocessed images are noted *-pre in Table I.

TABLE I: CNN results. These models are defined by a committee of classifiers using raw and preprocesed (pre) images respectively. For some architectures we also show the result using preprocessed images, only the best performing ones are shown. See the text for additional explanation on the columns.

Model	AUC $\times 10^2$	TP$\times 10^2$	FP	Thr	Overlap
LD-CNN	55.3 ± 2.2	92 ± 3.7	7.1 ± 7.0	0.7	0.4
MD-CNN	70.9 ± 1.7	92 ± 1.9	6.3 ± 5.9	0.6	0.3
DR-CNN	78.8 ± 1.8	95 ± 2.2	4.2 ± 4.1	0.5	0.3
DR-CNN-pre	69.3 ± 2.1	95 ± 1.1	6.2 ± 6.0	0.65	0.3
SCNN	86.2 ± 1.4	100	0.02 ± 10^{-5}	0.65	0.5
SCNN-pre	83.4 ± 2.7	100	0.03 ± 0.04	0.65	0.5
SCNN-MS	$\mathbf{87.4 \pm 0.5}$	**100**	$\mathbf{0.006 \pm 10^{-6}}$	**0.65**	**0.5**
SCNN-MS-pre	82.1 ± 4.2	100	0.1 ± 0.12	0.65	0.5
RF-HAAR-MS	75.3 ± 1.6	94 ± 0.9	4.0 ± 3.8	0.7	0.3

The best performing CNN methods are the ones fed with full images instead of patches. They can deal better with the spatial non-stationarity image noise and take advantage of analyzing the whole image at once. SCNN performs very well, reaching a 86% AUC, a 100% rate of positive detection and average of 0.02 false positives per image (FP). In comparison with the patch based architectures, these figures represent an enormous improvement. When multiscale information is considered in SCNN-MS, an additional improvement is achieved, reaching 87.4% AUC while at the same time decreasing significantly the FP rate by nearly one order of magnitude. As it can be observed, the best score, in bold, shows an over 12 point improvement over the score obtained by the best shallow architecture, RF-HAAR-MS, with the added benefit of having eliminated the feature extraction process. In terms of detection errors (TP and FP), these deep learning architectures also show an overwhelming improvement over the shallow ones. These findings clearly point out the negative influence of the image noise on the shallow architectures and their difficulty to remove the noise in the classification process. On the contrary, the deep learning models show robustness to image noise when the number of layers increases. It is also interesting to note that image processing does not help any CNN model. This could be explained by noticing that any image preprocessing technique introduces new spatial dependencies on the image values, making a much harder task for the model to extract uncorrelated features. In other words, the preprocessing step masks the signal and makes more difficult for the models to extract uncorrelated features. Although the reported figures correspond to applying one particular filter, see Appendix A, we also tried other filters with similar results.

Figure 11 shows some examples of the most relevant activation maps of the SCNN-MS model "last 2 Conv" layer (see Fig. 9). The first column contains the input images, columns Map [1-3] show the most relevant activation maps contributing to the classification layer. This layer before the sigmoidal function is applied is shown in the fifth column (see Before-σ column). The other nine activation maps do not contain much internal structure. They do not contribute much to the detection process. However, if they are removed the performance of the method decreases slightly. The last column shows the input image with the detected threat in green. The first two rows (True Positive) show examples where our approach detects a true threat. It can be seen that the areas with high activation in the column Before-σ correlate with lack of activation in the maps. In rows three and four (True Negative) the behavior is similar, but the intensity of the final activation is smaller and after the sigmoidal transformation it is eliminated. The last two rows (False Positive) show two examples where our model fails. In this case no threat is present but the model identifies a false positive. The discussion above suggests that the filters look for areas of high contrast in the images. Since our model detects all the threats in the database (see Table I), no false negative examples are shown.

We finally report training and test times. Obviously, the training time depends on the number of free parameters to be estimated and the number of convolutions to be carried out. Table II depicts the training and test computing times for all the CNN architectures. It is interesting to note the high influence of the architecture design on the training and test figures. Clearly those models trained using image patches require higher training times due to a much larger training set, additionally the layer design has a strong influence, see the [LD,MD,HD]-CNN models. The MD-CNN model needs much more time than the other models due to the higher number of convolutions associated to the higher number of filters. On the contrary, the SCNN model is much faster in both training and test times. This property together with its high efficiency makes it a very good candidate for real-time applications.

TABLE II: CNN training and test times (h hours, m minutes and s seconds). We show the one fold mean training time, the test time over all the images, and the mean time per image.

Model	Training one fold	Total test	Per image test
LD-CNN	1h and 32m	47.42m	0.86s
MD-CNN	14h and 27m	746.82m	13.54s
DR-CNN	9h and 50m	73.90m	1.34s
SCNN	**2m and 24s**	**48s**	**0.015s**
SCNN-MS	5m and 52s	2m and 24s	0.044s
RF-HAAR-MS	19h and 12m	12m and 20s	0.22s

VII. Conclusions

In this paper we have analyzed the use of CNN-DL classifiers to detect and localize concealed objects in PMMWIs approach. We have shown that to address the non-stationary noise present in the images, deep architectures should be used. They show an overwhelming improvement over the best performing shallow ones. We have also found that the two most important factors for this improvement have been the depth of the architecture and feeding the models with the whole image. The combined use of information from different scales improves the detection score and also greatly reduces the classification errors. The regularization process applied to our model has shown to be crucial in order to find a model with a high detection score (AUC, TP) and a low error rate (FP).

The obtained results allow us to conclude that our DL-CNN architectures define a new state of the art for the task of detecting and localizing concealed objects in PMMWIs. In addition, we have found that deep architectures are a very good tool to deal with signals immersed in non-stationary noise. Our experimental set-up uses the new UGR-PM^{2}WI dataset, to the best of our knowledge, the currently most comprehensive available dataset for this task.

Appendix

A. Image Enhancement

A natural question to consider is whether image filtering (smoothing, contrast enhancement, etc.) helps the detection process. Fig. 12 shows examples of applying mean (b), median (c), and bilateral (d) filters to the observed image, Fig. 12(a).

We observed that a better smoothing criterion is to assign to each pixel an estimation of the most frequent value in its neighbourhood. That is, for each pixel i, let $\mathbf{y}_i^{\mathrm{B}}$ denote a block around y_i in the observed image $\mathbf{y}$, $z_i(1), \ldots, z_i(K)$ denote K independent samples with replacement from $\mathbf{y}_i^{\mathrm{B}}$. We then redefine

$$q_i = \frac{\sum_{k=1}^{K} z_i(k)}{K}, \tag{2}$$

assign $\mathbf{y} = \mathbf{q}$ and repeat the process L times. The final processed image is

$$x_i = \text{median}(\mathbf{y}_i^B) \tag{3}$$

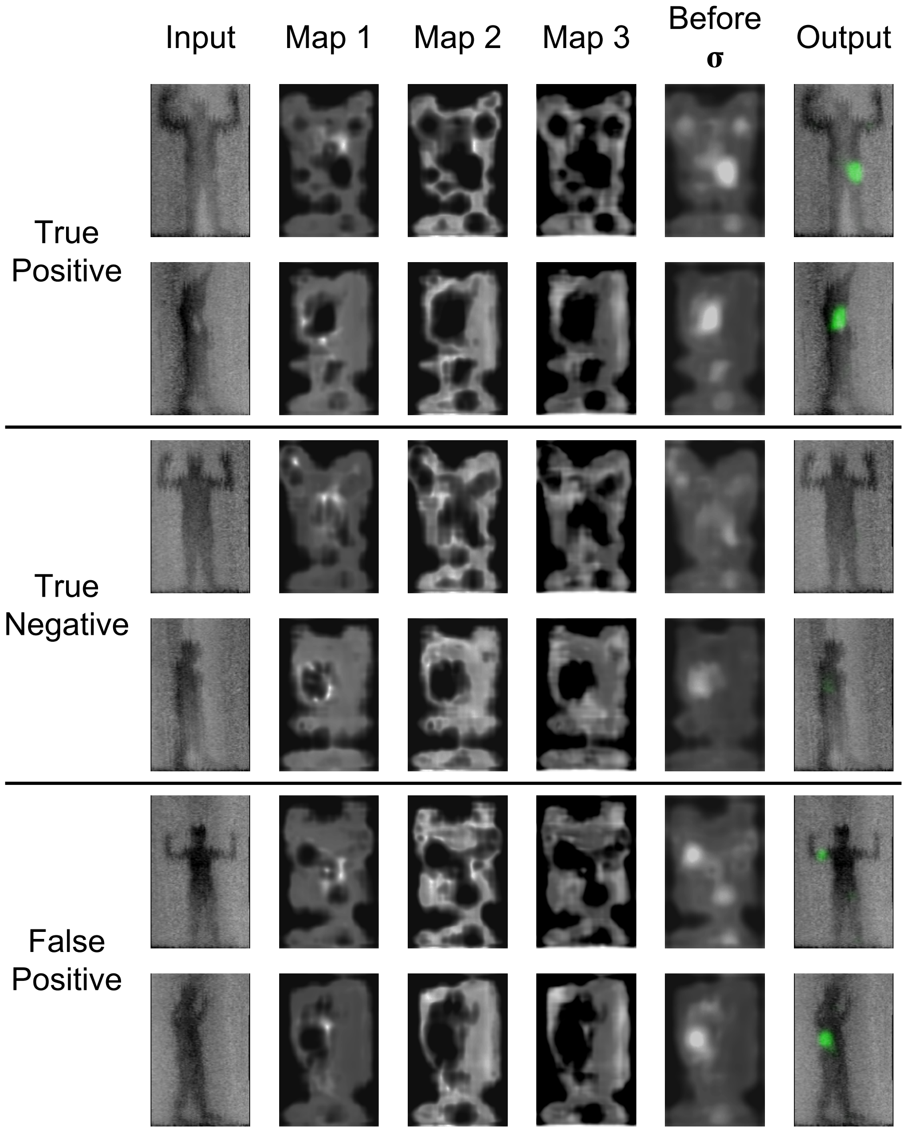


Fig. 11: Examples of the most relevant activation maps of the SCNN-MS “Last 2 Conv” layer. The first column contains the input images. Columns 2-4 display the activations of the most relevant filters. The fifth column shows the activations of the last convolutional layer just before the sigmoidal (σ) activation. The last column shows in green the predicted probabilities for each pixel over the input images (higher probabilities are associated to greener intensities). Notice that low responses in the feature maps correspond to high responses before the sigmoidal and the predicted highest probability zones (see text). Activation layers have been rescaled for better visualization.

We have found experimentally that $B = 5 \times 5$ and $L = 5$ produce good processed images, see Fig. 12(e). Notice that some contrast improvement can be observed. See the experimental section to analyse how image processing influences the performance of the proposed classifiers.

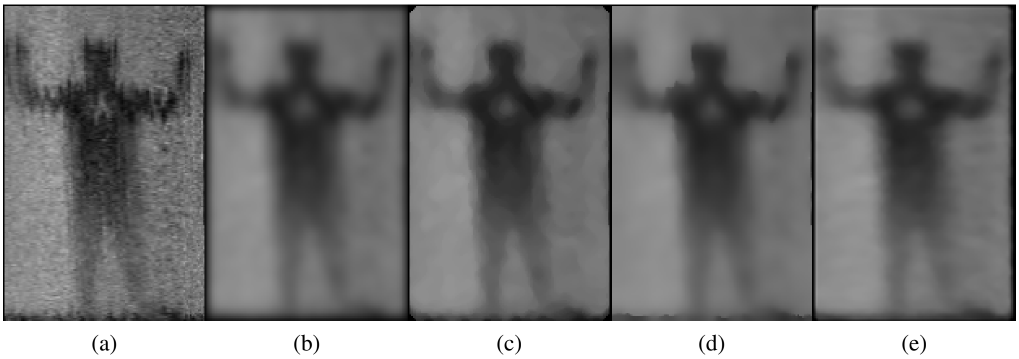


Fig. 12: Results of applying different image filtering (Best view in high resolution screen), see text for details.

B. UGR-PM^{2}WI Dataset

A comprehensive dataset of PMMWIs has been created. It consists of 3309, 125×195 images of 33 different people. Hidden objects are in the range of 35×39 to 10×10 pixels, which in our images corresponds roughly to $2752.39cm^2$ and $201.64cm^2$, respectively. Smaller hidden objects are not considered to be relevant.

We took pictures of different people (with different complexions and heights) wearing 12 different objects in 10 body locations: forearm, chest, stomach, thigh, ankle (front), waist (side), armpit (side), arm, ankle (lateral), thigh (lateral), and 2 images without any objects. Images of people wearing simultaneously two objects in different locations were also taken. In summary, the dataset consists of 463 pictures of people with no object, 2144 containing one object and 702 containing 2 objects.

Threats were simulated by bags containing objects/substances with different millimeter wave responses: a cutter, 325g of gel, a 200g clay bar, a simulated gun, 200g of sugar, 200g of frozen peas, 150ml of cologne, 160g of gel, a bag with metal pieces, 200g of flour, a 50cl water bottle, and a 250ml hydrogen peroxide bottle (see Fig. 13). Notice that different object sizes were used.

Fig. 14 shows PMMWIs of subjects with simulated threats on different locations. It is important to note that although in the experiments objects are visible and not hidden under clothing, this is irrelevant to PMMW sensors.

The visual images were taken at the same time as the millimeter ones. Objects are marked on the visual images by the smallest bounding boxes containing them. To transfer object bounding boxes from visible images to PMMWI ones, a homography, estimated from the visible image plane to the PMMWI one using a calibration pattern, was applied. These bounding boxes will later be used to assess the performance of the classifiers.

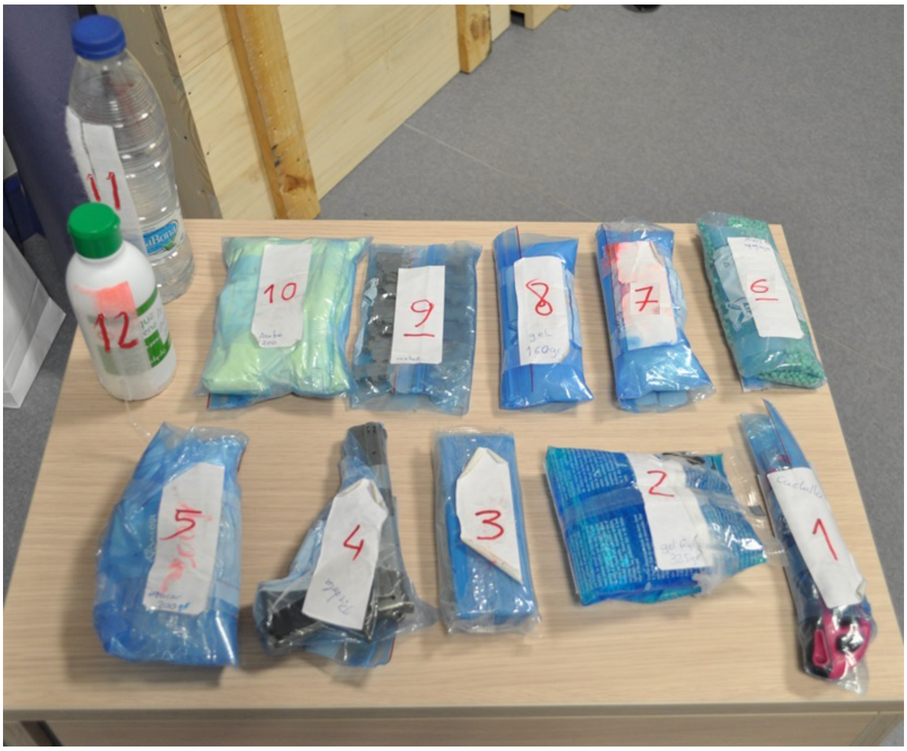


Fig. 13: Simulated threats in the dataset: a cutter (1), 325g of gel (2), a 200g clay bar (3), a simulated gun (4), 200g of sugar (5), 200g of frozen peas (6), 150ml of cologne (7), 160g of gel (8) , a bag with metal pieces (9), 200g of flour (10), a 50cl water bottle (11), and a 250ml hydrogen peroxide bottle (12).

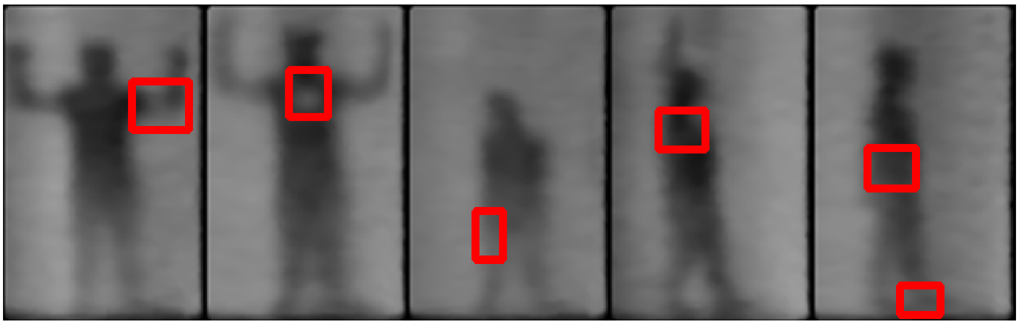


Fig. 14: Examples of PMMWIs with (hidden) objects. Red boxes indicate object locations. Objects to be detected correspond to whiter areas on the body.

References

[1] J. Accardo and M. A. Chaudhry, "Radiation exposure and privacy concerns surrounding full-body scanners in airports," *Journal of Radiation Research and Applied Sciences*, vol. 7, no. 2, pp. 198 – 200, 2014.

[2] N. E. Alexander, C. Callejero Andrés, and R. Gonzalo, "Multispectral mm-wave imaging: materials and images," *Proceedings of Society of Photo-Optical Instrumentation Engineers (SPIE)*, vol. 6948, pp. 694803–694803–10, 2008.

[3] L. Yujiri, M. Shoucri, and P. Moffa, "Passive millimeter wave imaging," *IEEE Microwave Magazine*, vol. 4, no. 3, pp. 39–50, 2003.

[4] L. Yujiri, "Passive millimeter wave imaging," in *2006 IEEE MTT-S International Microwave Symposium Digest*, pp. 98–101, 2006.

[5] S. Oka, H. Togo, N. Kukutsu, and T. Nagatsuma, "Latest Trends In Millimeter-wave Imaging Technology," *Progress in Electromagnetics Research Letters*, vol. 1, pp. 197–204, 2008.

[6] S. D. Babacan, M. Luessi, L. Spinoulas, A. K. Katsaggelos, N. Gopalsami, T. Elmer, R. Ahern, S. Liao, and A. Raptis, "Compressive passive millimeter-wave imaging," in *2011 18th IEEE International Conference on Image Processing*, pp. 2705–2708, 2011.

[7] W. Saafin, S. Villena, M. Vega, R. Molina, and A. K. Katsaggelos, "Compressive sensing super resolution from multiple observations with application to passive millimeter wave images," *Digital Signal Processing*, vol. 50, pp. 180–190, 2016.

[8] B. Han, J. Xiong, L. Li, J. Yang, and Z. Wang, "Research on millimeter-wave image denoising method based on contourlet and compressed sensing," in *Signal Processing Systems (ICSPS), 2010 2nd International Conference on*, vol. 2, pp. V2–471–V2–475, 2010.

[9] J. Mateos, A. López, M. Vega, R. Molina, and A. Katsaggelos, "Multiframe blind deconvolution of passive millimeter wave images using variational Dirichlet blur kernel estimation," in *IEEE International Conference on Image Processing*, pp. 2678–2682, 2016.

[10] J. Yang, J. Wang, and L. Li, "A new algorithm for passive millimeter-wave image enhancement," in *Signal Processing Systems (ICSPS), 2010 2nd International Conference on*, vol. 3, pp. V3–507–V3–511, 2010.

[11] W. Yu, X. Chen, S. Dong, and W. Shao, "Study on image enhancement algorithm applied to passive millimeter-wave imaging based on wavelet transformation," in *Electrical and Control Engineering (ICECE), 2011 International Conference on*, pp. 856–859, 2011.

[12] C. Haworth, B. Gonzalez, M. Tomsin, R. Appleby, P. Coward, A. Harvey, K. Lebart, Y. Petillot, and E. Trucco, "Image analysis for object detection in millimetre-wave images," in *Passive Millimetre-wave and Terahertz Imaginh and Technology*, vol. 5619, pp. 117–128, 2004.

[13] C. Haworth, Y. Petillot, and E. Trucco, "Image processing techniques for metallic object detection with millimetre-wave images," *Pattern Recognition Letters*, vol. 27, no. 15, pp. 1843 – 1851, 2006.

[14] O. Martínez, L. Ferraz, X. Binefa, I. Gómez, and C. Dorronsoro, "Concealed object detection and segmentation over millimetric waves images," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*, pp. 31–37, June 2010.

[15] I. Gómez Maqueda, N. Pérez de la Blanca, R. Molina, and A. Katsaggelos, "Fast millimetre

wave threat detection algorithm," in *European Signal Processing Conference (EUSIPCO 2015)*, pp. 599–603, Nice (France), September 2015.

[16] S. Yeom, D.-S. Lee, Y. Jang, M.-K. Lee, and S.-W. Jung, "Real-time concealed-object detection and recognition with passive millimeter wave imaging," *Optics Express*, vol. 20, pp. 9371–9381, Apr 2012.

[17] W. Yu, X. Chen, and L. Wu, "Segmentation of concealed objects in passive millimeter-wave images based on the gaussian mixture model," *Journal of Infrared, Millimeter, and Terahertz Waves*, vol. 36, no. 4, pp. 400–421, 2015.

[18] S. Yeom, D.-S. Lee, and J.-Y. Son, "Shape feature analysis of concealed objects with passive millimeter wave imaging," *Progress in Electromagnetics Research Letters*, vol. 57, pp. 131–137, 2015.

[19] S. López-Tapia, R. Molina, and N. Pérez de la Blanca, "Detection and localization of objects in passive millimeter wave images," in *24th European Signal Processing Conference*, pp. 2101–2105, Aug 2016.

[20] S. López Tapia, R. Molina, and N. Pérez de la Blanca, "Detection and localization of concealed objects in passive millimeter images," *Engineering Applications of Artificial Intelligence*, vol. 67, pp. 81–90, 2018.

[21] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25* (F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, eds.), pp. 1097–1105, Curran Associates, Inc., 2012.

[22] C.-Y. Lee, P. W. Gallagher, and Z. Tu, "Generalizing pooling functions in convolutional neural networks: Mixed, gated, and tree," in *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, vol. 51, pp. 464–472, 2016.

[23] D. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)," *CoRR*, vol. abs/1511.07289, 2015.

[24] L. Wan, M. Zeiler, S. Zhang, Y. L. Cun, and R. Fergus, "Regularization of neural networks using dropconnect," in *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, vol. 28, pp. 1058–1066, JMLR Workshop and Conference Proceedings, May 2013.

[25] B. Graham, "Fractional max-pooling," *CoRR*, vol. abs/1412.6071, 2014.

[26] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 640–651, 2017.

[27] F. Liu, G. Lin, and C. Shen, "CRF learning with CNN features for image segmentation," *Pattern Recognition*, vol. 48, pp. 2983–2992, Oct. 2015.

[28] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, pp. 2278–2324, Nov 1998.

[29] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, pp. 2278–2324, November 1998.

[30] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*,

pp. 807–814, Omnipress, 2010.

[31] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)* (D. Blei and F. Bach, eds.), pp. 448–456, JMLR Workshop and Conference Proceedings, 2015.

[32] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, pp. 1929–1958, 2014.

[33] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. A. Riedmiller, "Striving for simplicity: The all convolutional net," *CoRR*, vol. abs/1412.6806, 2014.

[34] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, June 2016.

[35] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1026–1034, 2015.

[36] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *arXiv preprint arXiv:1511.00561*, 2015.

[37] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[38] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Computer Vision and Pattern Recognition (CVPR)*, 2015.

[39] T. Schaul, S. Zhang, and Y. LeCun, "No more pesky learning rates," in *Proceedings of the 30th International Conference on Machine Learning*, vol. 28, pp. 343–351, 2013.

[40] A. Coates, H. Lee, and A. Ng, "An analysis of single-layer networks in unsupervised feature learning," in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, vol. 15 of *JMLR Workshop and Conference Proceedings*, pp. 215–223, JMLR W&CP, 2011.

[41] T. Dozat, "Incorporating Nesterov Momentum into Adam," tech. rep., Stanford University, 2015.

[42] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the 22Nd ACM International Conference on Multimedia*, pp. 675–678, 2014.

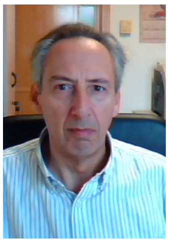
[43] F. Chollet, "Keras." https://github.com/fchollet/keras, 2015.

[44] Theano Development Team, "Theano: A Python framework for fast computation of mathematical expressions," *arXiv e-prints*, vol. abs/1605.02688, May 2016.

Santiago López-Tapia received the degree in Computer Science and the Master degree Computer Science in 2014 and 2015, respectively. Currently, he is a Ph.D. student of the Visual Information Processing group, at the Department of Computer Science and Artificial Intelligence of the University of Granada.

Rafael Molina received the degree in mathematics (statistics) and the Ph.D. degree in optimal design in linear models from the University of Granada, Granada, Spain, in 1979 and 1983, respectively. He became Professor of Computer Science and Artificial Intelligence at the University of Granada, Granada, Spain, in 2000. He is the former Dean of the Computer Engineering School at the University of Granada (1992–2002) and Head of the Computer Science and Artificial Intelligence department of the University of Granada (2005–2007). His research interest focuses mainly on using Bayesian modeling and inference in problems like image restoration (applications to astronomy and medicine), superresolution of images and video, blind deconvolution, computational photography, source recovery in medicine, compressive sensing, low-rank matrix decomposition, active learning, fusion, and classification.

Prof. Molina serves as an Associate Editor of Applied Signal Processing (2005–2007); the IEEE Transactions on Image Processing (2010–2014); and Progress in Artificial Intelligence (2011–present); and an Area Editor of Digital Signal Processing (2011–present). He is the recipient of an IEEE International Conference on Image Processing Paper Award (2007) and an ISPA Best Paper Award (2009). He is a coauthor of a paper awarded the runner-up prize at Reception for early-stage researchers at the House of Commons.



Nicolás Pérez de la Blanca received the M.S. degree in mathemathics from the University of Granada in 1975 and the Ph.D. degree from the same University in 1979. His research areas have involved, data analysis, biomedical image processing, pattern recognition and image classification. He has co-authored more than eighty research papers in these areas. Now is Full Professor at the Department of Computer Science and Artificial Intelligence of the University of Granada and is the head of the Visual Information Processing Group at this University. He has been president of the Spanish chapter of the IAPR (AERFAI) from 2000 to 2008. His current research interest areas are image and object classification, unsupervised learning and deep learning.

Chapter 5

Mitosis Detection in Whole-Slide Images

5.1 A Fast Pyramidal Bayesian Model for Mitosis Detection in Whole-Slide Images

5.1.1 Publication details

Authors: Santiago López-Tapia, Jose Aneiros-Fernández, and Nicolás Pérez de la Blanca.

Title: A Fast Pyramidal Bayesian Model for Mitosis Detection in Whole-Slide Images.

Publication: 15th European Congress on Digital Pathology, vol. 11435, 135-143, Coventry (UK), April 2019.

Status: Published.

5.1.2 Main Contributions

- In this work, we introduce a fast DL based pyramidal mitosis detection algorithm for WSIs. Our model propagates information between scales using the uncertainty calculated by a cascade of Bayesian CNN [107] models. Moreover, this uncertainty is also used to prune out false positives.

- We increase the performance of the system further by introducing geometric invariant features using Spatial Transforming Layers [108].

- We validate our proposal's contributions using a new database of 22 WSIs of melanoma skin cancer tissue with 8236 mitoses.

- Furthermore, we test whether our model can be used for knowledge transfer between tissues. The model is tested and compared to the state-of-art in mitosis detection in breast cancer tissue.

A fast pyramidal Bayesian model for mitosis detection in whole-slide images

Santiago López-Tapia *, José Aneiros-Fernández †, Nicolás Pérez de la Blanca*
*Dept. of Computer Science and Artificial Intelligence, University of Granada, Granada, Spain
Email: {sltapia, nicolas}@decsai.ugr.es
†Intercenter Unit of Pathological Anatomy, San Cecilio University Hospital, Granada,Spain
Email: janeirosf@hotmail.com

Abstract

Mitosis detection in Hematoxylin and Eosin images and its quantification for mm^2 is currently one of the most valuable prognostic indicators for some types of cancer and specifically for the breast cancer. In whole-slide images the main goal is to detect its presence on the full image. This paper makes several contributions to the mitosis detection task in whole-slide in order to improve the current state of the art and efficiency. A new coarse to fine pyramidal model to detect mitosis is proposed. On each pyramid level a Bayesian convolutional neural network is trained to compute class prediction and uncertainty on each pixel. This information is propagated top-down on the pyramid as a constraining mechanism from the above layers. To cope with local tissue and cell shape deformations geometric invariance is also introduced as a part of the model. The model achieves an F1-score of 82.6% on the MITOS ICPR-2012 test dataset when trained with samples from skin tissue. This is competitive with the current state of the art. In average a whole-slide is analyzed in less than 20 seconds. A new dataset of 8236 mitoses from skin tissue has been created to train our models.

Index Terms

Mitosis detection, Pyramid, Bayesian model, Multiscale processing

I. Introduction

The quantification of mitotic cells in Hematoxylin and Eosin (H&E) images and more specifically its density per square millimeter is one of the current most stronger markers in cancer prognosis.

The advent of the high-resolution scanner technology to the computational pathology field has allowed to obtain digital whole-slides images (WSI). Nevertheless, the huge size of the images and the computing time of the current detection algorithms impose in practice a partial rather than a fully image detection and counting.

This paper has been supported by the Spanish Ministry of Economy and Competitiveness and the European Regional Development Fund (FEDER) under the grant DPI2016-77869-C2-2-R. Santiago López-Tapia received financial support through the Spanish FPU program.

Several difficulties can be identified as responsible of the current low detection rate on H&E stained images. On one hand, the variability in RGB color map due to different stain intensities and scanners technology [1], [2]. On the other hand, the presence of very hard false positives due to Hematoxylin staining of non-cells tissue also makes harder the detection process. In addition, the mitosis undergoes four different stages with different shapes and appearances. This geometric variability and the low number of mitosis pixels per WSI also represent a new source of false positive. These difficulties all together make the design of an efficient and accurate mitosis detection algorithms a challenge task [3].

Different Challenges such as TUPAC-2016[4], MITOS-ATYPIA [5] and MITOS ICPR-2012 [6] have been organized in the last years to foster the detection algorithms. But the contributed datasets from them all are too small and only from breast cancer tissue. Currently, there are no other larger open access mitosis datasets. We have created a mitosis dataset from skin cancer images to train our model. In this type of cancer mitosis detection is also a very relevant prognostic indicator[7]. In order to compare our model with other results in the literature we have tested with MITOS ICPR-2012.

II. Related works

Many contributions to the use of CNN model for mitosis detection have been proposed since the ICPR-2012 challenge MITOS ICPR-2012 [6] was available [8], [9], [10]. The best result from all these approaches is an F1 score of 78.8%. In [11] an adaptation to the general object detection framework from CNN, Faster R-CNN, is proposed. They focus on the use of very deep architectures for mitosis detection achieving an F1-score of 83.2% in MITOS ICPR-2012. More recently in [12] a new way of approaching the detection task is proposed. They stain twice each slide using Phospho-histone H3 (PHH3) and H&E and leverage on the complementary properties of these stains to improve the detection. They succeed in removing many of the false positives but at the cost of a very complex processing. Our method addresses a similar goal but from a pyramidal approach. All mentioned approaches exploit the depth increment in the architectures as the main mechanism to generate good features. In [13] an approach inspired in Wide Residual networks (WRN) [14] focus on the wide of the layer, instead of the number of layers. This fact simplifies the architecture making it more efficient at test time and easier of training. They reached an F1 score of 64.8% in the challenge TUPAC-2016 [4], which is a result competitive with the state of the art for this dataset. Our architectures are inspired by this network.

The feature extraction stage of all above approaches either use the 40x scale or use a fine to coarse feature pyramid starting in 40x. In both cases the highest resolution scale is the input information. In contrast, here we propose a coarse to fine approach in a top-down pass through a pyramid representing three scales of the image. We find benefits in both efficiency and accuracy. The standard CNN models lack uncertainty measurements about the predictions as well as specific layers to obtain invariance to geometric deformations. The use of a Bayesian approach to CNN allows us to compute uncertainty in a natural way. On each internal pyramid level, prediction and uncertainty from the above levels are used as input to improve the final model prediction. We find that information from lower resolutions allow us to constraint the optimization process at the highest resolution. In addition, and to cope with both the cell shape variability induced by the phases of the mitosis and the tissue local deformations, our model incorporate specific layers to compute geometric invariant features [15].

In summary our contributions are: a) A new and fast pyramidal mitosis detection algorithm for WSI achieving a F1-score competitive with the state of the art on MITOS ICPR-2012 dataset; b) A new information propagation mechanism between scales from a cascade of Bayesian CNN model; c) The use of uncertainty and geometric invariance to improve the detection score; d) A model able of learning knowledge transfer between tissues; f) Mitosis detection time on WSI faster than ever before.

The rest of the paper is as follows. Section.III we describes the model. Section.IV describes the training and test stages. Section.V shows the experiment and Section.VI show the discussion and conclusions.

III. MODEL DESCRIPTION

Our model is defined as a forward cascade of classifiers applied on a course-to-fine image pyramid build from a WSI at three magnification scales 10x, 20x and 40x. We assume 40x represents the sample image and the lowest pyramid level. Fig.1 shows a diagram of the architecture. On each pyramid level a Bayesian CNN classifier inspired in the design of a Wide Residual Network [14] is trained. The three classifiers in the cascade output a mask of detected mitosis, a feature map, and the uncertainty per feature in terms of standard deviation as shown in Fig.1.

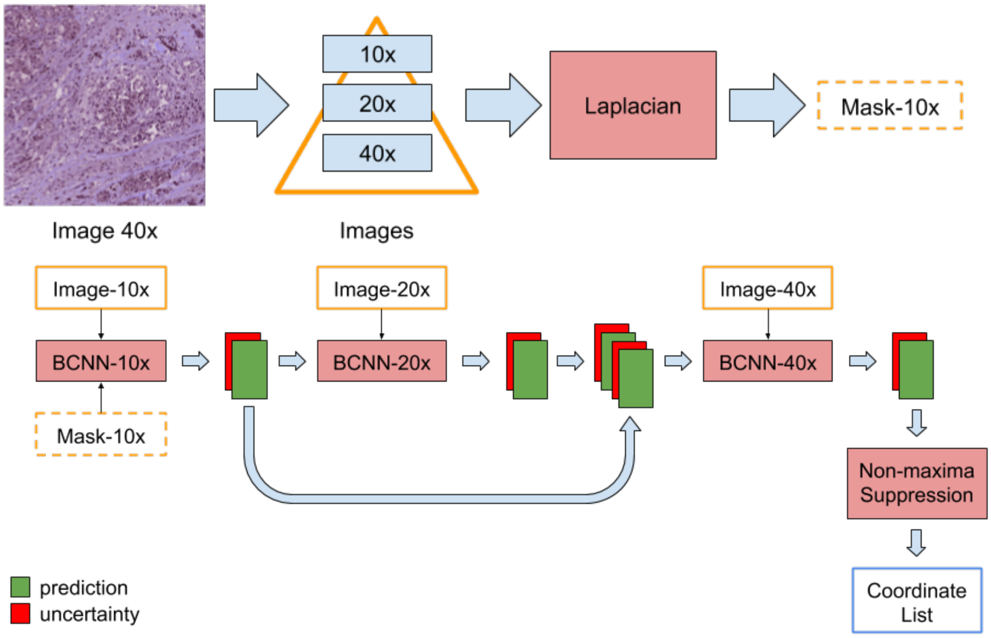


Fig. 1: Diagram showing the pyramidal model and the cascade of classifier to process the pyramid. The top of the figure shows, in this order, the input image, the pyramid building and the computation of the initial mask at 10x. The bottom shows how the first two pyramid levels provide input information to the third pyramid level. The result is the output of a non-maxima suppression process. See details in the text.

These feature maps are used by the next detectors, top-down, as soft constraints to focus the training on the most difficult negative samples (see Fig.1). We call this model PB-CNN. Furthermore, to make the model resistant to local and shape deformations, appearing by both the process of collecting and staining tissue and cells shape deformation, a Spatial Transforming Layer[15] is applied before the residual blocks 4th and 7th in scale x40 (see Fig.2). We call this model PB-CNN-STP. On the output of last classifier, we put to zero the predictions of those pixels which uncertainty is higher than a threshold fixed in training. The experiments show that these higher values are usually associated to WSI artifacts of low frequency in the training dataset. Finally, a non-maximum suppression step is carried out to keep, in cluster of overlapping regions, only the one with the highest probability. Our final output is a list of coordinates joint to their corresponding probability and standard deviation. As it can be seen in Fig.2, we use a late fusion criteria incorporating feature maps and uncertainty, of the above levels, at the end of network. We have found in our experiments that this late fusion of features provides better results that doing it earlier.

The architecture of our detector is shown in Fig.2. The architecture is a Wide Residual Network [14] that uses three Wide Residual Units (WRU) (see right block). To reduce the spatial size of processed patch, we use a stride of 2 at certain layers (indicated by ”/2” in the figure), in the case of the WRU block, the stride is applied at the first convolutional layer. The same architecture is used for PB-CNN-SPT adding a Spatial Transforming Layer[15] at the scale 40x as indicated previously.

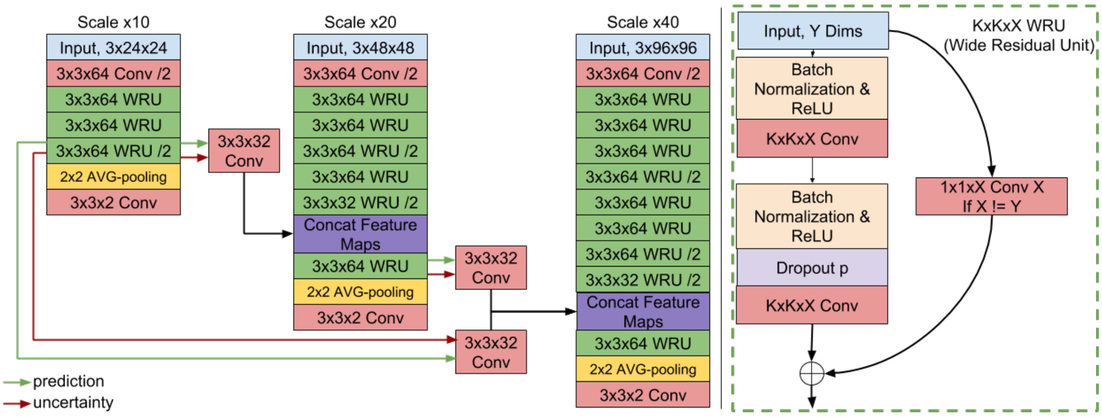


Fig. 2: Network architecture used for PB-CNN (see left figure). $K \times K \times N$ Conv indicates a convolution layer with $K \times K$ kernel and N filters. $/2$ idicates that the convolutution layer uses a stride of 2. 2×2 AVG-pooling indicates an average pooling layer of kernel 2×2.

IV. Training and Test

A. Dataset

The dataset is created from 22 WSI of melanoma skin cancer. The images were acquired using a scanner Philips with a resolution of 0.25 micron per pixel. A senior pathologist of the Unit of Computational Pathology of the University Hospital San Cecilio in Granada labeled the WSI at 40x by indicating the center of the mitosis. 8236 were annotated mitosis.

B. Input mask at 10x

To detect initial relevant regions at 10x we apply a Laplacian of the Gaussian (LoG) filter with $\sigma = 9$ over the Hematoxylin band obtained by color deconvolution [2]. Then a thresholding for negative values that are less than -0.28 is applied. We select the windows centered at each connected component as possible candidates to contain a mitosis. A window of size 24×24 pixels is used.

C. Learning and testing

Let's denote by GTL the pyramid ground-truth labeling defined by the coordinates of mitosis centers in all scales. A strong labeling pyramid (MGTL) is generated, by labeling as 1 those windows inside circles of fixed radius centered at the GTL's mitosis centers. Radius of 96, 48 and 24 pixels are used for 40x, 20x and 10x scale respectively. The windows of label 0 on each level are computed in runtime as the difference between the MGTL mask and the mask obtained by thresholding and extrapolating the predicted probabilities from the above pyramid level, let denote it as PR. The used threshold is fixed in training time in order to keep all GTL windows within the class 1. The label 0 at each level represents hard false positive, since not being mitosis were predicted as such with high probability by the above level. In the case of the first level (10x), the mask computed in section IV-B is used as PR. At each pyramid level the negative training samples are obtained by sampling of the mask of class 0. The positive samples are patches centered at the coordinates indicated in GTL. The patch size used is 24×24, 48×48 and 96×96 for scales 10x, 20x and 40x, respectively. Before extracting the patches for training, we use the stain normalization algorithm proposed in [1] to reduce the variation in the training dataset. This normalization is also used during testing before the WSI is processed.

Each classifier in the pyramid is trained for 90 epochs with the Adam optimizer [16] to minimized the binary cross-entropy loss: $\mathrm{BCE}(y, p) = y \log(p) + (1 - y) \log(1 - p)$ where y is the label and p the predicted probability of the sample being mitosis. The probability value of the dropout used in all the models was 0.4 and the weight decay was set to 10^{-4}. The learning rate was set to 10^{-3} for PB-CNN and $5 \cdot 10^{-4}$ for the scale 40x of PB-CNN-STP; in both cases was divided by 10 each 30 epochs. Each batch was constructed by randomly sampling 32 positive samples and 32 negative samples. Each epoch samples 10^6 batches.

Data augmentation has been applied from random rotations and mirroring. We also apply random shifting up to 4, 8 and 16 pixels for scales 10x, 20x and 40x respectively, as well as random scaling by a factor sampled in the range [0.75, 1.25]. Additive Gaussian noise with 0.05 of standard deviation was also added to the input. Finally, in order to introduce robustness to color variation, we use the stain augmentation process proposed in [11] with α and β parameters sampled in the ranges [0.995, 1.05] and [-0.05, 0.05] respectively.

We implement the Bayesian approach according to [17]. For it, we sample the dropout units from a Bernoulli distribution with probability $p = 0.4$. Once trained, the prediction and uncertainty of the network per each input image are computed as the average of the values of 10 new samples of the dropout units after weight adaptation by the forward pass.

Finally, we have found necessary to apply a high Dropout rate to the feature maps of previous levels at the beginning of the training process. This was done in order to force not to rely too

TABLE I: Two first rows show a comparison with state of the art methods on ICPR-2012 MITOSIS test set [6]. Last three rows show a comparison on our test dataset of 5 WSI. Evolution of the F1-score and processing time are shown by scales. Results for the times were calculated applying sliding window on each pixel and using a Nvidia Titan X.

Method	**PBCNN-STP**	**DeepDet[11]**	**RR[18]**	**CasNN[19]**	
F1 score	82.6%	83.2%	82.3%	78.8%	
Method	**PBCNN10x**	**PBCNN20x**	**PBCNN**	**PBCNN-STP**	**WRCNN40x**
F1-score	62.8%	72.5%	78.1%	81.3%	71.2%
Ave.Time WSI	27 ± 11	28 ± 10	29 ± 11	31 ± 11	56 ± 23

much in previous predictions and extract useful information from the current scale. We set this dropout rate to 0.8 and linearly decrease it to 0 at epoch 40.

V. EXPERIMENTS

To demonstrate the benefits of our proposed PB-CNN, we first test it on our dataset conformed by 22 WSIs. We separate the WSIs in training and test sets by randomly selecting 5 WSIs as the test set and leaving the remaining 17 ones for training. We have 7133 mitoses for training and 1103 for testing. At the second row of Table I we show the F1-score and time increase of adding each level of the pyramid, as well as using the Spatial Transforming Layer[15]. All models were tested using the same framework and the same computer with a Nvidia Titan X with 12GB of RAM. As the table shows, each level comes with a significant increase in performance at the cost of a small increase in computational time. Adding the Spatial Transforming Layer [15] we get an increase of 3.2% in F1-score at the cost of a slightly impact on the processing time. For the sake of comparison, we train a Bayesian Wide Residual Network identical to the one used on the 40x scale only, we call it WR-CNN-40x. The training process was the same as described for our PB-CNN in Section. IV-C, although we change the dropout probability to 0.3 since we find it gives better results. The results of this WR-CNN-40x are show in the two first rows of the Table I. The propose PB-CNN is almost two times faster and gets a significant better F1-score than this WR-CNN-40x, showing that the increase obtained is due to the pyramid architecture.

In order to compare our models with other in the literature, we train our best performing model PB-CNN-STP with our 22 WSI and test it on MITOS-ICPR2012 test set containing images produced by the Aperio XT scanner. Then, we extract the features provide by each scale before the last classification layer and train a Random Forest classifier on the training dataset of the Aperio XT scanner. Table I shows the results in comparison with other state of the art methods. We can see that the best of our proposed method get a competitive result against current state of the art in F1-score, despite being trained on WSI of a different tissue.

VI. DISCUSSION AND CONCLUSIONS

A new coarse to fine cascade of CNN Bayesian models for mitosis detection has been proposed. The new mechanism of information propagation from top to bottom, using the uncertainty of the prediction, allow to get results competitive with the state of the art on MITOS ICPR-2012 dataset. To the best of our knowledge, this is the first time that a coarse to fine approach combined with uncertainty is used in mitosis detection. In our experiments, the Bayesian pyramid approach reduces the computation time by a factor of two and increases by 7% the F1-score with respect to the same CNN architecture applied only over the 40x scale. We have also shown the benefits of using Spatial Transforming Layers to deal with local geometric deformations. On our dataset this invariance increases the F1-score score by a 3.2%. It is also remarkable that our architecture is trained with samples from a different tissue than breast cancer. This shows that our model is able of learning useful mitosis features for the transfer of learning between tissues. Regarding efficiency, the times measured on whole WSI make our method a good candidate for daily clinic. More experiments on harder databases have to be carried out in order to assess the good properties pointed out for the model. The addition of new input information from inmunohistochemistry stains is also other relevant issue for future work.

REFERENCES

[1] E. Reinhard, M. Ashikhmin, B. Gooch, and P. Shirley, "Color transfer between images," *IEEE Computer Graphics and Applications*, vol. 21, pp. 34–41, July 2001.

[2] M. Macenko, M. Niethammer, J. S. Marron, D. Borland, J. T. Woosley, X. Guan, C. Schmitt, and N. E. Thomas, "A method for normalizing histology slides for quantitative analysis," in *IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pp. 1107—-1110, 2009.

[3] M. Veta, P. J. van Diest, and and col., "Assessment of algorithms for mitosis detection in breast cancer histopathology images," *Medical Image Analysis*, vol. 20, no. 1, pp. 237 – 248, 2015.

[4] MICCAI, "http://tupac.tue-image.nl/," 2016.

[5] ICPR, "https://mitos-atypia-14.grand-challenge.org/," 2014.

[6] L. Roux, D. Racoceanu, N. Lomenie, M. Kulikova, H. Irshad, J. Klossa, F. Capron, C. Genestie, G. L. Naour, and M. N. Gurcan, "Mitosis detection in breast cancer histological images an icpr 2012 contest," *J Pathol Inform*, 2013.

[7] A. Tejera-Vaquerizo, G. Pérez-Cabello, and and col., "Is mitotic rate still useful in the management of patients with thin melanoma?," *Journal of the European Academy of Dermatology and Venereology*, vol. 31, no. 12, pp. 2025–2029, 2017.

[8] D. C. Cireşan, A. Giusti, L. M. Gambardella, and J. Schmidhuber, "Mitosis detection in breast cancer histology images with deep neural networks," in *International Conference on Medical Image Computing and Computer-assisted Intervention*, pp. 411–418, Springer, 2013.

[9] H. Wang, A. Cruz-Roa, A. Basavanhally, H. Gilmore, N. Shih, M. Feldman, J. Tomaszewski, F. A. González, and A. Madabhushi, "Mitosis detection in breast cancer pathology images by combining handcrafted and convolutional neural network features," *Jour. Medical Imaging*, vol. 1, 2014.

[10] H. Chen, X. Wang, and P. A. Heng, "Automated mitosis detection with deep regression networks," in *IEEE Int Symp Biomedical Imaging.*, pp. 1204—-1207, 2016.

[11] C. Li, X. Wanga, W. Liua, and L. J. Latecki, "Deepmitosis: Mitosis detection via deep detection, verification and segmentation networks," *Medical Image Analysis*, pp. 121–133, 2018.

[12] D. Tellez, M. Balkenhol, and and col., "Whole-slide mitosis detection in "h&e" breast histology using phh3 as a reference to train distilled stain-invariant convolutional networks," *IEEE Transactions on Medical Imaging*, vol. 37, pp. 2126–2136, Sept 2018.

[13] E. Zerhouni, D. Lányi, M. Viana, and M. Gabrani, "Wide residual networks for mitosis detection," in *IEEE 14th International Symposium on Biomedical Imaging (ISBI)*, pp. 924–928, 2017.

[14] S. Zagoruyko and N. Komodakis, "Wide residual networks," in *Proceedings of the British Machine Vision Conference (BMVC)*, pp. 87.1–87.12, September 2016.

[15] M. Jaderberg, K. Simonyan, A. Zisserman, and k. kavukcuoglu, "Spatial transformer networks," in *Advances in Neural Information Processing Systems 28*, pp. 2017–2025, 2015.

[16] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014.

[17] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning," in *Proceedings of the 33rd International Conference on Machine Learning (ICML-16)*, 2016.

[18] A. Paul, A. Dey, and and col., “Regenerative random forest with automatic feature selection to detect mitosis in histopathological breast cancer images,” in *MICCAI 2015*, pp. 94–102, Springer, 2015.

[19] H. Chen, Q. Dou, X. Wang, J. Qin, and P. A. Heng, “Mitosis detection in breast cancer histology images via deep cascaded networks,” in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16)*, pp. 1160–1166, 2016.

5.2 Improvement of Mitosis Detection Through the Combination of PHH3 and HE Features

5.2.1 Publication details

Authors: Santiago López-Tapia, Cristobal Olivencia, Jose Aneiros-Fernández, and Nicolás Pérez de la Blanca.
Title: Improvement of Mitosis Detection Through the Combination of PHH3 and HE Features.
Publication: 15th European Congress on Digital Pathology, vol. 11435, 144-152, Coventry (UK), April 2019.
Status: Published.

5.2.2 Main Contributions

- A new technique for mitosis detection in WSIs is proposed. This new method uses a CNN fed with images of H&E and PHH3 stained tissue. To find the relevant matches between both stains, a fast and accurate technique is proposed. It consists of an initial registration using Surf[135] points and a Siamese CNN [136] that validates the initial matches.

- Experiments are carried out using a new database of 65 WSIs (47 of skin cancer and 18 of breast cancer).

Improvement of mitosis detection through the combination of PHH3 and HE features

Santiago López-Tapia *, Cristobal Olivencia*, José Aneiros-Fernández †, Nicolás Pérez de la Blanca*

*Dept. of Computer Science and Artificial Intelligence, University of Granada, Granada, Spain

Email: sltapia@decsai.ugr.es, cristoly94@gmail.com, nicolas@ugr.es

†Intercenter Unit of Pathological Anatomy, San Cecilio University Hospital, Granada,Spain

Email: janeirosf@hotmail.com

Abstract

Mitosis detection in hematoxylin and eosin (H&E) images is prone to error due to the unspecificity of the stain for this purpose. Alternatively, the inmunohistochemistry phospho-histone H3 (PHH3) stain has improved the task with a significant reduction of the false negatives. These facts point out on the interest in combining features from both stains to improve mitosis detection. Here we propose an algorithm that, taking as input a pair of whole-slides images(WSI) scanned from the same slide and stained with H&E and PHH3 respectively, find the matching between the stains of the same object. This allows to use both stains in the detection stage. Linear filtering in combination with local search based on a kd-tree structure is used to find potential matches between objects. A Siamese convolutional neural network (SCNN) is trained to detect the correct matches and a CNN model is trained for mitosis detection from matches. At the best of our knowledge, this is the first time that mitosis detection in WSI is assessed combining two stains. The experiments show a strong improvement of the detection F1-score when H&E and PHH3 are used jointly compared to the single stain F1-scores.

Index Terms

Mitosis detection, WSI, PHH3 and HE, Siamese CNN

I. Introduction

The quantification of mitosis in histopathological tissues and specifically its ratio per square millimeter is one of the most relevant factors in the prognosis of cancer. Unfortunately, the process of mitosis detection on images stained with standard hematoxylin and eosin (H&E) is difficult and prone to errors due to multiple factors consequence of its unspecificity [1]. H&E staining only helps indirectly to mitosis identification, being the hyperchromaticity induced on the mitotic cell

This paper has been supported by the Spanish Ministry of Economy and Competitiveness and the European Regional Development Fund (FEDER) under the grant DPI2016-77869-C2-2-R. Santiago López-Tapia received financial support through the Spanish FPU program.

nucleus one of the its most salient features. Unfortunately, many others tissue parts are stained with a similar color too.

Phospho-histone H3 (PHH3) is a well-known immunomarker, specific for cells undergoing mitoses [2]. This fact causes PHH3 to improve the inter-observer variability of the mitosis count by a decrease in false negatives, but at the same time is prone to false positives as for instance in inter-phase tumor cells with phosphorylated core protein H3. The staining with PHH3 has meant an important improvement in mitosis detection for many type of cancers [3], [4].

The technology for the whole scanning of tissue slides (WSI) is able of digitizing a slide at resolutions of $0.25 - 0.16$ microns per pixel, which means image sizes of 10^{10} pixels. In this setting, the task of mitosis detection can only be addressed using accurate and efficient algorithms. The convolutional neuronal network (CNN) models have demonstrated, in recent years, a clear superiority over traditional approaches in this task. [5], [6]. Here we focus on these kind of models.

An issue that remains to be explored in some detail is the relevance of the combination of stains in the mitosis detection process. Recently, in [7] an interesting approach taking advantage of the properties of both stains, H&E and PHH3, to build a mitosis detector on H&E has been proposed. This approach uses the PHH3 information to locate ground-truth mitosis on WSI but the goal is a classifier on H&E. Although the approach means an important step in the detection of mitosis in WSI, several issues still remain open. First, to design a simple training model taking advantage of both stains simultaneously. Second, the labeling process should take into account both stains. Fig.1 shows some cases of mitosis where the labeling from a single stain is misleading. Finally, assessing the contribution of trained detectors with both stains is a relevant issue to improve routine in daily practice.

In contrast to the above discussed approach, here we propose the simultaneous use of both stains in the labeling and detection stages. To do that we stain twice each slide taking advantage of the property of the antigenic recovering of the immunochemistry for destaining the H&E. This strategy reports important benefits: (i) better labeling, (ii) training dataset with both stains, (iii) improvement in detection score. The two most important challenges in our approach are a fast search for potential correct matches and an assessment model for matches.

Our main contributions in this paper are: (i) a fast and efficient technique to generate matching between both stains of the same object, (ii) the proposal of a SCNN model to validate the matches; (iii) we show that training from both stains means a clear improvement in detection score compared to use of only one. Finally, we emphasize that our searching algorithm makes very easy the labeling of pairs.

The rest of the paper is as follows. Section.II defines the problem. Section.III discuss the proposed approach. Section.IV shows the experimental results, and in Section.V the discussion and conclusions are presented.

II. Problem definition

To begin with we focus on the automatic object matching between stains of the same histological tissue. The relevance of this task is due to the lack of consensus between pathologists when they are asked to label a set of cells as mitosis or no-mitosis in H&E images. In the MYTHOS-ATYPIA

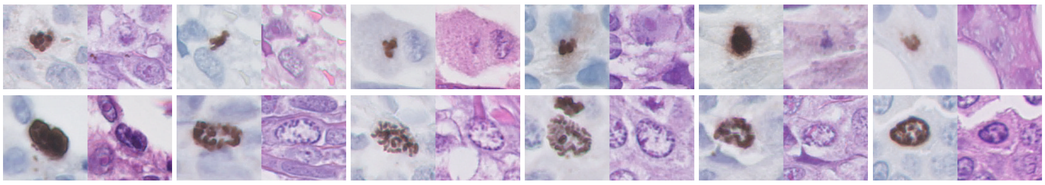


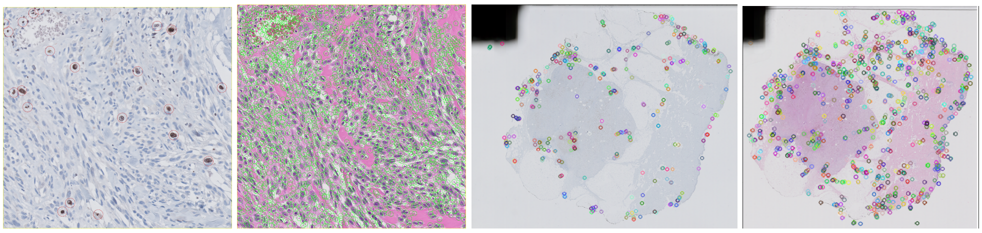
Fig. 1: This figure shows by rows examples of two difficult scenarios regarding mitosis detection in H&E or PHH3. The first row shows examples where the mitoses are very difficult to detect in H&E but can be easily detected on PHH3. The second row shows examples where the PHH3 stain indicates positive mitosis but the H&E stain shows that it is not.

challenge[8], for instance, multi-labels had to be considered. Daily practice has shown that many ambiguities can be solve when both stains are observed together. Fig.1 shows some examples. This has motivated the interest to know how much a detector can improve when training with both stains. The automatic identification of correct matches between stains it is not a straightforward task. The manipulation of the slide in the double staining process, that is, staining with H&E and scanning, destaining, and restaining again with PHH3 and new scanning, introduce small local deformations on the tissue that makes impossible automatic matching of the images using geometrical registering. In addition, the different response of the tissue to each one of the stains also introduce strong differences in the shape and color of the surfaces of the cells as shows Fig.1. To overcome all these deformations, we propose a search strategy to extract possible matches and a similarity distance to find correct matches. For this latter task, we propose a Siamese CNN (SCNN) [9], [10] since the CNN models have shown to be very efficient in extracting similar features from images, that being visually different, are similar in a some semantic context. At one last step, the correct matches are assessed, for mitosis presence, by a CNN classifier.

III. Methodology

A. Matches extraction

Let's denote by p-WSI=$(\mathrm{I}_{\mathrm{PHH3}}, \mathrm{I}_{\mathrm{HE}})$ the two WSI images of the same slide with different stain. We extract the objects present in each image applying standard cell detection functions, [11], and eliminating all those objects with a size greater than a preset threshold. For this, we use the hematoxiline and DAB bands of the H&E and PHH3 images respectively. The center of mass of the remaining connected components (CC) is computed. A kd-tree data structure (KdT) [12] is fed with the coordinates of the centers of the H&E image. The centers of the PHH3 image are saved as a list of points, L_{DAB}. In order to reduce the number of pair to analyze we take advantage of the specificity of the PHH3 stain to identify the potential mitosis presents in the image. To this end, each vector of coordinates in the DAB list is used as query to the KdT to retrieve matching candidates from the H&E image. Fig.2(a,b)) shows an example of how unbalanced is the number of detections in both stains. In order to make easier the searching process we register the bounding boxes of the tissue area in both images through an affine transformation, $\mathcal{A} : \mathrm{I}_{\mathrm{PHH3}} \to \mathrm{I}_{\mathrm{HE}}$, estimate from SURF points [12] detected from grey levels after sub-sampling the image by a factor of ten. For each point $x \in L_{DAB}$, its coordinates are projected onto the axes of H&E by the affine transformation, $y = \mathcal{A}x$, and all points $z \in \mathrm{KdT}$



(a) (b) (c) (d)

Fig. 2: Images (a) and (b) show, inside circles, objects detected on PHH3 and H&E respectively. Images (c) and (d) show, in circles of color, SURF points detected in PHH3 and H&E respectively. (Best see it at higher magnification)

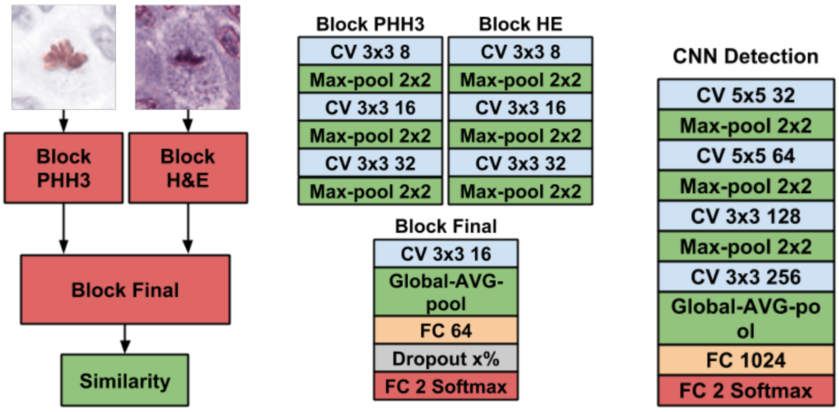
such that $distance(y, z) < thr$ are extracted, where thr is a prefixed threshold. Let's denote by p-center the pair formed by the coordinates of the query-point, x, and the coordinates of anyone of its matches. For each p-center, image-patches of size 80×80 pixels centered on them are extracted from the images. Let's denote them as p-patch. These p-patch are assessed by the SCNN that output a similarity distance in terms of a probability. For each x the p-patch with maximun probability is considered the true match. Let's denote a correct p-patch as p-match. In summary, our matching algorithms is as follows:

```
ALGORITHM: MS($H\&E, PHH3, T, \mathcal{P}_{\text{HE}}, \mathcal{P}_{\text{PHH3}}$)
Input:
- ($H\&E, PHH3$): WSI of the same slide
- $T$: distance-threshold for searching
- $\mathcal{P}_{\text{HE}}$: list of coordinates of the object centers detected in H&E
- $\mathcal{P}_{\text{PHH3}}$: list of coordinates of the object centers detected in in PHH3
Preprocessing:
- Build a KdT from $\mathcal{P}_{\text{HE}}$.
- Compute SURF points: $\text{SURF}_{\text{HE}}$, $\text{SURF}_{\text{PHH3}}$
- Compute Global affine transformation: $\mathcal{A} : \text{SURF}_{\text{PHH3}} \rightarrow \text{SURF}_{\text{HE}}$. .
Correspondences:
For each item $p \in \mathcal{P}_{\text{PHH3}}$
1.- Compute $\hat{p} = \mathcal{A}p$
2.- Compute $\mathcal{P}_{\text{KdT}}(p) = \{q | q \in \text{KdT}, \text{distance}(\hat{p}, q) < T\}$
3.- Extract patches $\{o_q\}$ centered in $q \in \mathcal{P}_{\text{KdT}}(p)$
4.- Compute $\hat{q} = argmax_{q \in \mathcal{P}_{\text{KdT}}(p)} Similarity_{SCCN}(o_p, o_q)$
5.- Output $(o_{\hat{q}}, o_p)$
```

where $Similarity_{SCCN}$ denote the probability computed by the Siamese network.

B. Dataset and Labeling

Two datasets of p-match have been created. The first dataset, DS1, is defined by 57k (1k=1000) p-match extracted after staining and scanning 48 slides, 30 of skin cancer (melanoma) and 18 of breast cancer. The second dataset, DS2, is defined by 11k p-match of mitosis and 75k p-patch no



(a) (b) (c)

Fig. 3: Siamese architecture: (a) shows the global network design composed of two parallel branches to process each one of the images. After the feature extraction a function of the feature vectors compute the similarity between the images. In (b) we show the three main blocks that compose the model. CV correspond to Convolution and ReLU activation and FC to full connected layer followed by ReLU. We use batch normalization before each ReLU. In (c) the architecture of the CNN model used for mitosis detection is shown.

mitosis extracted from 17 slides of melanoma. The slides were scanned with a Philips Ultra-Fast Scanner at a spatial resolution of 0.25 microns per pixel. All p-patch were labeled by a senior pathologist of the Saint Cecilio Universitary Hospital in Granada, who annotated a percentage of the correct matches on each p-WSI. An interactive software which iterates showing p-patches and their surrounding areas was used for this task. A p-patch is tagged with a maximum of two clicks: one click to decide correspondence vs. no correspondence and another click to decide mitosis vs. non-mitosis. This is a very simple routine that allows to label many pairs in a short period of time.

C. Training

Our specific SCNN model is shown in Fig.3(a-b). It can be observed that Block-PHH3 y Block-H&E share the same architecture based on a standard Lenet model of CNN [13]. Block-Final processes the features from the input blocks to learn the similarities. The network is trained during 100 epochs using a batch size of 128 with Adam[14] optimizer and initial learning rate of 0.0002. We reduce the learning rate by a factor of 10 each 10 epochs if the training loss has not been reduced. The training stops if the loss keeps without reducing after another 20 epochs. The networks outputs the probability of a p-patch, $\{\text{he}, \text{phh3}\}$, of being a p-match. We train the network to minimize the binary cross entropy loss $\mathcal{L}(\cdot,\cdot)$, defined for each sample as,

$$\mathcal{L}(\text{he}, \text{phh3}) = -y(\log(\text{f}_\theta(\text{he}, \text{phh3}) + (1-y)\log(1-\text{f}_\theta(\text{he}, \text{phh3}))$$

where $y \in \{+1, -1\}$ represents the image-pair's label and f_θ represents the function computed by our SCNN. To regularize the model, we use L2-weight decay of strength 1.0 on the parameters of the network and Dropout[15] with probability of 0.3 before the last full connected layer. The CNN used for mitosis detection from p-match is shown in Fig. 3(c). We minimize the binary

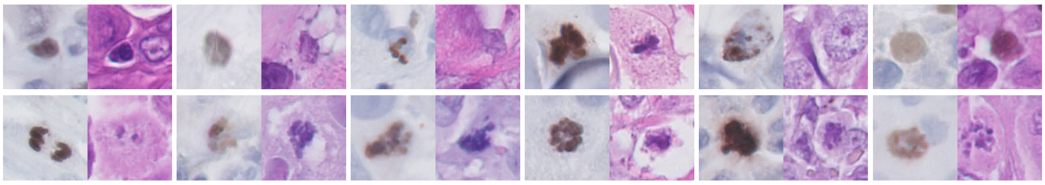


Fig. 4: This Fig. show some examples of the errors of the algorithm-MS. The first row shows examples of false negative p-match. The second row shows examples of false positives p-match. See the pair as (PHH3,HE)

cross-entropy loss function using the Adam[14] optimizer with learning rate set to 0.001 during the first 50 epochs, then reduced to 0.0001 for 25 epochs and finally set to 0.00001 for another 25 epochs. We set the weight decay parameter to 0.0001 and use Dropout of 0.5 before each non-linearity except before the Softmax layer. Also, we use data augmentation on the p-match by rotating the input patches by 90°, 180° and 270° and performing horizontal and vertical flips. We also add Gaussian noise with $\sigma = 0.0001$ to the input.

IV. Experimental results

We assess the performance of our algorithm-MS by cross-validation. To do this, we define five folds from the set of 48 p-WSI. On each fold 43 p-WSI are used for training and 5 for testing. In total we use 25 different p-WSI in testing. On each fold the set of p-match, extracted from each image, is used according to the role of the image in that fold. Table.I shows the number of p-match used in training and testing for each fold. The items for the negative class are generated by random combinations of the p-match items. We generate as many negative item as there are p-match. The test with each fold begins by detecting and extracting the coordinates of the centers of the objects in the p-WSI test. We use cell detection routines of the QuPath[11] free software to extract the center of the object on each p-WSI. The kd-tree structure is build using [12]. From them the set of p-patch is estimated. Eventually, the p-patch are assessed by the SCNN. In this experiment what we measure is the accuracy of the p-match test elements (see Table.I(top)). In order to evaluate the effect of the number of p-match in the testing matching error, we design the folds to cover a broad range of values in testing. A value of thr=60 is used as searching distance in the KdT. The average query time per image is about 3s. Third row in Table.I shows the accuracy achieved on each fold. The estimated accuracy of the algorithm-MS for matches is 99.6%±0.58. Fig..4 shows some examples of p-match errors from SCNN. We assess the H&E+PHH3 improvement versus the single stains, on dataset labeled from both stains, using the detector architecture shown in Fig. 3(c). We select this architecture for two reasons. First, it represents an adaptation of Lenet model which is the most popular CNN used for mitosis detection. Second, our dataset is filtered by the matching algorithm that removes much of the false positives. This makes unnecessary a complex architecture for this task. In a first experiment we train and test our detector using each one of the components, H&E and PHH3, of the p-WSI. In the second experiment we use full p-WSI. In all cases the color of the images was normalized using the algorithm given in [16]. From the dataset, DS2, we constructed 5 partitions of WSIs and used them for cross-validation. Table.I in the bottom shows the detection F1-score achieved

TABLE I: Top: results of the correspondence experiment. 1k=1000. The second row shows the number of corresponding pairs used, in each fold, in training and validation respectively. The third row shows the validation accuracy in each fold for patches of 80×80 pixels. Bottom: detection F1-score using the different stains.

	Fold-1	Fold-2	Fold-3	Fold-4	Fold-5
Train matches	56.6k	52.5k	47k	54k	45.6k
Valid. matches	721	4.8k	10.4k	3.4k	11.7k
Valid.Accuracy (80×80)	98.6%	99.9%	100%	99.9%	99.6%

Patches	H&E	PHH3	H&E+PHH3
Detection F1-score	$73.3 \pm 0.5\%$	$77.6 \pm 0.2\%$	$80.7 \pm 0.4\%$

by our detector using patches from H&E, PHH3 and H&E+PHH3 respectively. The result shows that using together both stains greatly improve the F1 score with respect to only using one. To evaluate the impact of the p-match errors in detection we test our CNN with the same image dataset but computing the p-match using the algorithm-MS. In this case an F1 score of $80.1 \pm 0.4\%$ is achieved, which means a drop of only 0.6 points.

V. Discussion and conclusions

The proposed approach shows that both stains H&E and PHH3 when used together make a significant contribution to the detection of mitosis. In addition, our approach contributes with a new technique for the labeling of mitosis using both stains simultaneously. The size of the datasets makes our results preliminary but also reliable. It remains to be done a full evaluation of the matching errors and the influence of the detector. The help of our algorithm-MS in the complete labeling of p-WSI opens the door to create larger and more challenging training data sets to evaluate new algorithms. This will be one goal for future work.

References

[1] M. Veta, P. J. van Diest, and and col., “Assessment of algorithms for mitosis detection in breast cancer histopathology images,” *Medical Image Analysis*, vol. 20, no. 1, pp. 237 – 248, 2015.

[2] C. Tapia, H. Kutzner, T. Mentzel, S. Savic, D. Baumhoer, and K. Glatz, “Two mitosis-specific antibodies, mpm-2 and phospho-histone h3 (ser28), allow rapid and precise determination of mitotic activity,” *Am J Surg Pathol*, vol. 30(1), pp. 83–9, 2006.

[3] P. S. Nielsen, R. Riber-Hansen, T. O. Jensen, H. Schmidt, and T. Steiniche, “Proliferation indices of phosphohistone h3 and ki67: strong prognostic markers in a consecutive cohort with stage i/ii melanoma,” *Modern Pathology*, vol. 26, p. 404, Nov. 2012.

[4] B. F. Dessauvagie, C. Thomas, C. Robinson, F. A. Frost, J. Harvey, and G. F. Sterrett, “Validation of mitosis counting by automated phosphohistone h3 (phh3) digital image analysis in a breast carcinoma tissue microarray,” *Pathology*, vol. 47, pp. 329–334, June 2015.

[5] G. J. S. Litjens, T. Kooi, and and col., “A survey on deep learning in medical image analysis,” *CoRR*, vol. abs/1702.05747, 2017.

[6] A. Janowczyk and A. Madabhushi, “Deep learning for digital pathology image analysis: A comprehensive tutorial with selected use cases,” *J Pathol Inform*, vol. 7, pp. 29–29, Jan. 2016.

[7] D. Tellez, M. Balkenhol, and and col., “Whole-slide mitosis detection in ”h&e” breast histology using phh3 as a reference to train distilled stain-invariant convolutional networks,” *IEEE Transactions on Medical Imaging*, vol. 37, pp. 2126–2136, Sept 2018.

[8] ICPR, “https://mitos-atypia-14.grand-challenge.org/,” 2014.

[9] S. Chopra, R. Hadsell, and Y. LeCun, “Learning a similarity metric discriminatively with application to face verification,” in *Computer Vision and Pattern Recognition*, vol. 1, pp. 539–546, 2005.

[10] G. Koch, R. Zemel, and R. Salakhutdinov, “Siamese neural networks for one-shot image recognition,” in *ICML workshop on Deep Learning*, 2015.

[11] P. Bankhead, M. B. Loughrey, and and col., “QuPath: Open source software for digital pathology image analysis,” *Scientific Reports*, vol. 7, no. 1, p. 16878.

[12] G. Bradski, “The OpenCV Library,” *Dr. Dobb’s Journal of Software Tools*, 2000.

[13] Y. LeCun and Y. Bengio, “Convolutional networks for images, speech, and time series,” *The handbook of brain theory and neural networks*, vol. 3361, no. 10, 1995.

[14] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *CoRR*, vol. abs/1412.6980, 2014.

[15] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting,” *Journal of Machine Learning Research*, vol. 15, pp. 1929–1958, 2014.

[16] M. Macenko, M. Niethammer, J. S. Marron, D. Borland, J. T. Woosley, X. Guan, C. Schmitt, and N. E. Thomas, “A method for normalizing histology slides for quantitative analysis,” in *IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pp. 1107—-1110, 2009.

Chapter 6

Conclusions and Future work

In this dissertation, we have addressed several challenging image and video processing and classification problems using DL based models. We have shown that DL-based models can be applied successfully to the tasks of image and video SR, image restoration, threat detection in PMMWIs and mitosis detection in WSIs. In all previous problems, the introduction of domain knowledge into the DL model is a critical factor, significantly increasing the model's performance and overcoming the limitations of other state-of-art DL-based approaches. We have used several methods to introduce this information into the model: through the architecture design, introducing specialized layers, modifying the loss function, constraints or combining the model with other non-DL-based approaches. In all the problems studied in this dissertation, the solutions proposed outperform current state-of-art methods in critical scenarios.

This dissertation has been presented in the modality of "compendium" and structured in the following four blocks: video super-resolution, image restoration and super-resolution, threat detection in passive millimeter-wave images and mitosis detection in whole-slide images. The conclusions and vias of future work of each one of them are detailed below:

6.1 Video Super-Resolution

Conclusions:

- First, we have introduced a new DL-based model for multiple degradations VSR. Our model explicitly uses the LR image formation model as an input and to constrain the output of the network. It obtains better results and is more robust to multiple degradations than current VSR state-of-art, outperforming the closest competing model by 0.5dB-0.006(SSIM) on average.

- Second, we have proposed a new loss to train VSR models. This loss is a combination of adversarial and feature losses and a spatial smoothness constraint. Compared to other perceptual losses, this loss can improve the super-resolved frames' quality without introducing noticeable high-frequency artifacts. Our experiments have shown that the spatial smoothness constraint is the term responsible for removing these artifacts, significantly improving PSNR and SSIM by 0.47dB-0.15(SSIM) without harming the perceptual quality that improves sightly by 0.0002.

- Third, we have shown that the use of datasets with a diverse set of scenes and motions significantly improves the performance of GAN-based VSR models, improving the perceptual quality by 0.0015.
- Finally, we have proposed a new Recurrent CNN model for VSR that adapts the minimal Gated Recurrent Unit [129] to VSR. This model outperforms current state-of-art in terms of PSNR, SSIM and temporal consistency (0.6dB-0.026(SSIM)-0.03(T-RRED[130]) compared to non-recurrent CNN AVSR[117]), while being significantly faster (almost two times faster than AVSR[117]) thanks to the reuse of features of previous frames.
- Experiments show that without deformable attention, the gated combination cannot be performed properly. Thus, no significant improvement can be seen in the results. The combination of deformable attention and the gated combination significantly improve the results in terms of PSNR, SSIM and temporal consistency, showing an improvement of 0.15dB-0.004(SSIM)-0.10(T-RRED[130]) over a normal recurrent CNN.

Future work:

- Our GAN-based model for multiple degradations VSR can be adapted to more advanced GAN frameworks, like Wasserstein GANs [104], being able to benefit from improvements in efficiency and stability.
- Our recurrent CNN model can be improved with the introduction of local recurrent connections, enhancing the flow of information across time and feature reuse.
- The two models proposed for VSR can be combined to obtain a model for multiple degradations VSR with the advantages of both.

6.2 Image Restoration and Super-Resolution

Conclusions:

- We have introduced two models, one for BID and the other for multiple degradations SR. Both models outperform current state-of-art for BID with uniform blur

(0.62dB-0.02(SSIM) over the second-best) and multiple degradations SR (1.14dB-0.03(SSIM) over the second-best for scaling factor 2 and complex blurs). Furthermore, both models are faster than most competing methods.

- Through our experiments, we have found that Wiener filtered images are an effective way of introducing blur kernel information in a CNN. This improves the performance of the network and allows it to generalize to unseen blurs during training. In the case of our cascade CNN model for multiple degradations SR, motion-corrected features are obtained by combining multi-scale features from the LR and the Wiener filtered image using deformable convolutions. This significantly improves the performance of the model (0.37dB-0.010(SSIM)).

- Affine projection is an effective method to constrain the network's function space, improving the model's performance by easing the learning and allowing for better minima. Furthermore, the residual error introduced by the presence of noise and blur kernel estimation errors is not too large, so it can be removed using a refinement sub-module or the filters from a dynamic filter network [133]. We have used affine projection for our BID approximation and found that it allows our model to compete with similar networks that are twice as deep (less than 0.05dB of difference). We have also used the affine projection in our cascade CNN model to separate the last sub-module into upsampling and image refinement sub-modules, further improving the results by 0.14dB-0.003(SSIM).

- The filters from a dynamic filter network [133] can be used to remove artifacts in the restored image caused by an inexact PSF estimation or other degradation model inconsistencies. In BID, this improves the model's performance significantly by 0.22dB-0.018(SSIM). In the case of multiple degradations SR, these filters can be used to reuse partial results from previous sub-modules in the cascade. If combined with connecting each sub-module using the previous sub-modules' features, the results improve by 0.26dB-0.007(SSIM).

- In the case of a cascade CNN for multiple degradation SR, "Privileged Information" [134] can be used to guide the training of the deblur sub-module to produce images that are easier to upsample. This increases the performance of the model by 0.28dB-0.007(SSIM).

Future work:

- The current implementation of the blur estimation is done in the CPU. If implemented in the GPU, the speed of our BID approximation will increase significantly.

- The model proposed for multiple degradation SR is not blind, needing an estimation of the blur kernel. Although we have shown that it could work with any kernel estimation method, for very high scaling factors LR images do not contain enough information to make a good kernel approximation. However, if the image is super-resolved progressively, following [137], the blur kernel estimation can be done progressively. This will allow our model to be applied to increasing scaling factors.

6.3 Threat Detection in Passive Millimeter-Wave Images

Conclusions:

- Based on our study, a new shallow ML approach to the detection task was developed using Haar features and Random Forest. This method detects 94% of all the database threats but produces a high number of false positives. Nevertheless, it can obtain a 0.75 AUC when detecting images containing a threat. Our approach outperforms previous threat detection methods for PMMWIs, detecting 34% more threats for a similar rate of false positives. This establishes a new state-of-art. Previous methods can be used but are most effective with other types of images, namely active millimeter-wave images.

- Our experiments show that this task's main difficulty arises from the low signal to noise ratio and non-stationary noise that populates an image. Because of this noise, shallow ML models perform better if Haar features and noise-removal prepossessing algorithms are used.

- Finally, we have performed a second study incorporating DL for threat detection. Two DL approaches have been studied: DLA and SGA. To deal with the non-stationary noise of PMMWIs, a DL method using the whole image (SGA) should be used. This increases in 5% the number of true positives and reduces in 4 the false positives per image, reaching almost zero.

- Multi-scale information and strong regularization are critical elements of our approximation. Thanks to these two elements, our model can detect all the database threats while having almost no false positives.

- Surprisingly and in contrast to shallow ML models, DL models perform better if noise-removal image processing techniques are not used. We have observed a decrease in AUC of 0.095 for DLA and 0.028 for SGA.

Future work:

- Although the database of PMMWIs used in this dissertation is the one with the largest number of instances, to the best of our knowledge, it is still too small for DL-based methods (only 3309 images). Obtaining a larger number of samples with more variety of threats will allow us to increase the model's complexity without overfitting.

- Since one of the main difficulties in this task arises from the low quality of the images, we believe that this problem is an excellent candidate to be solved using a cascade architecture similar to the one proposed in Section 3.2. Moreover, active millimeter-wave images can guide the part of the network responsible for enhancing the images. This would be a form of "Privileged Information" [134].

6.4 Mitosis Detection in Whole-Slide Images

Conclusions:

- We have proposed a fast DL-based pyramidal mitosis detection algorithm and shown that the uncertainty calculated using Bayesian dropout [107] can be used to pass information through the scales and prune out false positives caused by artifacts in the WSI. Experiments done using WSIs of melanoma skin cancer tissue have shown that it can outperform DL models of similar complexity by 7% in F1 score while being almost two times faster.

- Furthermore, the introduction of Spatial Transforming Layers [108] has been shown to improve our model's robustness to geometric deformations of the tissue, increasing the performance in F1 score by 3%.

- Moreover, we have shown that our model can be used for knowledge transfer between tissues. State-of-art performance in breast mitosis detection is reached by training a shallow classifier with the features of our model, which was trained on melanoma skin cancer tissue.

- Finally, a novel model for detecting mitosis using information from H&E and PHH3 stained tissue has been proposed. The method uses Surf[135] points and a Siamese CNN [136] to find matches between images of both tissues and classify those matches using a CNN. Our experiments show that using both stains significantly increases the performance of the model (7% increase in F1 score over a model trained with H&E and 3% over only using PHH3), even when errors in the matching algorithm are taken into account.

Future work:

- Self-supervised learning [138, 139] can be used to learn useful representations of the image data that work across tissue types without the need for labeled data. A classifier for each tissue type can then be learned using this representation and a small amount of labeled data.

- Current mitosis detection algorithm will benefit from an improvement in the quality of available images. Because of the nature of the scanning process, a significant percentage of the WSIs of some tissue types presents out-of-focus blur. Mitosis detection is a very challenging problem that requires much detail; thus, detection algorithms' performance significantly drops in regions of the WSI that presents blur.

Bibliography

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1026–1034.

[2] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, "Deep learning for generic object detection: A survey," *International Journal of Computer Vision*, vol. 128, no. 2, pp. 261–318, Feb 2020. [Online]. Available: https://doi.org/10.1007/s11263-019-01247-4

[3] L. Yujiri, M. Shoucri, and P. Moffa, "Passive millimeter wave imaging," *IEEE Microwave Magazine*, vol. 4, no. 3, pp. 39–50, 2003.

[4] L. Yujiri, "Passive millimeter wave imaging," in *IEEE MTT-S International Microwave Symposium Digest*, June 2006, pp. 98–101.

[5] S. Oka, H. Togo, N. Kukutsu, and T. Nagatsuma, "Latest Trends in Millimeter-Wave Imaging Technology," *Progress In Electromagnetics Research Letters*, vol. 1, pp. 197–204, 2008.

[6] T. K. Ten Kate, J. A. M. Belien, A. W. M. Smeulders, and J. P. A. Baak, "Method for counting mitoses by image processing in feulgen stained breast cancer sections," *Cytometry*, vol. 14, no. 3, pp. 241–250, 1993. [Online]. Available: https://ivi.fnwi.uva.nl/isis/publications/1993/TenKateCytometry1993

[7] D. C. Cireşan, A. Giusti, L. M. Gambardella, and J. Schmidhuber, "Mitosis detection in breast cancer histology images with deep neural networks," in *International Conference on Medical Image Computing and Computer-assisted Intervention*. Springer, 2013, pp. 411–418.

[8] S. Punitha, A. Amuthan, and K. S. Joseph, "Benign and malignant breast cancer segmentation using optimized region growing technique," *Future Computing and Informatics Journal*, vol. 3, no. 2, pp. 348 – 358, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2314728818300679

[9] K. Senthil Kumar, K. Venkatalakshmi, and K. Karthikeyan, "Lung cancer detection using image segmentation by means of various evolutionary algorithms," *Computational and Mathematical Methods in Medicine*, vol. 2019, p. 4909846, Jan 2019. [Online]. Available: https://doi.org/10.1155/2019/4909846

[10] S. D. Babacan, R. Molina, and A. K. Katsaggelos, "Variational Bayesian super resolution," *IEEE Transactions on Image Processing*, vol. 20, no. 4, pp. 984–999, 2011.

[11] S. P. Belekos, N. P. Galatsanos, and A. K. Katsaggelos, "Maximum a posteriori video super-resolution using a new multichannel image prior," *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1451–1464, 2010.

[12] P. Ruiz, X. Zhou, J. Mateos, R. Molina, and A. K. Katsaggelos, "Variational bayesian blind image deconvolution: A review," *Digital Signal Processing*, vol. 47, pp. 116 – 127, 2015, special Issue in Honour of William J. (Bill) Fitzgerald. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S105120041500144X

[13] E. J. Kaman, A. W. M. Smeulders, P. W. Verbeek, I. T. Young, and J. P. A. Baak, "Image processing for mitoses in sections of breast cancer: A feasibility study," *Cytometry*, vol. 5, no. 3, pp. 244–249, 1984. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/cyto.990050305

[14] J. A. M. Beliën, J. P. A. Baak, P. J. van Diest, and A. H. M. van Ginkel, "Counting mitoses by image processing in feulgen stained breast cancer sections: The influence of resolution," *Cytometry*, vol. 28, no. 2, pp. 135–140, 1997. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/%28SICI%291097-0320%2819970601%2928%3A2%3C135%3A%3AAID-CYTO6%3E3.0.CO%3B2-E

[15] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern Recognition*, vol. 29, pp. 51–59, Jan. 1996.

[16] C. Papageorgiou, M. Oren, and T. Poggio, "A general framework for object detection," in *6th International Conference on Computer Vision*, Jan 1998, pp. 555–562.

[17] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *European Conference on Computer Vision (ECCV)*, K. Daniilidis, P. Maragos, and N. Paragios, Eds., Berlin, Heidelberg, 2010, pp. 143–156.

[18] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010, pp. 3304–3311.

[19] L. Deng and D. Yu, "Deep learning: Methods and applications," Microsoft, Tech. Rep. MSR-TR-2014-21, May 2014. [Online]. Available: https://www.microsoft.com/en-us/research/publication/deep-learning-methods-and-applications/

[20] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning.* MIT Press, 2016, http://www.deeplearningbook.org.

[21] A. Damianou and N. Lawrence, "Deep gaussian processes," in *Proceedings of Machine Learning Research (PMLR)*, C. M. Carvalho and P. Ravikumar, Eds., vol. 31, Scottsdale, Arizona, USA, 29 Apr–01 May 2013, pp. 207–215.

[22] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

[23] F. Liu, G. Lin, and C. Shen, "CRF learning with CNN features for image segmentation," *Pattern Recognition*, vol. 48, no. 10, pp. 2983–2992, Oct. 2015.

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[25] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[26] R. Girshick, "Fast r-cnn," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 1440–1448.

[27] S. Valipour, M. Siam, M. Jagersand, and N. Ray, "Recurrent fully convolutional networks for video segmentation," in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2017, pp. 29–36.

[28] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 2, pp. 295–307, Feb. 2016.

[29] J. Kim, J. K. Lee, and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 1646–1654.

[30] A. Kappeler, S. Yoo, Q. Dai, and A. K. Katsaggelos, "Video super-resolution with convolutional neural networks," *IEEE Transactions on Computational Imaging*, vol. 2, no. 2, pp. 109–122, 2016.

[31] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3142–3155, 2017.

[32] O. Kupyn, T. Martyniuk, J. Wu, and Z. Wang, "Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 8877–8886.

[33] Y. Bengio and Y. Lecun, *Scaling Learning Algorithms towards AI.* MIT Press, 2007.

[34] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," in *Proceedings of the IEEE*, 1998, pp. 2278–2324.

[35] S. Ghosh, R. Shet, P. Amon, A. Hutter, and A. Kaup, "Robustness of deep convolutional neural networks for image degradations," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 2916–2920.

[36] Y. Pei, Y. Huang, Q. Zou, X. Zhang, and S. Wang, "Effects of image degradation and degradation removal to cnn-based image classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2019.

[37] J. Hadamard, *Lectures on Cauchy's Problem in Linear Partial Differential Equations.* New Haven, CT: Yale University Press, 1923.

[38] D. Perrone, R. Diethelm, and P. Favaro, "Blind deconvolution via lower-bounded logarithmic image priors," in *International Conference on Energy Minimization Methods in Computer Vision and Pattern Recognition (EMMCVPR)*, 2015.

[39] J. Pan, D. Sun, H. Pfister, and M. Yang, "Blind image deblurring using dark channel prior," in *2016 IEEE Conf. Comput. Vision and Pattern Recognition*, 2016, pp. 1628–1636.

[40] B. Zhang, R. Liu, H. Li, Q. Yuan, X. Fan, and Z. Luo, "Blind Image Deblurring Using Adaptive Priors," in *Internet Multimedia Computing and Service*, ser. Communications in Computer and Information Science. Springer, Singapore, Aug. 2017, pp. 13–22.

[41] J. Kotera, V. Šmídl, and F. Šroubek, "Blind deconvolution with model discrepancies," *IEEE Transactions on Image Processing*, vol. 26, no. 5, pp. 2533–2544, May 2017.

[42] Y. Yan, W. Ren, Y. Guo, R. Wang, and X. Cao, "Image deblurring via extreme channels prior," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6978–6986.

[43] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," *arXiv preprint arXiv:1609.04802*, 2016.

[44] C. Sonderby, J. Caballero, L. Theis, W. Shi, and F. Huszar, "Amortised MAP inference for image super-resolution," in *International Conference on Learning Representations*, 2017. [Online]. Available: https://arxiv.org/abs/1610.04490

[45] J. Caballero, C. Ledig, A. Aitken, A. Acosta, J. Totz, Z. Wang, and W. Shi, "Real-time video super-resolution with spatio-temporal networks and motion compensation," *arXiv preprint arXiv:1611.05250*, 2016.

[46] S. Nah, T. H. Kim, and K. M. Lee, "Deep multi-scale convolutional neural network for dynamic scene deblurring," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 257–265.

[47] T. Meinhardt, M. Möller, C. Hazirbas, and D. Cremers, "Learning proximal operators: Using denoising networks for regularizing inverse imaging problems," in *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, 2017, pp. 1799–1808.

[48] W. Dong, P. Wang, W. Yin, G. Shi, F. Wu, and X. Lu, "Denoising prior driven deep neural network for image restoration," *Computing Research Repository*, vol. abs/1801.06756, 2018.

[49] K. Zhang, W. Zuo, and L. Zhang, "Learning a single convolutional super-resolution network for multiple degradations," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[50] J. Gu, H. Lu, W. Zuo, and C. Dong, "Blind super-resolution with iterative kernel correction," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[51] K. Zhang, W. Zuo, and L. Zhang, "Deep plug-and-play super-resolution for arbitrary blur kernels," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[52] N. Alexander, C. Callejero Andres, and R. Gonzalo, "Multispectral mm-wave imaging: materials and images," *Proceedings of Society of Photo-Optical Instrumentation Engineers (SPIE)*, vol. 6948, pp. 694 803–694 803–10, 2008.

[53] J. Accardo and M. A. Chaudhry, “Radiation exposure and privacy concerns surrounding full-body scanners in airports,” *Journal of Radiation Research and Applied Sciences*, vol. 7, no. 2, pp. 198–200, 2014.

[54] Y. Meng, A. Qing, C. Lin, J. Zang, Y. Zhao, and C. Zhang, “Passive millimeter wave imaging system based on helical scanning,” *Scientific Reports*, vol. 8, no. 1, p. 7852, May 2018. [Online]. Available: https://doi.org/10.1038/s41598-018-25637-9

[55] S. D. Babacan, M. Luessi, L. Spinoulas, A. K. Katsaggelos, N. Gopalsami, T. Elmer, R. Ahern, S. Liao, and A. Raptis, “Compressive passive millimeter-wave imaging,” in *18th IEEE International Conference on Image Processing*, Sept 2011, pp. 2705–2708.

[56] B. Han, J. Xiong, L. Li, J. Yang, and Z. Wang, “Research on millimeter-wave image denoising method based on contourlet and compressed sensing,” in *2nd International Conference on Signal Processing Systems*, vol. 2, July 2010, pp. 471–475.

[57] W. Yu, X. Chen, S. Dong, and W. Shao, “Study on image enhancement algorithm applied to passive millimeter-wave imaging based on wavelet transformation,” in *International Conference on Electrical and Control Engineering*, Sept 2011, pp. 856–859.

[58] J. Mateos, A. López, M. Vega, R. Molina, and A. Katsaggelos, “Multiframe blind deconvolution of passive millimeter wave images using variational dirichlet blur kernel estimation,” in *IEEE International Conference on Image Processing*, Sept 2016, pp. 2678–2682.

[59] W. Saafin, S. Villena, M. Vega, R. Molina, and A. Katsaggelos, “Compressive sensing super resolution from multiple observations with application to passive millimeter wave images,” *Digital Signal Processing*, vol. 50, pp. 180–190, 2016.

[60] C. Haworth, B. Gonzalez, M. Tomsin, R. Appleby, P. Coward, A. Harvey, K. Lebart, Y. Petillot, and E. Trucco, “Image analysis for object detection in millimetre-wave images,” in *Passive Millimetre-wave and Terahertz Imaging and Technology*, vol. 5619, December 2004, pp. 117–128.

[61] C. Haworth, Y. Petillot, and E. Trucco, “Image processing techniques for metallic object detection with millimetre-wave images,” *Pattern Recognition Letters*, vol. 27, no. 15, pp. 1843–1851, 2006.

[62] C. D. Haworth, Y. De Saint-Pern, D. Clark, E. Trucco, and Y. R. Petillot, “Detection and tracking of multiple metallic objects in millimetre-wave images,” *International Journal of Computer Vision*, vol. 71, no. 2, pp. 183–196, 2007.

[63] O. Martínez, L. Ferraz, X. Binefa, I. Gómez, and C. Dorronsoro, "Concealed object detection and segmentation over millimetric waves images," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*, June 2010, pp. 31–37.

[64] S. Yeom, D.-S. Lee, Y. Jang, M.-K. Lee, and S.-W. Jung, "Real-time concealed-object detection and recognition with passive millimeter wave imaging," *Optics Express*, vol. 20, no. 9, pp. 9371–9381, Apr 2012.

[65] S. Yeom, D.-S. Lee, and J.-Y. Son, "Shape feature analysis of concealed objects with passive millimeter wave imaging," *Progress In Electromagnetics Research Letters*, vol. 57, pp. 131–137, 2015.

[66] S. Agarwal, A. S. Bisht, D. Singh, and N. P. Pathak, "A novel neural network based image reconstruction model with scale and rotation invariance for target identification and classification for active millimetre wave imaging," *Journal of Infrared, Millimeter, and Terahertz Waves*, vol. 35, no. 12, pp. 1045–1067, 2014.

[67] V. Jain and S. Sebastian, "Natural image denoising with convolutional networks," in *Advances in Neural Information Processing Systems 21*, D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, Eds. Curran Associates, Inc., 2009, pp. 769–776.

[68] M. Titford, "The long history of hematoxylin," *Biotechnic & histochemistry : official publication of the Biological Stain Commission*, vol. 80, pp. 73–8, 03 2005.

[69] M. Sebai, X. Wang, and T. Wang, "Maskmitosis: a deep learning framework for fully supervised, weakly supervised, and unsupervised mitosis detection in histopathology images," *Medical & Biological Engineering & Computing*, vol. 58, no. 7, pp. 1603–1623, Jul 2020. [Online]. Available: https://doi.org/10.1007/s11517-020-02175-z

[70] E. Reinhard, M. Ashikhmin, B. Gooch, and P. Shirley, "Color transfer between images," *IEEE Computer Graphics and Applications*, vol. 21, no. 5, pp. 34–41, July 2001.

[71] M. Macenko, M. Niethammer, J. S. Marron, D. Borland, J. T. Woosley, X. Guan, C. Schmitt, and N. E. Thomas, "A method for normalizing histology slides for quantitative analysis," in *IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, 2009, pp. 1107—-1110.

[72] N. Hidalgo-Gavira, J. Mateos, M. Vega, R. Molina, and A. K. Katsaggelos, "Variational bayesian blind color deconvolution of histopathological images," *IEEE Transactions on Image Processing*, vol. 29, pp. 2026–2036, 2020.

[73] L. Roux, D. Racoceanu, N. Lomenie, M. Kulikova, H. Irshad, J. Klossa, F. Capron, C. Genestie, G. L. Naour, and M. N. Gurcan, “Mitosis detection in breast cancer histological images an icpr 2012 contest,” *J Pathol Inform*, 2013.

[74] ICPR, “https://mitos-atypia-14.grand-challenge.org/,” 2014.

[75] MICCAI, “http://tupac.tue-image.nl/,” 2016.

[76] H. Wang, A. Cruz-Roa, A. Basavanhally, H. Gilmore, N. Shih, M. Feldman, J. Tomaszewski, F. A. González, and A. Madabhushi, “Mitosis detection in breast cancer pathology images by combining handcrafted and convolutional neural network features,” *Jour. Medical Imaging*, vol. 1, 2014.

[77] H. Chen, X. Wang, and P. A. Heng, “Automated mitosis detection with deep regression networks,” in *IEEE Int Symp Biomedical Imaging.*, 2016, pp. 1204—-1207.

[78] E. Zerhouni, D. Lányi, M. Viana, and M. Gabrani, “Wide residual networks for mitosis detection,” in *IEEE 14th International Symposium on Biomedical Imaging (ISBI)*, 2017, pp. 924–928.

[79] C. Li, X. Wanga, W. Liua, and L. J. Latecki, “Deepmitosis: Mitosis detection via deep detection, verification and segmentation networks,” *Medical Image Analysis*, pp. 121–133, 2018.

[80] X. Glorot, A. Bordes, and Y. Bengio, “Deep sparse rectifier neural networks,” in *Proceedings of Machine Learning Research (PMLR)*, G. Gordon, D. Dunson, and M. Dudík, Eds., vol. 15. Fort Lauderdale, FL, USA: JMLR Workshop and Conference Proceedings, 11–13 Apr 2011, pp. 315–323.

[81] R. Rojas, *The Backpropagation Algorithm.* Berlin, Heidelberg: Springer Berlin Heidelberg, 1996, pp. 149–182.

[82] J. Duchi, E. Hazan, and Y. Singer, “Adaptive subgradient methods for online learning and stochastic optimization,” *Journal of Machine Learning Research*, vol. 12, no. 61, pp. 2121–2159, 2011. [Online]. Available: http://jmlr.org/papers/v12/duchi11a.html

[83] M. D. Zeiler, “Adadelta: An adaptive learning rate method,” *CoRR*, vol. abs/1212.5701, 2012. [Online]. Available: http://dblp.uni-trier.de/db/journals/corr/corr1212.html#abs-1212-5701

[84] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980

[85] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," ser. Proceedings of Machine Learning Research, vol. 9. Chia Laguna Resort, Sardinia, Italy: JMLR Workshop and Conference Proceedings, 13–15 May 2010, pp. 249–256.

[86] J. F. Kolen and S. C. Kremer, *Gradient Flow in Recurrent Nets: The Difficulty of Learning LongTerm Dependencies.* Wiley-IEEE Press, 2001, pp. 237–243.

[87] D. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)," in *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2016.

[88] O. Ronneberger, P.Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, ser. LNCS, vol. 9351. Springer, 2015, pp. 234–241, (available on arXiv:1505.04597 [cs.CV]). [Online]. Available: http://lmb.informatik.uni-freiburg.de/Publications/2015/RFB15a

[89] Y. Kim, "Convolutional neural networks for sentence classification," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP).* Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1746–1751. [Online]. Available: https://www.aclweb.org/anthology/D14-1181

[90] S. Abdoli, P. Cardinal, and A. Lameiras Koerich, "End-to-end environmental sound classification using a 1d convolutional neural network," *Expert Systems with Applications*, vol. 136, pp. 252 – 263, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957417419304403

[91] C. Szegedy, Wei Liu, Yangqing Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1–9.

[92] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

[93] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788.

[94] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel, "Imagenet-trained CNNs are biased towards texture;

increasing shape bias improves accuracy and robustness." in *International Conference on Learning Representations*, 2019. [Online]. Available: https://openreview.net/forum?id=Bygh9j09KX

[95] D. P. Kingma and M. Welling, "Auto-encoding variational bayes." in *2014 The International Conference on Learning Representations (ICLR)*, Y. Bengio and Y. LeCun, Eds., 2014.

[96] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[97] X. Wang, K. Yu, C. Dong, and C. Change Loy, "Recovering realistic texture in image super-resolution by deep spatial feature transform," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 606–615.

[98] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, C. C. Loy, Y. Qiao, and X. Tang, "Esrgan: Enhanced super-resolution generative adversarial networks," in *ECCV Workshops*, 2018.

[99] M. Zareapoor, H. Zhou, and J. Yang, "Perceptual image quality using dual generative adversarial network," *Neural Computing and Applications*, pp. 1–11, 2019.

[100] T. M. Quan, T. Nguyen-Duc, and W.-K. Jeong, "Compressed sensing mri reconstruction with cyclic loss in generative adversarial networks," *arXiv preprint arXiv:1709.00753*, 2017.

[101] O. Kupyn, V. Budzan, M. Mykhailych, D. Mishkin, and J. Matas, "Deblurgan: Blind motion deblurring using conditional adversarial networks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2018, pp. 8183–8192.

[102] S. Nowozin, B. Cseke, and R. Tomioka, "f-gan: Training generative neural samplers using variational divergence minimization," in *Advances in Neural Information Processing Systems (NIPS)*, 2016.

[103] T. Che, Y. Li, A. P. Jacob, Y. Bengio, and W. Li, "Mode regularized generative adversarial networks," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017.

[104] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proceedings of Machine Learning Research (PMLR)*, D. Precup and Y. W. Teh, Eds., vol. 70, International Convention Centre, Sydney, Australia, 06–11 Aug 2017, pp. 214–223.

[105] D. Ulyanov, A. Vedaldi, and V. S. Lempitsky, “Deep image prior,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[106] X. Pan, X. Zhan, B. Dai, D. Lin, C. C. Loy, and P. Luo, “Exploiting deep generative prior for versatile image restoration and manipulation,” *European Conference on Computer Vision (ECCV)*, 2020.

[107] Y. Gal and Z. Ghahramani, “Dropout as a Bayesian approximation: Representing model uncertainty in deep learning,” in *Proceedings of the 33rd International Conference on Machine Learning (ICML-16)*, 2016.

[108] M. Jaderberg, K. Simonyan, A. Zisserman, and k. kavukcuoglu, “Spatial transformer networks,” in *Advances in Neural Information Processing Systems 28*, 2015, pp. 2017–2025.

[109] N. Shazeer, A. Mirhoseini, K. Maziarz, A. Davis, Q. Le, G. Hinton, and J. Dean, “Outrageously large neural networks: The sparsely-gated mixture-of-experts layer,” in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017.

[110] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, “Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising,” *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3142–3155, 2017.

[111] F. Xing, Y. Xie, X. Shi, P. Chen, Z. Zhang, and L. Yang, “Towards pixel-to-pixel deep nucleus detection in microscopy images,” *BMC Bioinformatics*, vol. 20, no. 1, p. 472, Sep 2019. [Online]. Available: https://doi.org/10.1186/s12859-019-3037-5

[112] C. Tapia, H. Kutzner, T. Mentzel, S. Savic, D. Baumhoer, and K. Glatz, “Two mitosis-specific antibodies, mpm-2 and phospho-histone h3 (ser28), allow rapid and precise determination of mitotic activity,” *Am J Surg Pathol*, vol. 30(1), pp. 83–9, 2006.

[113] P. S. Nielsen, R. Riber-Hansen, T. O. Jensen, H. Schmidt, and T. Steiniche, “Proliferation indices of phosphohistone h3 and ki67: strong prognostic markers in a consecutive cohort with stage i/ii melanoma,” *Modern Pathology*, vol. 26, p. 404, Nov. 2012.

[114] B. F. Dessauvagie, C. Thomas, C. Robinson, F. A. Frost, J. Harvey, and G. F. Sterrett, “Validation of mitosis counting by automated phosphohistone h3 (phh3) digital image analysis in a breast carcinoma tissue microarray,” *Pathology*, vol. 47, no. 4, pp. 329–334, Jun. 2015.

[115] S. López-Tapia, A. Lucas, R. Molina, and A. K. Katsaggelos, "Gan-based video super-resolution with direct regularized inversion of the low-resolution formation model," in *2019 IEEE International Conference on Image Processing (ICIP)*, 2019, pp. 2886–2890.

[116] S. López-Tapia, A. Lucas, R. Molina, and A. K. Katsaggelos, "Multiple-degradation video super-resolution with direct inversion of the low-resolution formation model," in *2019 27th European Signal Processing Conference (EUSIPCO)*, 2019, pp. 1–5.

[117] S. López-Tapia, A. Lucas, R. Molina, and A. K. Katsaggelos, "A single video super-resolution gan for multiple downsampling operators based on pseudo-inverse image formation models," *Digital Signal Processing*, vol. 104, p. 102801, 2020. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1051200420301469

[118] S. López-Tapia, A. Lucas, R. Molina, and A. K. Katsaggelos, "Gated recurrent networks for video super resolution," in *2020 28th European Signal Processing Conference (EUSIPCO)*, 2020.

[119] S. López-Tapia, J. Mateos, R. Molina, and A. K. Katsaggelos, "Combining analytical and deep learning methods in blind image deconvolution (submitted)," *IEEE Transactions on Image Processing*, 2020.

[120] S. López-Tapia and N. Pérez de la Blanca, "Fast and robust cascade model for multiple degradation single image super-resolution (submitted)," *IEEE Transactions on Image Processing*, 2020.

[121] S. López-Tapia, R. Molina, and N. Pérez de la Blanca, "Using machine learning to detect and localize concealed objects in passive millimeter-wave images," *Engineering Applications of Artificial Intelligence*, vol. 67, pp. 81 – 90, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0952197617302099

[122] S. López-Tapia, R. Molina, and N. P. de la Blanca, "Deep cnns for object detection using passive millimeter sensors," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 9, pp. 2580–2589, 2019.

[123] S. López-Tapia, J. Aneiros-Fernández, and N. Pérez de la Blanca, "A fast pyramidal bayesian model for mitosis detection in whole-slide images," in *Digital Pathology*, C. C. Reyes-Aldasoro, A. Janowczyk, M. Veta, P. Bankhead, and K. Sirinukunwattana, Eds. Cham: Springer International Publishing, 2019, pp. 135–143.

[124] S. López-Tapia, C. Olivencia, J. Aneiros-Fernández, and N. Pérez de la Blanca, "Improvement of mitosis detection through the combination of phh3 and he features," in *Digital Pathology*, C. C. Reyes-Aldasoro, A. Janowczyk, M. Veta, P. Bankhead, and K. Sirinukunwattana, Eds. Cham: Springer International Publishing, 2019, pp. 144–152.

[125] S. Abu-El-Haija, N. Kothari, J. Lee, A. P. Natsev, G. Toderici, B. Varadarajan, and S. Vijayanarasimhan, "Youtube-8m: A large-scale video classification benchmark," in *arXiv:1609.08675*, 2016. [Online]. Available: https://arxiv.org/pdf/1609.08675v1.pdf

[126] "Myanmar 60p, harmonic inc. (2014)," http://www.harmonicinc.com/resources/videos/4k-video-clip-center.

[127] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep networks as a perceptual metric," in *CVPR*, 2018.

[128] X. Zhu, H. Hu, S. Lin, and J. Dai, "Deformable convnets v2: More deformable, better results," *arXiv preprint arXiv:1811.11168*, 2018.

[129] J. C. Heck and F. M. Salem, "Simplified minimal gated unit variations for recurrent neural networks," in *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*, Aug 2017, pp. 1593–1596.

[130] R. Soundararajan and A. C. Bovik, "Video quality assessment by reduced reference spatio-temporal entropic differencing," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 4, pp. 684–694, April 2013.

[131] A. Lucas, A. K. Katsaggelos, S. Lopez-Tapuia, and R. Molina, "Generative adversarial networks and perceptual losses for video super-resolution," in *2018 25th IEEE International Conference on Image Processing (ICIP)*, 2018, pp. 51–55.

[132] A. Lucas, S. Lopez-Tapia, R. Molina, and A. K. Katsaggelos, "Efficient fine-tuning of neural networks for artifact removal in deep learning for inverse imaging problems," in *2019 IEEE International Conference on Image Processing (ICIP)*, 2019, pp. 3591–3595.

[133] X. Jia, B. De Brabandere, T. Tuytelaars, and L. V. Gool, "Dynamic filter networks," in *Advances in Neural Information Processing Systems 29*, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, Eds. Curran Associates, Inc., 2016, pp. 667–675.

[134] V. Vladimir and I. Rauf, "Learning using privileged information: Similarity control and knowledge transfer," *Journal of Machine Learning Research*, pp. 2023–2049, 2015.

[135] H. Bay, T. Tuytelaars, and L. Van Gool, “Surf: Speeded up robust features,” in *Computer Vision – ECCV 2006*, A. Leonardis, H. Bischof, and A. Pinz, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 404–417.

[136] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, “Signature verification using a ”siamese” time delay neural network,” in *Proceedings of the 6th International Conference on Neural Information Processing Systems*, ser. NIPS’93. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993, p. 737–744.

[137] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, “Fast and accurate image super-resolution with deep laplacian pyramid networks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[138] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton, “A simple framework for contrastive learning of visual representations,” *ArXiv*, vol. abs/2002.05709, 2020.

[139] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, “Unsupervised learning of visual features by contrasting cluster assignments,” *arXiv preprint arXiv:2006.09882*, 2020.