

IET Image Processing

Special issue Call for Papers

**Be Seen. Be Cited.
Submit your work to a new
IET special issue**

Connect with researchers and experts in your field and share knowledge.


Be part of the latest research trends, faster.

[Read more](#)



The Institution of
Engineering and Technology

Nonlinear kernel based feature maps for blur-sensitive unsharp masking of JPEG images

Jhilik Bhattacharya¹  | Stefano Marsi² | Giovanni Ramponi²

¹Thapar Institute of Engineering and Technology, Patiala, India

²DIA, University of Trieste, Trieste, Friuli-Venezia Giulia, Italy

Correspondence

Thapar Institute of Engineering and Technology, India.

Email: jhilik@thapar.edu

Funding information

University of Trieste; DST-TIDE, India, Grant/Award Number: SEED/TIDE/2018/82

Abstract

In this paper, a method for estimating the blur regions of an image is first proposed, resorting to a mixture of linear and nonlinear convolutional kernels. The blur map obtained is then utilized to enhance images such that the enhancement strength is an inverse function of the amount of measured blur. The blur map can also be used for tasks such as attention-based object classification, low light image enhancement, and more. A CNN architecture is trained with nonlinear upsampling layers using a standard blur detection benchmark dataset, with the help of blur target maps. Further, it is proposed to use the same architecture to build maps of areas affected by the typical JPEG artifacts, ringing and blockiness. The blur map and the artifact map pair permit to build an activation map for the enhancement of a (possibly JPEG compressed) image. Extensive experiments on standard test images verify the quality of the maps obtained using the algorithm and their effectiveness in locally controlling the enhancement, for superior perceptual quality. Last but not least, the computation time for generating these maps is much lower than the one of other comparable algorithms.

1 | INTRODUCTION

Detecting the presence, intensity and location of blur in an image (blur mapping) has a large set of useful applications including depth recovery [1], image segmentation [2], image enhancement [3], image quality assessment [4–9], video encoding quality assessment via blur analysis [10], and object detection [11]. In this paper we study an image enhancement case, where the amount of blur locally controls the intensity of the sharpening action of a proper operator in order to achieve a good balance between detail enhancement and the generation of artifacts. We also consider the very common case in which JPEG compression artifacts such as ringing and blocking are present in the image to be processed. Both artifacts, indeed, can be particularly insidious in images that contain sharp and blurred areas: the blocking effect is more visible in blurred areas, and the ringing effect caused by the border between an in-focus and an out-of-focus image portion is conspicuous in the blurred side. For this reason we also detect the presence, intensity and location of these artifacts. While the basic enhancement

operator can be straightforward Unsharp Masking (UM), the generation of a suitable pair of blur and artifact maps, and the way they are used in the system, are far from obvious: this is why we focus here on this system component.

UM-based image enhancement involves the addition of an amplified detail image layer to a base image layer. Existing research contributions in this area differ in the type of operator used to generate the detail layer (e.g. edge-preserving nonlinear filters [12–16]), or in the approach for computing the amplification factor. Adaptive UM techniques use local contrast, local variance, mean intensity [17, 18] to guide the enhancement operation. Recently, Ye and Ma presented the BUM method [3], a blurriness-guided scaling which proved to be effective in enhancing unblurred regions without amplifying noise. There, a scaling matrix for the detail layer is derived from a blur map, which in turn is obtained from the Just Noticeable Blur Estimation (JNBE) algorithm [19]. The present work focuses towards learning a blur map using a convolutional neural network; this dramatically shortens the processing time required for the map generation. Our blur map also provides better

This is an open access article under the terms of the [Creative Commons Attribution-NonCommercial-NoDerivs](https://creativecommons.org/licenses/by-nc-nd/4.0/) License, which permits use and distribution in any medium, provided the original work is properly cited, the use is non-commercial and no modifications or adaptations are made.

© 2022 The Authors. *IET Image Processing* published by John Wiley & Sons Ltd on behalf of The Institution of Engineering and Technology.

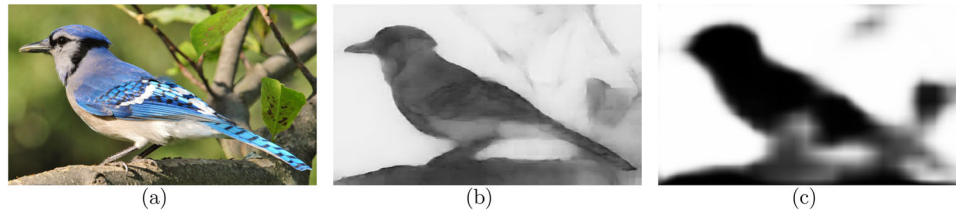


FIGURE 1 Example Bird image (a), our blur map (larger blurriness is represented by higher luminance) (b), and a state-of-the-art map (c)

enhancement results in terms of visual quality, especially for cases in which the portion of the image where blur is present is not clearly determined. Additionally, we propose and use a second map (the artifact map) to deal with JPEG compressed images so that blocking and ringing effects are not amplified during enhancement.

While there are a few researches in which blur maps are generated via a deep learning network [20], [30], our work focuses towards obtaining a continuous (256 levels) map instead of two or four discrete levels, as shown in Figure 1. This leads to a better final image quality, since a continuous map permits to perform a sharpening action whose strength varies smoothly across the image. Our other important contribution is that the companion artifact map we generate can avoid the amplification of the ringing and blocking effects typically present in compressed images. Together, the blur map and the artifact map form an activation map for the UM operator. It is interesting to note that blur-based Unsharp Masking and removal of blocking or ringing from JPEG images are dealt separately in the literature, without any overlap. With respect to [3], we also simplify the function that controls the sharpening action basing on the blur detection. This paper is organized in the following manner: Section 2 provides a brief review of existing blur analysis literature. The motivation for the current work is discussed in Section 2.1. Sections 3 and 4 provide the architecture details and the experimental results, respectively. Discussion and conclusion are given in Sections 5 and 6.

2 | STATE OF THE ART IN BLUR ANALYSIS

Traditional techniques used for blur analysis relied on phase prediction at step edges [21], image gradient distributions, maximum saturation, and power spectrum analysis. Indeed, blurred regions lead to a faster drop of power spectral scope compared to properly focused ones. Statistics of derivative filters are also altered prominently by blur, and were hence used to design deconvolution kernels [22, 23]. Gradient distribution measures were also used for classifying or identifying blur scores. A blur score was computed in [24] using a posterior measure on power spectrum, the local gradient, and some local discriminative features; a blur map was generated by combining blur responses from three scales using a loopy belief propagation technique. The Authors of [25] trained a Bayesian classifier using power spectral slopes, image gradient distributions, and maximum sat-

uration features to distinguish between blurred and unblurred regions. They further exploited the local autocorrelation congruency to identify motion blur and focal blur. Motion blur typically has strong gradients in one direction, whereas for focal blur all gradients are small. Blur resulting from object motion or defocus changes the blur kernel spatially across the image plane, making estimation difficult from a single image. A stronger image model, as compared to the uniform blur model, is required to extract the blur properties. Also, algorithms modeling a specific kind of blur proved unsuitable to remove other kinds of blur. For example, Gaussian difference signature [26] for modeling out-of-focus blur was not effective for non-Gaussian blurs. A global power spectrum analysis in these cases seems ineffective. While some algorithms [22] apply maximum log-likelihood estimates over multiple kernels to cater such cases, their assumptions were argued to be too simplistic. A local Fourier transform solution for this problem was proposed in [27]. It was further pointed out in [19] that with only local information it is not possible to distinguish between blurred regions and flat regions.

The success of such local statistical features depends on the type of image and blur. For example, they may be able to detect strong blur due to a narrow depth of field, but slight blur may go undetected. Currently, images captured with both professional cameras and mobiles may intentionally contain out-of-focus blur to make the foreground more prominent. This is achieved by exploiting the properties of the optics in high-range cameras, or by using dedicated software tools in low-range cameras. Enhancing these images, or performing quality assessment for these images, require a proper estimation of this blur. A sparse dictionary learning based technique presented by [19] claimed to tackle these efficiently. They used external data to capture statistical and structural data of blur templates. These are later used to classify new patches. Combinations of handcrafted as well as deep features were also applied for blur estimation. Park et al. [28] used DCT, gradient, SVD as well as deep features with a neural network classifier to predict the blur level. Their work uses 11 different σ values, treated as labels for blurring image patches. During testing, each input patch is classified into a particular label with some probability. These are later used as sparse maps and confidence maps, respectively, to generate the final blur map using joined bilateral filtering and Laplacian matting [29]. Blur map generation in this manner involves two basic challenges. Firstly, the precision of the map will depend on the number of classes (i.e. σ) used to generate the map. Also, the size of the image patches, that is, the patch scale, will

play an influential role. To deal with this issue, [28] selected patches based on the output of an edge detection operator. They relied on the fact that sharp patches can use a small patch size, while blurred patches require a much larger size. Hence they used smaller windows in strong edge regions and larger windows in weak edge regions. Though used for a slightly different purpose, viz. classifying faulty input images into seven fault categories including blur, [31] deals with scale by using a multicolumn GoogleNet architecture, where the first 8 inception layers are shared and the final layers are trained separately. One column deals with the downsampled holistic image, whereas the other uses image patches. Blur segmentation maps have also been generated using genetic programming, local binary patterns, local ternary patterns and Pulse coupled Neural Networks [48–50]. Different from all these works, [20] used a fully convolutional architecture to generate image blur maps from an entire image input. They argue that high-level semantic information is more crucial for detecting local blur. It should be noted that the desired map generated has values 0 and 1 only. Also, as patch-based σ values were not involved in the map generation process, their technique is free from class size or patch size variational factors. Deep CNN based blur detection is also carried out in [45], where a residual learning network was utilized to restore the structures in the scale space during deconvolutions. Another deep architecture for blur understanding was presented in [30], using a combination of blur map, attention map and content map to categorize the blur in an image as 'good', 'ok', 'bad' or 'no blur'. The blur map, attention map and content map are fused to classify the blur as desirable or undesired. The Authors use the Inception V2 network for generating blur maps and attention maps. Content feature maps are generated using features from a ResNet. A spatial pyramid pooling feature is used to combine local and global cues for detecting blurs in multiple scale objects. In order to detect blur in small regions, low level and high level features are combined for the blur map in contrast to using a local-global multicolumn architecture in [31]. While most deep learning based blur map generation aims towards a binary blur map, there are some deblurring CNNs which estimate the blur maps [46, 47] as a prerequisite to generate deblurred images. These maps are further used for deblurring purposes.

2.1 | Motivation

Motivated by the latter results, this work aims at generating a blur map from the entire input image by combining multi-level features along with nonlinear upscaling. The combination of features at different levels promote encapsulating feature resolutions at different scales.

Our method is suitable to work on both out-of-focus blur images and motion blur images. In both cases we aim at enhancing the object or region in focus, leaving the rest of the image as is. A similar approach is taken to deal with JPEG artifacts; that is, regions with ringing and blockiness are not enhanced. We do *not* aim at removing blur or JPEG artifacts: our goal is to use Deep Learning with nonlinear blocks to estimate local blur and

accordingly control UM. We will hence call the tool we propose Non-Linear Blur Map (NLBM).

To obtain satisfactory results in all circumstances, it is necessary to cope with various blurring effects. Both out-of-focus and motion-dependent effects strongly depend on the three-dimensional structure of the scene: images which are acquired range from being formed by two clearly different and well defined areas (we will label this as the *2-plane case*), to the case in which several different object planes are present and are thus subject to different amounts of blurriness (*n-plane case*). The most general case includes objects that extend at an angle of the camera axis, and generate images formed by a set of progressively scaled textures affected by a continuously varying amount of blurriness (*continuous-plane case*). For n-plane and continuous-plane cases, in particular, we need a graded rather than binary or multi-level blur map [20, 30, 45]. This is indeed the goal in the design of our nonlinear kernel based deep learning method. The generated blur map will be used as an attention guide for different tasks like low-light enhancement as well as detection and classification. The architecture used for the current purpose ensures that the intermediate features along with the target map can be used for multi-task learning.

Hence, ours is the first attempt at generating a continuous blur map for complex real-world scenes using supervised deep features only, compared to blur maps using combinations of deep and handcrafted features and handcrafted features only. The maps we produce capture enough blur level details without improperly including edge level details in both the in-focus and out-of-focus areas. Novel non-linear blocks introduce spatially variable weights, contributing to the spatial information retention during reconstruction. Further, the network architecture permits simultaneous use of features from different levels, thus analysing the image at multiple scale factors. This makes the map useful not only as blur-sensitive Unsharp Masking (one of the applications discussed in the paper) but also as an attention map for tasks like low-light enhancement, image translations, image segmentation etc. Standard operators for low light enhancement on poor quality images result in noise amplification. As shown in the experimental results, NLBM is able to suppress enhancements of blurred regions such that a simple gamma transformation works satisfactorily during inference. It is hence evident that using these maps as training attention maps will further enhance the performances.

3 | METHODOLOGY

We use the same network architecture to build both the blur map and the JPEG artifacts map. For the blur map, each image fed into the deep network outputs an image of the same size where each pixel is mapped in the range [0-1] continuously, signifying [low-high] blur level. Hence the out-of-focus regions in the map image are brighter compared to the focused parts. According to our experiments, the ability of the map to accurately represent the amount of blurring present even in the fine details of the image increases with the depth of the network.

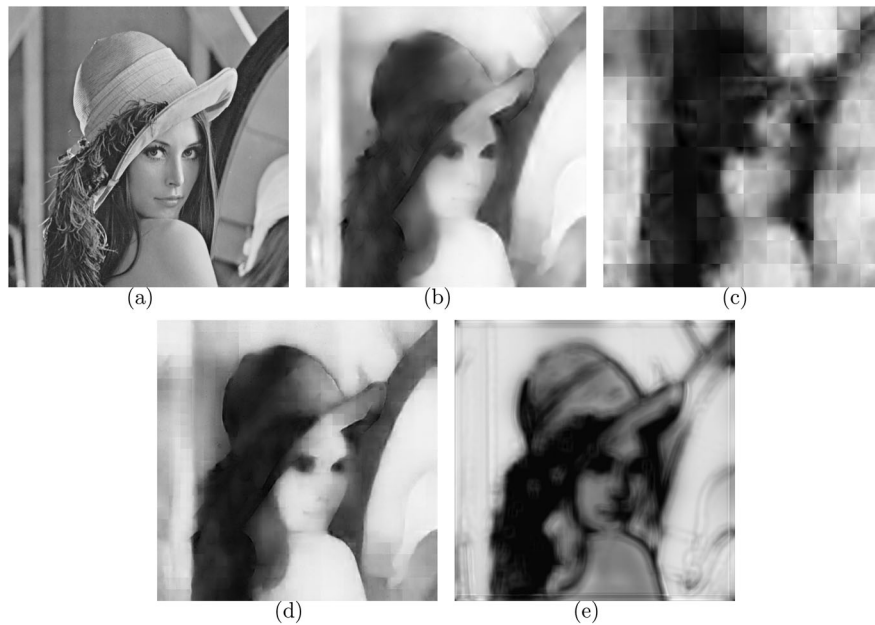


FIGURE 2 (a) Lena image, (b) our blur map, (c) map if single deconvolution is used, (d) multiple deconvolution showing blockiness, (e) shallow net usage

Hence, we have tried to select the best compromise between efficacy and complexity.

The architecture can be described subdividing it into three macro blocks: a feature generator module, a feature processor module, and a feature mixer. The *feature generator module* is used to generate and extract various features, representative of the input image, at different levels of abstraction. It is made up of multiple convolution-pooling and non-linear activation blocks. Any pretrained feature map extractor like ResNet, VGG or even AlexNet can be used for the purpose. The main difference will be in terms of training time (AlexNet will be faster). There will also be minute blur level differences depending on the backbone used even if they are not objectively noticeable via quality measurement scores as discussed later in Tables 5, 6. The use of a particular backbone is a user's choice, and also application dependent.

In decoding networks used for image reconstruction, the feature maps are then up-scaled with the use of skip connections. Incidentally, this strategy has often been used in different quite common neural architectures, such for instance in the U-Net where the bypass branches operates for mixing information extracted at different levels, leading to a considerable benefit in performance both in the inference phase and in the training. Although the blur map generation task is not a typical image reconstruction task, but similar to a reconstruction operation, we need to generate a single-channel blur map from a multi-channel feature space hence involving upsampling or deconvolution operations. We observe that better blur map reconstructions are obtained if multiple small step-ups (convolution plus upscaling) are carried out instead of a single convolution and large-factor interpolation. Skip connections and deeper layers further improve the performance. This is depicted in Figure 2. This finding is similar across different

applications. Different detection, classification or reconstruction tasks improve when the architecture involves deeper layers and skip connections.

After the first stage has provided a rich amount of different features, these are fed into *feature processor blocks* at various levels. Each feature processing block has 2 inputs, one from its parent block and another from the feature-map blocks. The cardinality of i can be varied. We have used up to $i = 4$ as shown in Figure 3.

Through an appropriate processing and recombination of features extracted from different levels, followed by a final stage of *feature mixing*, we obtain the desired output map. The simultaneous use of features at different levels, permitted by the upscale process, enables the system to analyze the image at different scale factors and therefore to exploit information extracted from areas quite far from the processing pixel, extending significantly the receptive field. Unlike the U-Net (or other networks that employ linear convolutional blocks to recombine the features before the upsampling), we designed an architecture based on an innovative NonLinear Convolutional block (NLC) which has proved very effective in processing images and features. The proposed architecture, which we call Non-Linear Convolutional Resampling Block (NLCRB), is formed by a sequence of nonlinear convolutions with concatenation operations as shown in Figure 4. There are two specific advantages of using a nonlinear convolution compared to a linear one; first, the number of learnable parameters in the model is far less compared to its linear counterpart. Secondly, the spatial variation-based weights retain more spatial information during reconstruction. These NLCRBs have a double function: they carry out a complex nonlinear processing of the input features, and they upscale the output doubling its spatial size. The latter step is achieved through the pixel-shuffle function which

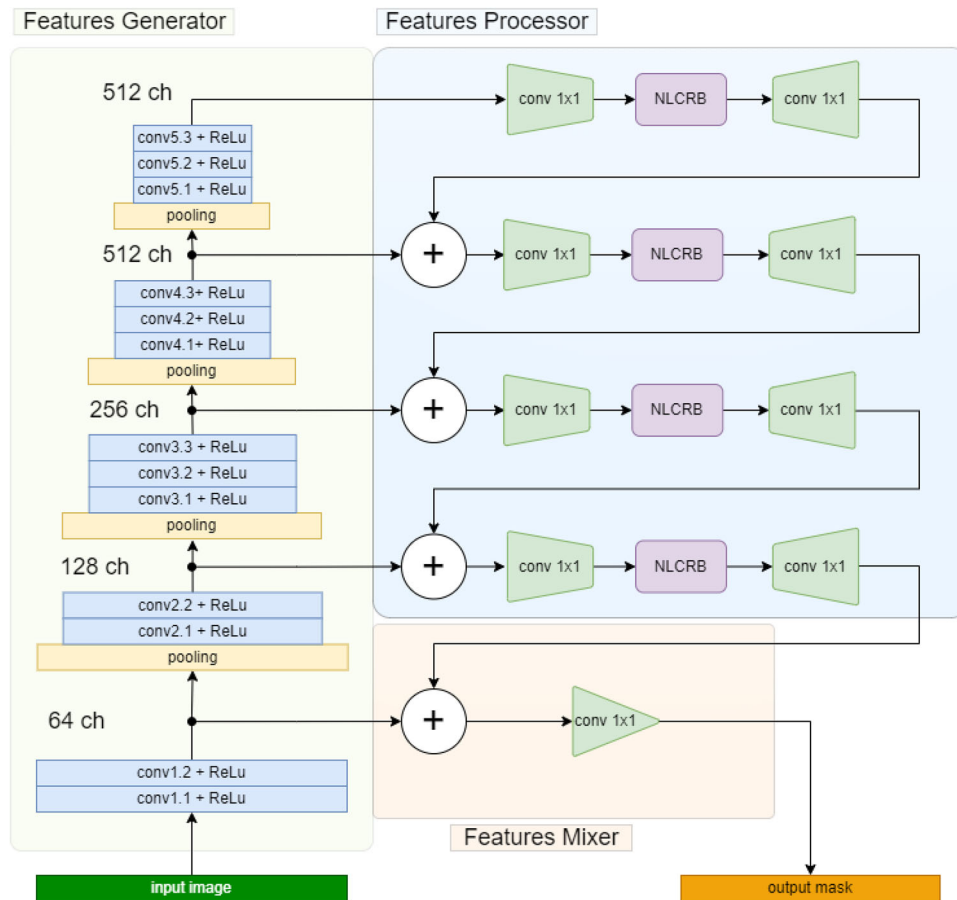


FIGURE 3 Blur map generation block (NLBM)

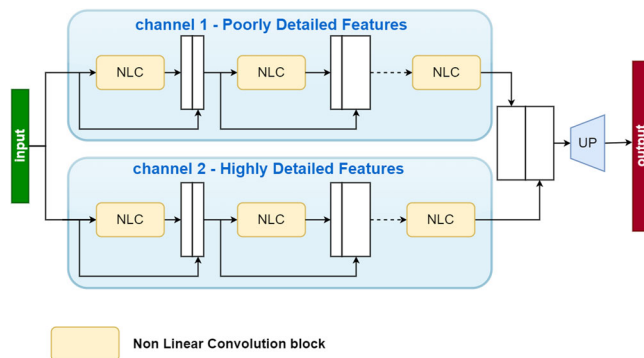


FIGURE 4 Structure of the nonlinear convolutional based resampling block (NLCRB)

reorders the pixels extracted from four separate input channels into a single output channel with double spatial size, performing actually an interpolation process. The nonlinear processing, instead, is realized through a series of NLCs, organized in two parallel channels (Figure 4).

The two channels have been created partially following the ResNet strategy. Like a ResNet, an NLCRB uses two channels, one of which processes the features while the second transfers the input signal “as is” to the output to combine them

together. In our case, however, the channels operate in a complementary way. One channel generates a signal characterized by smooth features, while the second one generates essentially highly detailed data. The output features are sent to the following stages, where they are combined to create a signal with all the components. The different channel behaviour is obtained through the choice of the parameters within the NLC blocks, that is, adopting different activation and normalization functions, as described below.

The NLC blocks, already introduced in [51], are nonlinear convolution blocks. While the linear convolution blocks commonly used in neural networks implement a convolution with fixed weights, in the NLC blocks such weights are continuously modified so that the system can take into account the different local characteristics of the input signal. The weights are in fact calculated continuously through three disjoint operations: a classic linear convolution with fixed weights, followed by a nonlinear activation function, which is finally followed by a normalization operation among all the weights. The convolution allows developing a system which, if properly trained, can change continuously its weights. Therefore, it can modulate its response differently in the different parts of the input signal. The choice of the activation function and the subsequent normalization is useful to impose a particular behaviour on the NLC. If the activation function is set to force the weights

to be all positive, like for instance a ReLU, and the normalization function forces the sum to be unitary (Equation 5), a system that penalizes high spatial frequencies and guarantees a unitary gain at low frequencies results. On the other hand, if we adopt an activation function that provides both positive and negative values and if the normalization is, for example, the sum of the absolute values of the weights (Equation 6), the system has a null continuous response and blocks low-frequency components. A general representation of the spatial dependent weights w_n is given in the following equations. Considering $x(i, j, p)$ and $y(i, j, l)$ as input and output and a convolution kernel of odd size $K \times W_1 \times W_1$, similar padding, unit stride and no dilation, the 2D convolution is given by Equation (1).

$$y(i, j, l) = \sum_{j=1}^K \sum_{n=1}^{W_1} \sum_{m=1}^{W_1} x(i+n-o_1, j+m-o_1, p) \cdot v_{i,j,l}(n, m, p), \quad (1)$$

where

$$o_1 = \frac{W_1 - 1}{2}. \quad (2)$$

K is the number of input features and W_1 is the size of the convolution kernel. Each weight $v_{i,j,l}(n, m, p)$ changes according to the input image through a further convolution of the input data with a kernel of odd size $K \times W_2 \times W_2$ composed by constant weights $u_{n,m,p}(r, s, q)_i$, followed by a nonlinear point-wise activation function (\mathcal{AF}) and a normalization process.

$$\hat{v}_{i,j,l}(n, m, p) = NL \left(\sum_{q=1}^K \sum_{r=1}^{W_2} \sum_{s=1}^{W_2} x(i+r-o_2, j+s-o_2, q) \cdot u_{n,m,p}(r, s, q)_i \right), \quad (3)$$

$$o_2 = \frac{W_2 - 1}{2}, \quad (4)$$

$$v_{i,j,l}(n, m, p) = \frac{\hat{v}_{i,j,l}(n, m, p)}{\sum_{n,m,p} \hat{v}_{i,j,l}(n, m, p) + \epsilon}, \quad (5)$$

or alternatively

$$v_{i,j,l}(n, m, p) = \frac{\hat{v}_{i,j,l}(n, m, p)}{\sum_{n,m,p} |\hat{v}_{i,j,l}(n, m, p)| + \epsilon}, \quad (6)$$

according to the desired behavior of the system.

The succession of several similar blocks in series also allows to improve the response of the system subdividing it in various stages and extending its receptive field. A further benefit is obtained if each stage of the chain can access all the features developed by all the previous stages.

The complete system, depicted in Figure 3, also makes extensive use of linear convolution blocks with a core dimension

$n \times 1 \times 1$. These layers are essential to compress or expand series of features through a linear weighted sum combination.

With an RGB image as its input, the developed architecture provides a pixel-by-pixel map of the amount of blurriness of the input image, that is, the blur map we need. The same architecture, with a different training, can recognize the pixels affected by JPEG artifacts and provide the artifact map.

To train the *blur map* network we use out-of-focus images from the CUHK [24] dataset. We also generated blur maps of these images using the technique presented in [3] and used them as target images for our network. A mean square error-based loss function is used for training. The target blur map is a smoothed blur version of the map generated using the JNBE algorithm [19]. 256 basis vectors are trained using K-SVD from 100,000 image patches randomly cropped from 1000 images. The blurriness of an image patch is the measure of its sparseness obtained using an l_0 norm. The sparseness score is further mapped as a blur score for every pixel (the patch center) using a logistic regression. The map is further smoothed using weighted least square estimation [32], such that it can accurately differentiate between smooth and blur regions and prevent artifacts around object boundaries. They essentially involve a smoothness term computed from the horizontal and vertical gradients obtained from the initial map. The amount of smoothing at each pixel is controlled by smoothness weights determined from horizontal and vertical derivatives of the original image.

To train the *artifacts map* network we take as input a compressed image and expect a 256-level target map in the range [0-1]. Zeros indicate regions perfectly suitable for enhancement and ones are regions which have quite large blocking or ringing effects. The target map is the normalized absolute difference between the original image and the compressed image across the three channels, weighted with the gradient image. Each image fed into the deep network outputs an image of the same size, where each pixel is mapped in the range [0-1]. Figure 5 demonstrates some example images taken from the LIVE dataset.

We performed a number of different experiments with architectures as well as datasets to obtain a map which can avoid to enhance JPEG artifacts, and especially ringing. We initially use the Direct Architecture model of [42] to generate images with reduced artifacts from ones with ringing and blocking effects.

We next use these training weights as an initialization to generate our JPEG artifact map. In this second step the input images are the same as before, that is, compressed images with artifacts, while the target images are now the JPEG artifact map. We use the Berkeley Segmentation dataset for both the steps.

The results as seen in Figure 5 demonstrate that the network is indeed successful to analyse regions where artifacts are more prominent. We also analyse differences in maps obtained using the Berkeley Segmentation dataset, the CUHK dataset (also used for blur map generation) and the Single-Capture-Images [43] captured using ten different cameras. In all cases we

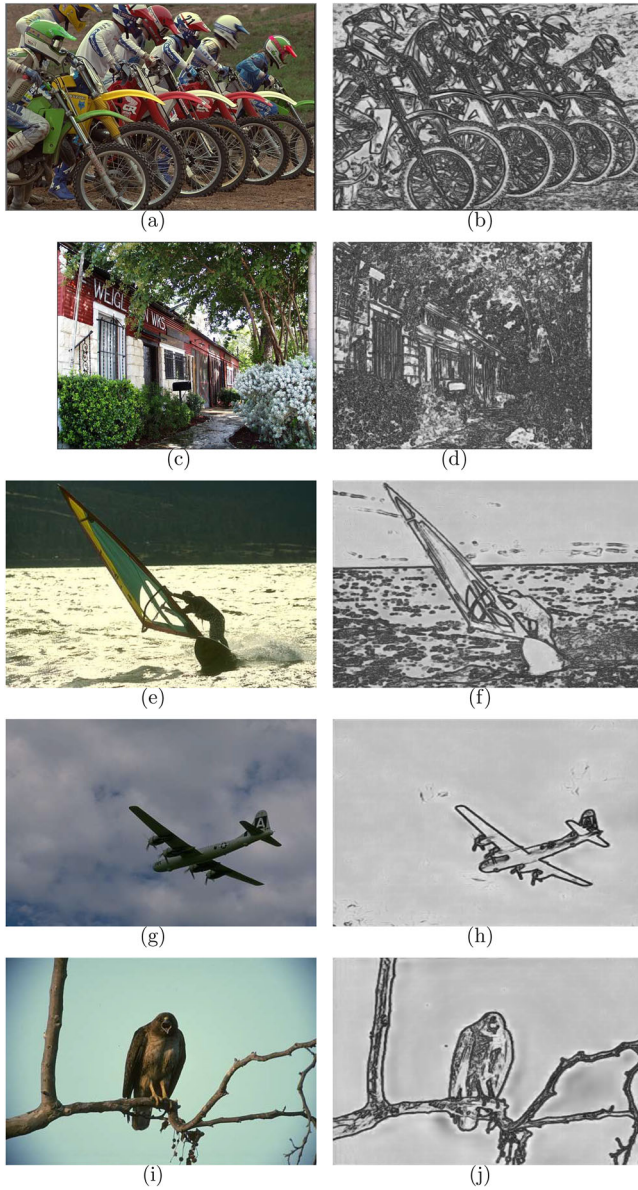


FIGURE 5 Sample compressed images (left) and corresponding ringing maps (right) obtained by the proposed method

observe that results obtained using the deeper network provide better maps.

3.1 | Nonlinear map-based image enhancement

The blur and artifact maps generated by our approach are exploited to perform the controlled sharpening action. Both maps are generated in the range $0 - 1$, where 0 indicates regions to be sharpened and 1 indicates regions which should not be enhanced due to blur (for blur map) or artifacts (for artifact map). Hence both maps can be treated in the same way.

The values of the control parameters are derived from the map data at each image position. A simple pointwise function

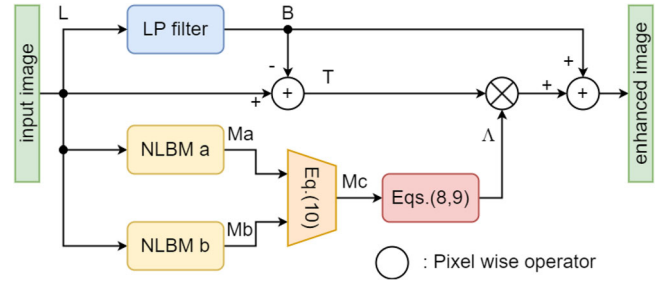


FIGURE 6 Structure of the whole system

can be used for our map. A 2-D function depending also on a local evaluation of the contrast of the image could also be chosen; its principle would be to reduce the sharpening action where the image contrast is high and where visible JPEG artifacts are present. A 2-D function is essential when the map is a binary or a roughly quantized object, that implies an almost on/off operation of the sharpening. The technique we propose, however, has been designed to yield a blur map consisting of blobs having a smoothly varying interior and soft contours: this permits to obtain a gentle variation of the sharpening effect across the main objects borders, which typically already have large contrast and thus generate an undesired overshoot if the blur map switches abruptly. The soft blur map we generate is able to counteract this effect. Moreover, as we shall show below, our map permits to manage pictures in which the blurred region is not precisely defined; this often happens when cheap image acquisition equipment having low-aperture optics is used, or in many open-space scenes, or when it is not desired to show a single portrayed subject standing out in the picture.

The entire sharpening operation is depicted in Figure 6 and described by the simple Equation (7):

$$\tilde{L} = B + \Lambda \odot T, \quad (7)$$

where B and T , respectively, are the *base* and *detail* components of the original image L . If the input is a color image, the operator acts on its luminance component only. B is derived from L by a simple lowpass filter, having, for example, a Gaussian impulse response, and T is the $L - B$ difference. It should be noted that [3] suggests to use also other more sophisticated edge-preserving smoothing filters like the bilateral filter. This is not necessary for NLBM, due to the characteristics of our blur map. Λ is a scaling matrix, the same size as the image, that controls the amount of detail enhancement at each image position via the \odot operator, which is an element-by-element product.

The local sharpening control is straightforwardly provided by Equations (8) and (9):

$$\mathbf{\Lambda} = \rho G_b(\mathbf{M}) + \eta, \quad (8)$$

$$G_b(\mathbf{M}) = e^{-\left(\frac{\mathbf{M}}{\alpha}\right)^\beta}. \quad (9)$$

In these equations, ρ and η (1.5 and 1, respectively, in the cases we show) determine the *global* maximum and minimum

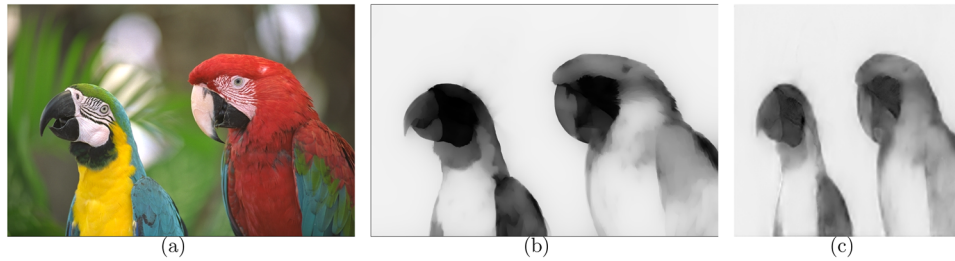


FIGURE 7 Original 2-plane image (a), and the blur maps derived using the BUM method (b) and our NLBM method (c) (without rescaling)

amount of sharpening and can be freely selected by the user. The parameters β and α determine the shape of the exponential function and permit a properly responsive control of the *local* amount of the enhancement; they should be selected according to the characteristics of the image and of the amount of noise that affects it. We used $\beta = 5$ in all the experiments we show. About α , our experiments indicate that a value of 0.4 can be effective in portrait-like pictures, while open-space scenes may be best processed using a value in the range $0.2 \leq \alpha \leq 1$. In all cases, the choice of α is simple and does not require fine tuning (selection steps can have size 0.2 or even larger). \mathbf{M} can be replaced with the combined map \mathbf{M}_c generated using the blur \mathbf{M}_b as well as the artifact \mathbf{M}_a maps. The combined activation map is obtained by the pixel-by-pixel product of the inverses of the blur map and of the artifact map as shown in Equation (10).

$$\mathbf{M}_c = \mathbf{M}_b^{-1} \odot \mathbf{M}_a^{-1}. \quad (10)$$

Even if this aspect is sometimes disregarded in the literature, the best parameter choice also depends on the display and on the visualization context, for example, on the maximum luminance and contrast of the display, on the intensity and spatial distribution of the ambient illumination, and on the viewing distance vs. screen size ratio [33]. As put forward in [34], the quality assessment of an image or video is viewer-dependent. Also, the type of method used for analysing the quality plays an important role. For example, using toggling for comparisons proves more effective than side-by-side or sequential comparisons.

4 | RESULTS

We report in the following on various types of experimental results. In Section 4.1 we compare blur maps obtained by our DL method with concurrent techniques presented in [3], [20] and [30]. We next show enhancement results using our blur map as compared to maps obtained in [3] and [20] in Section 4.2. We consider the enhancement of images affected by visible JPEG artifacts in Section 4.3, and image enhancement comparisons with state-of-the-art low light image enhancement techniques in 4.4. We then make some comments about the feasibility of objective and subjective image quality assessment (IQA) methods in Section 5.1. A brief observation about processing speed

is made in Section 5.2. In Section 5.3, we demonstrate some ablation experiments.

4.1 | Blur maps

We first compare the blur maps we obtain with our method to those derived and used in the BUM method [3]. We start from a 2-plane image, Parrots, Figure 7a. Figure 7b,c shows the two main planes of the scene effectively discerned, even if it is apparent that in both maps the representation of the foreground is dependent not only on the different local depths but also on the presence of fine details. This is not a problem in fact: the result of the sharpening action using either BUM or NLBM is very good, as shown in the next subsection. It should be observed that the resolution anisotropy present in (c) and due to the square CNN shape does not perceptibly affect the results. Figure 8 shows more 2-plane, n-plane, continuous-plane images, and the corresponding maps.

We then compare the blur maps we obtain with the DL method to the ones provided by [20]. We selected images from the CUHK dataset [24] affected either by motion or defocus blur. We remind that we aim at generating graded blur maps, better suited for sharpening operations. The maps indeed are quite different from those presented in [20], and have some properties that prove useful for local UM control. For example, our map for image PushTail (first image in Figure 9) shows the change of focus moving vertically in the box, while in [20] the map is uniform. A gentle management of the focus permits to obtain a pleasantly graded sharpening as shown in the next subsection. Other images with out-of-focus blur as well as motion blur are shown in Figures 9 and 10, respectively.

We further provide comparisons of blur maps using our method and [30] in Figures 11 and 12. As seen from the figure, maps obtained from [30] have four levels of blur compared to the two levels obtained from [20]. Some DCNN based blur maps learnt during the deblurring task is also added in Figures 13 and 14.

4.2 | Enhancement of locally blurred images

Some examples permit to verify the quality of the images enhanced using the NLBM blur map. In summary, the results

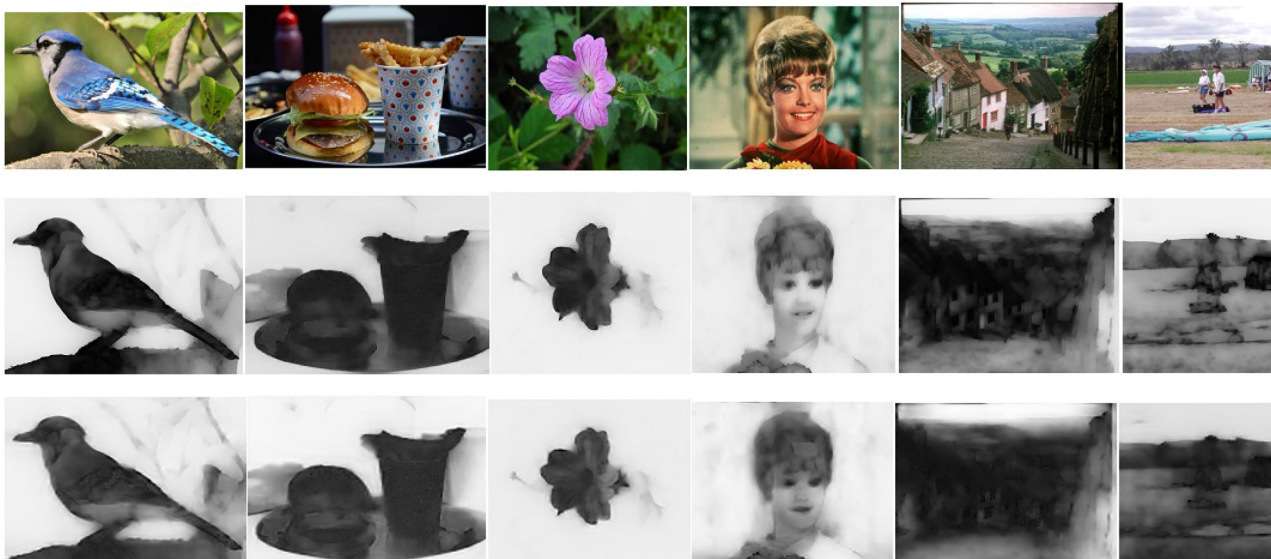


FIGURE 8 Various 2-plane, n-plane and continuous-plane images (top row) and corresponding blur maps derived with the BUM method (middle row) and our NLBM method (bottom row)

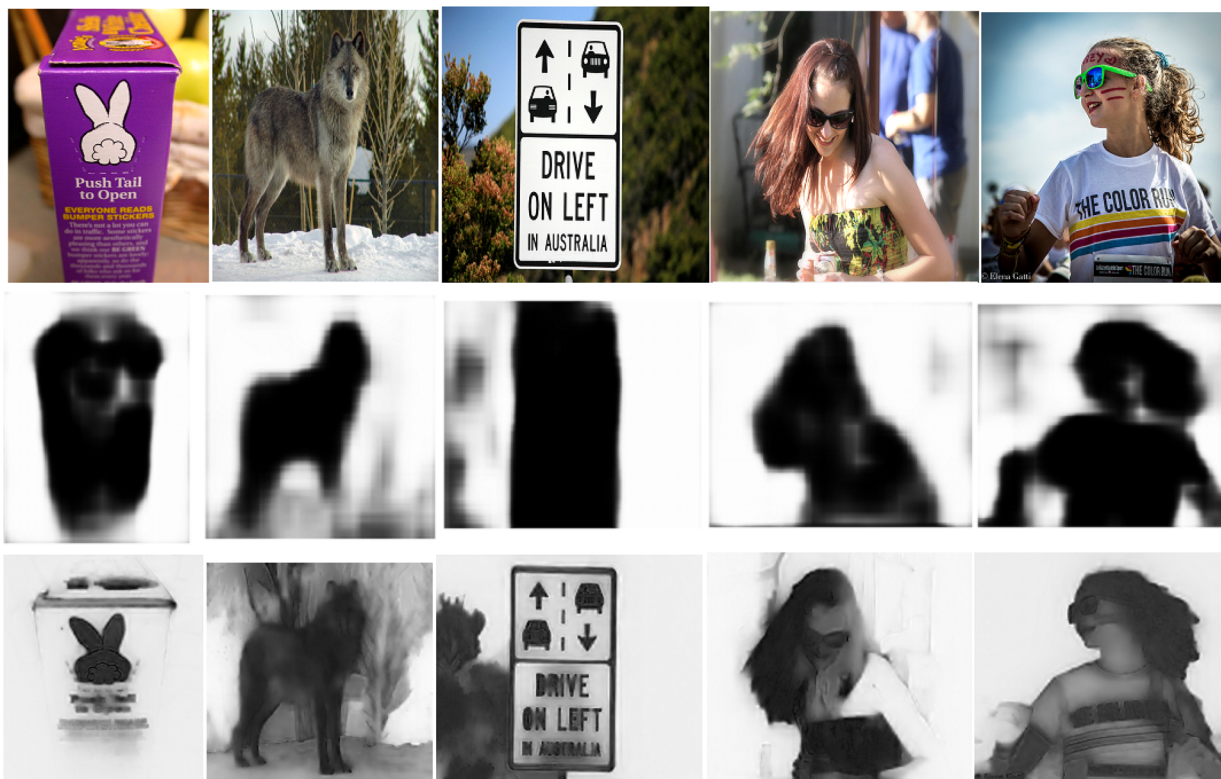


FIGURE 9 Blur maps derived with the DBM [20] method (middle row), and our DL method (bottom row) using sample out-of-focus blurred images f0132, f0374, f0556, f0596 and f0676 from CUHK dataset

provided by NLBM are at least on a par with those by [3], and in many cases have slightly better quality. As clarified below, a large advantage is instead always provided by NLBM in terms of computation speed.

Looking again at Figure 7, it is apparent that in the map (b) provided by [3] the in-focus area has very neat borders, that will yield some overshoots in the main contours of the objects; in our blur map (c), the softer map borders will produce a more

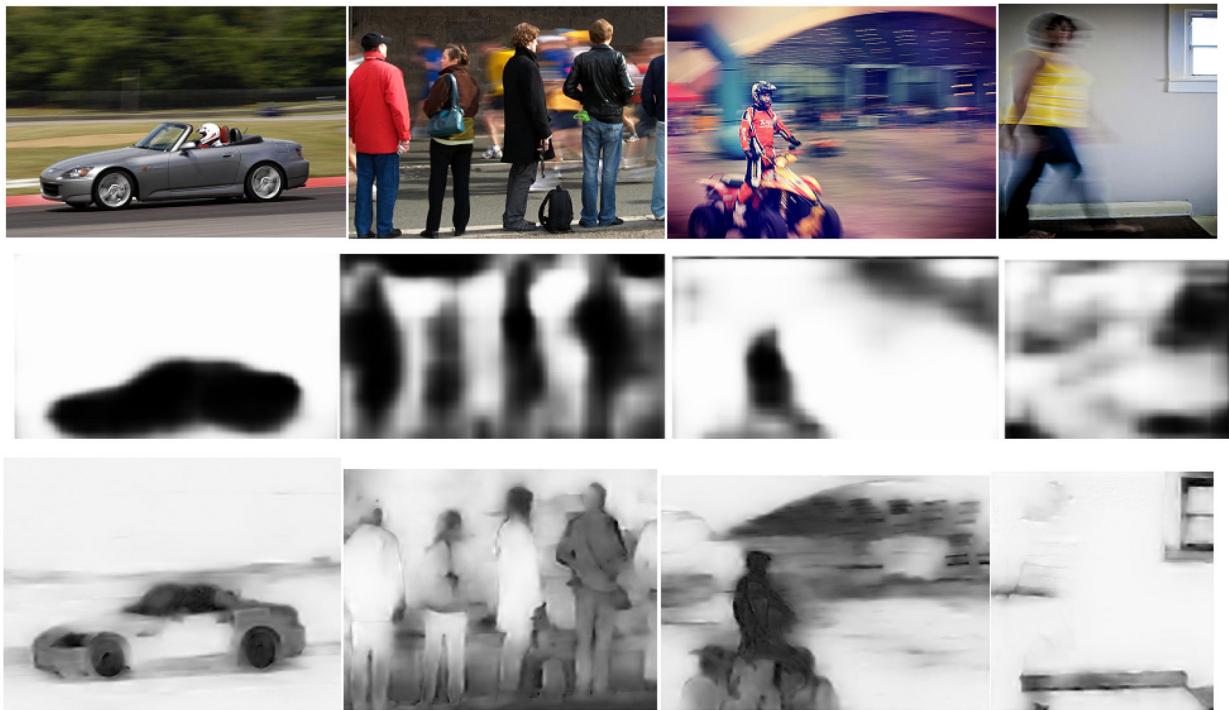


FIGURE 10 Blur maps derived with the DBM [20] method (middle row), and our method (bottom row) using sample motion blur images m0014, m0020 and m0194 from CUHK dataset

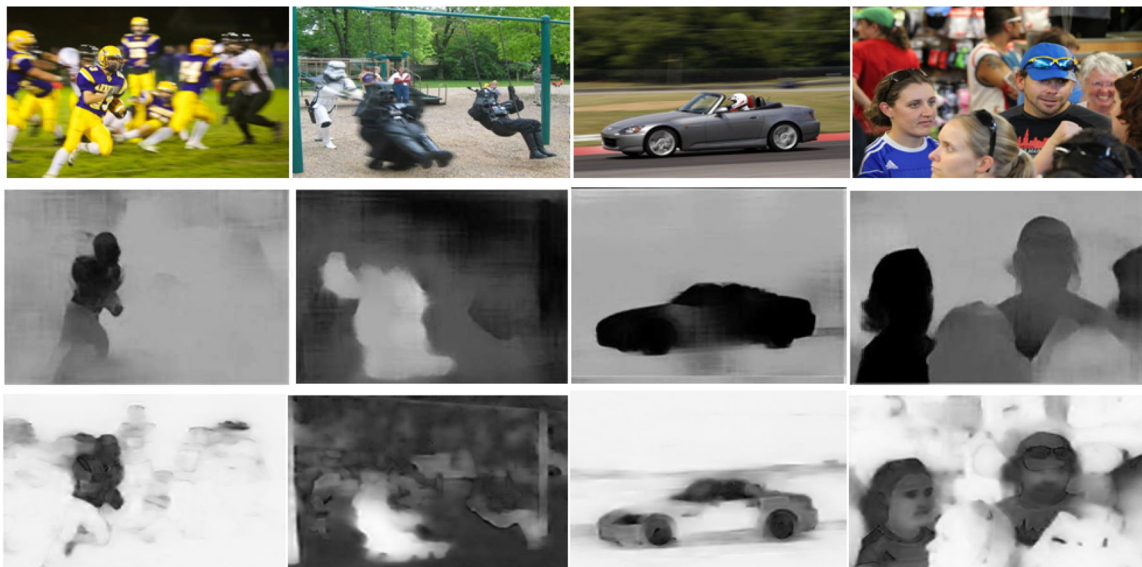


FIGURE 11 Blur maps derived with the method in [30] (middle row), and our method (bottom row) using sample images from CUHK dataset

realistic enhanced image. Indeed Figure 15 shows the image after processing with [3] (top) and with our NLBM (bottom). It is recognizable that the finest details are better sharpened in our case, overshoot effects are not annoying, and noise amplification is avoided in both cases. Results on more images can be found on the Web site we indicate at the end of this paper.

With respect to visual quality it is important to observe that NLBM works well also if the concept of blur is taken more loosely (not only optical but also atmospheric), for example, on images like Goldhill and Kites. Indeed, the blur map that our method provides enables the user to enhance the input image also coping with the continuous-plane blur case mentioned in Section 2.1. The multilevel map activates the sharpening



FIGURE 12 Blur maps derived with the method in [30] (middle row), and our method (bottom row) using sample images from CUHK dataset

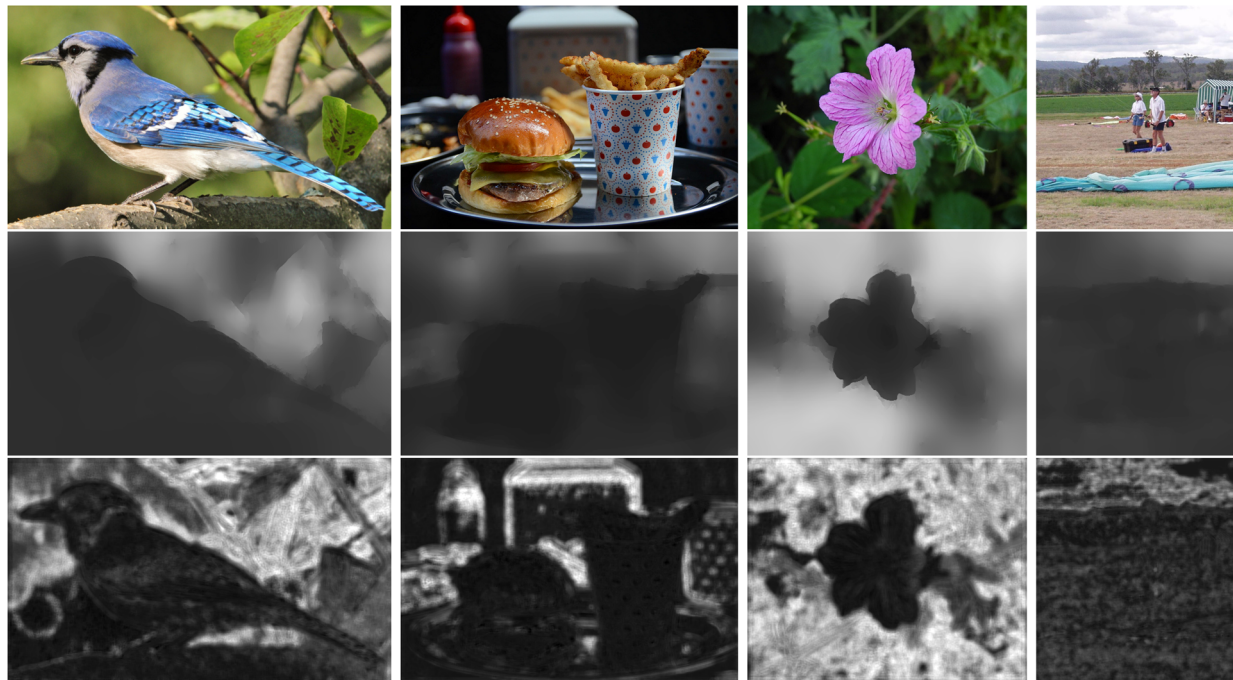


FIGURE 13 Intermediate blur maps extracted from [46] (middle row) and [47] (bottom row) using 2 plane and continuous plane images. While the BENET maps are smoother, the DIDANET maps show distinct edge information

operator progressively, enhancing also partially blurred details up to the level in which no noise amplification is perceived in the smooth areas of the image. This effect is demonstrated for example in the Goldhill test image, where the depth of

field is not enough to adequately represent all the details of the pavement along the road. Figure 16 shows the original image (a) and the results obtained by various sharpening methods: fixed UM (b), BUM (c), and NLBM (d). Our method

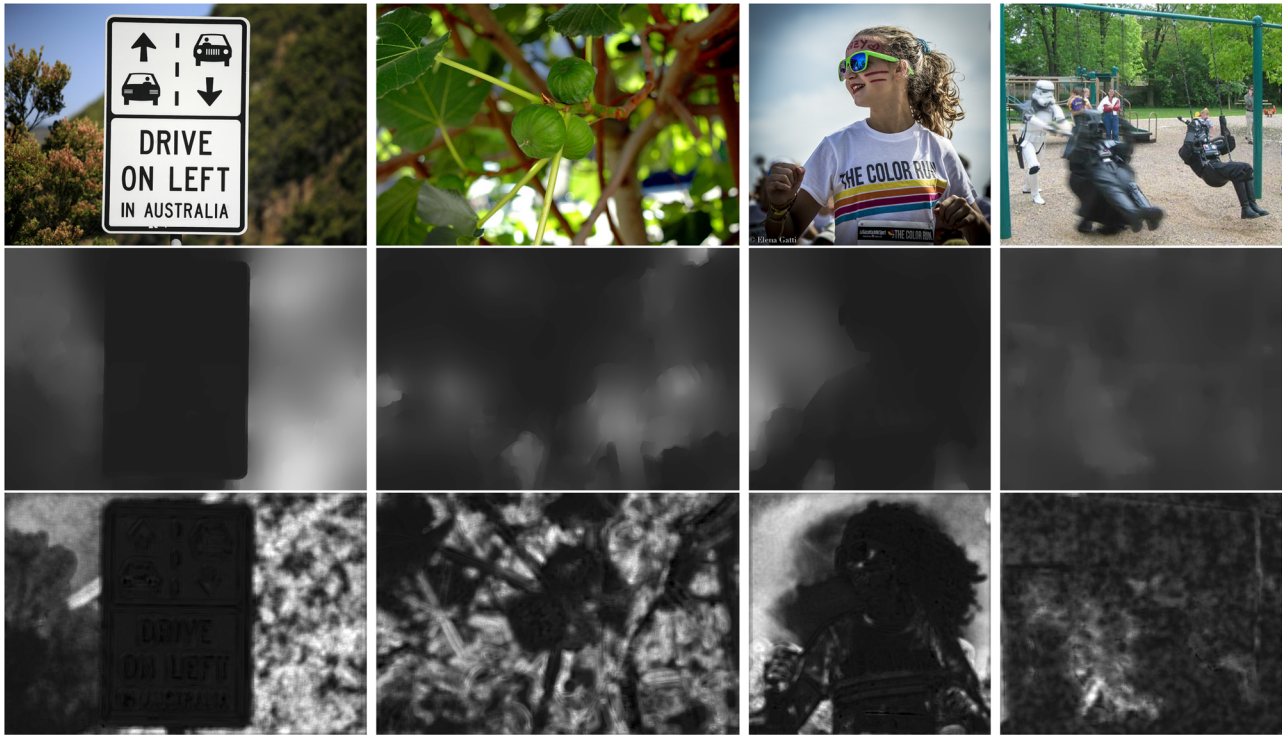


FIGURE 14 Intermediate blur maps extracted from [46] (middle row) and [47] (bottom row) using out-of-focus and motion blur images



FIGURE 15 A zoomed portion of the Parrots image is shown to depict the enhancements obtained using [3] (top) and the proposed NLBM (bottom)

provides a satisfactory sharpening effect everywhere (different parts of the road), better than the one by BUM, and without incurring noise amplification (observe the sky region) as fixed UM does.

In Figures 17 and 18, we demonstrate a few examples of Unsharp Masking application using the map obtained from [20] and our map. Having a uniform map, enhancement using



FIGURE 16 Continuous-plane image enhancement (Goldhill). (a) Original image, (b) Fixed UM, (c) BUM enhancement, (d) NLBM enhancement

maps from [20] (DBM) often introduces noise. For example, in Figure 17 the bottom portion of the car or hair and leg of the person has additional noise in case of DBM (marked by red box). Similarly, the lower part of the PushTail image is out of focus and hence not enhanced by NLBM; enhancement obtained using DBM results instead in unwanted noise in that area (marked by red box). Also, in the third image in Figure 18 DBM enhances only the sign board ignoring the bush which is also in focus.

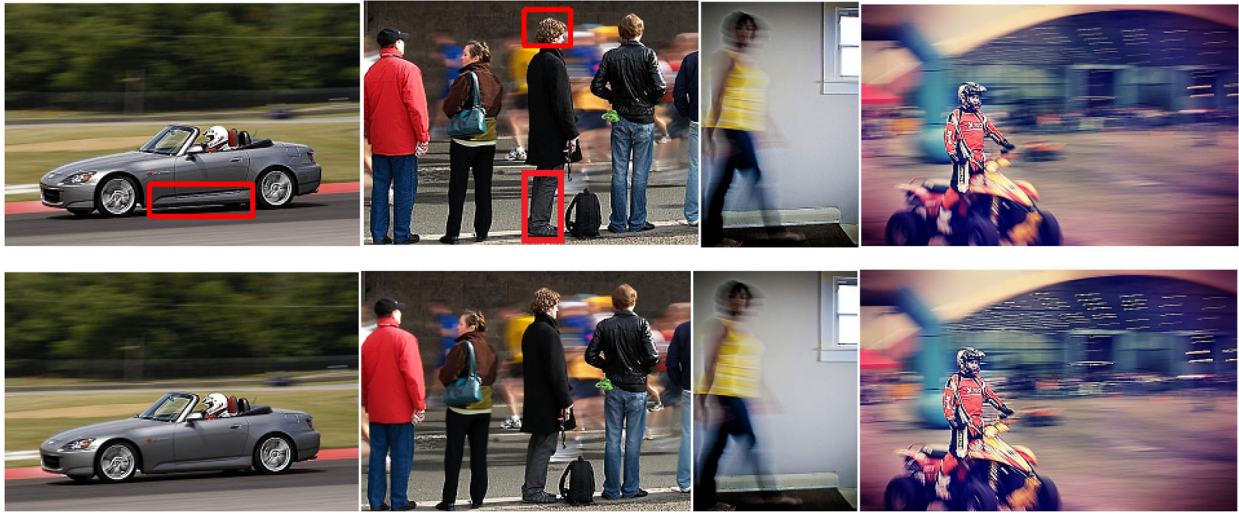


FIGURE 17 Image enhancement using blur maps from [20] (top row) and our blur maps (bottom row)



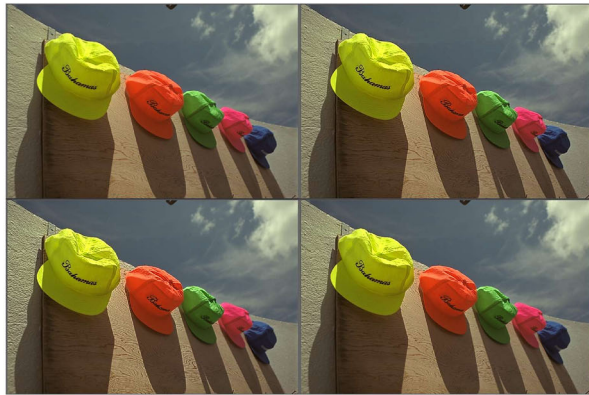
FIGURE 18 Image enhancement using blur maps from [20] (top row) and our blur maps (bottom row)

4.3 | Enhancement of JPEG compressed images

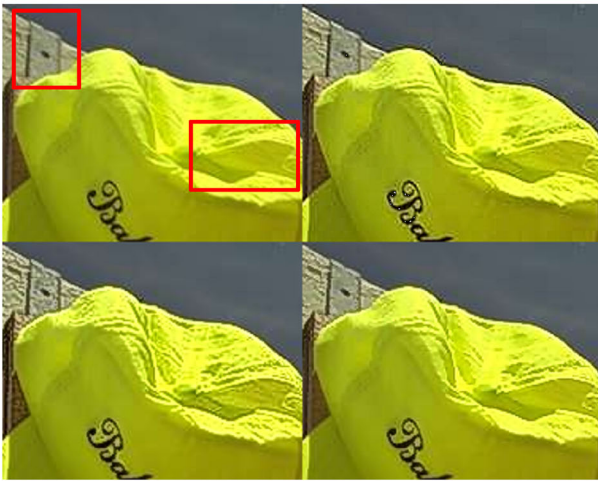
As discussed above, real world images often have visible JPEG artifacts; it is extremely important that they are not emphasized by the UM processing. We obtain this behaviour exploiting our combined activation map, obtained by the pixel-by-pixel product of the inverses of the blur map and of the artifact map as shown in Figure 19. We conduct some experiments to observe its effect: an example is shown in Figure 20. Figure 20a shows the “Caps” image after JPEG compression with $Q=60$ (top-left), and its enhanced versions using a standard UM (top-right), BUM (bottom-left), and NLBM (bottom-right). Even if the results can appear similar at first sight, a closer look permits to notice unpleasant artifact amplification in some cases. In a well focused area (Figure 20b, notice the red boxes in the JPEG original), ringing on the contour of the cap is amplified by UM, while BUM and NLBM avoid its amplification; however, BUM



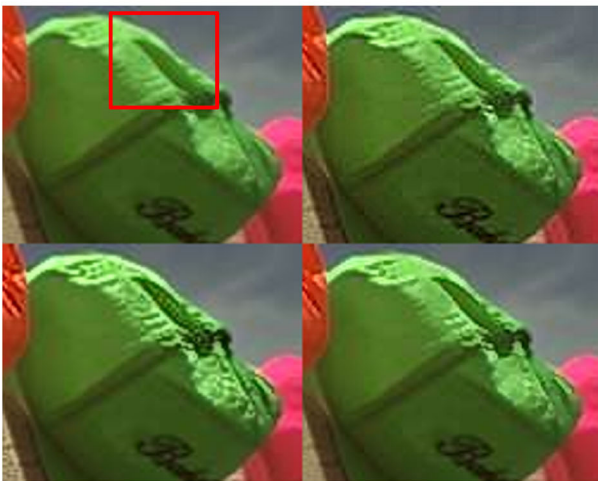
FIGURE 19 JPEG image and corresponding maps. Top-left is “Caps” after JPEG compression with $Q = 60$, top-right is the blur map for the image, bottom-left gives the artifact map while bottom-right gives our combined map used for artifact-reduced enhancement



(a)



(b)



(c)

FIGURE 20 Enhancement of “Caps” (a), and corresponding details (b,c). In each subfigure, top-left is the JPEG original, top-right its standard sharpening (UM), bottom-left the BUM enhanced image, and bottom-right our NLBM generated using the combined map shown in Figure 19

suffers from ringing amplification visible in the wall detail and from blockiness in the soft canvas areas, while NLBM is more faithful to the original data. Similar effects are recognizable in an out-of-focus portion of the same image (Figure 20c, notice the red box in the JPEG original). We stress that it is not our

TABLE 1 BIQA scores for image enhancements using Berkeley Driving Dataset Images

Image	Mean	StD
Original	36.89	3.598
NLBM	33.55	2.471
ZDCE	34.03	2.824
DUPE	33.94	3.578
MBLLEN	42.50	5.238
EGAN	30.69	4.711

purpose to reduce JPEG artifacts: we just aim at avoiding their amplification.

4.4 | Low light image enhancement

Enhancing low light images for better aesthetic quality or image understanding in terms of object detection, face detection etc. is a research area receiving increasing attention. Though the concepts of histogram equalization and of Retinex-based illumination enhancement date back to more than three decades, the more recent popularity of deep learning resulted in many image enhancement networks [52–55]. Each of these networks is trained with a pair of low and bright light images, and when provided with a relatively dark image as input it generates an enhanced brighter image. It should also be noted that enhancing low light images amplifies the noise and JPEG artifacts and is hence a complex problem which in most cases will not be handled effectively if simple gamma corrections are used. We have applied NLBM on low light images to emphasise the fact that gamma correction on an NLBM enhanced low light image provides acceptable performance. Indeed, NLBM is able to enhance edges, without amplifying blockiness in defocused areas. We pick moderately low light scenes from the Berkeley Driving dataset and generate outputs for each of the methods as shown in Figure 21. It is observed (Table 1) that NLBM provides visually better images as well as better BIQA scores compared to [52, 53, 55]. EGAN, which is specifically designed for overall quality improvement, has a better BIQA average but its standard deviation is poorer, meaning it may fail for some images.

5 | DISCUSSION

5.1 | Objective and subjective quality assessment

For the quantitative evaluation of enhancement methods a reference ideal image does not exist in principle. No-reference, also called Blind, image quality assessment (BIQA) tools have to be used, such as DB-CNN [35] and BRISQUE [36]. These tools have been tested on normal, blurred, noisy and compressed images and show generally reliable results. However, it is quite



FIGURE 21 Enhancement results using different low light enhancement techniques on Berkeley Driving Dataset Images. Top row shows Original, MBLLEN, NLBM from the left while bottom row shows DUPE, ZDCE, EGAN from the left

tricky to use the same for analysing images the details of which have been sharpened, since in some cases they provide good scores even when the subjective quality is not satisfactory, for example, in the presence of noise or of over-enhanced contours. The images in question have a certain amount of out-of-focus blur which is left as is by enhancing techniques like our NLBM, [20], and [3]. Among such three techniques, DB-CNN does mark images processed by our method to have a better quality. However, the results are not considered in this paper since in some cases the tool marks the general sharpening method to provide the highest quality notwithstanding noise amplification, while in other cases it marks the original image as a better quality image compared to any kind of processing.

The BRISQUE method on the other side did not seem suited to estimate the quality of fine details, probably because it is designed to mainly measure the “naturalness” of an image.

Other methods which are not suitable for our case are those based on local phase coherence [37], or S3 [38]: in fact, they measure the amount of blurriness in an image rather than the sharpening effect obtained by an operator.

The situation is more manageable if the image to be enhanced contains compression artifacts. Conventional sharpening methods tend to make such artifacts more visible, and BIQA tools

TABLE 2 BIQA scores (using DB-CNN)

Image	NLBM	UM	BUM	DBM
Ship	18.19	19.92	19.14	19.42
Bikes	24.41	25.22	25.05	25.04
Building	27.29	28.25	28.16	28.17
Airplane	38.29	39.71	39.04	39.40
Bird	32.60	33.71	32.85	33.33

sense the resulting reduced visual quality. Tables 2 and 3 provide the BIQA scores (provided by DB-CNN) of some compressed image samples with slightly visible JPEG artifacts, enhanced using UM, BUM, DBM, and NLBM. It can be seen that our results are better compared to all other techniques. Further, in Table 3 it can be observed that even when we increase the amount of sharpening in the image the artifact map can successfully avoid to enhance ringing and blocking effects.

We also setup an experiment to collect data about subjective preferences, complying with the ITU-R guidelines [39]. We used a Panasonic professional full-HD LCD display, with maximum luminance 350 cd/m² and 2% antiglare coating, in a room with

TABLE 3 BIQA scores with stronger sharpening

ρ	Ship	Ship	Bikes	Bikes
	NLBM	UM	NLBM	UM
0.5	18.19	19.92	24.41	25.22
1.0	18.64	20.77	25.31	26.86
1.5	19.52	22.16	25.96	28.15
2.0	20.28	24.13	26.39	28.99
2.5	21.48	26.17	26.99	29.75
3.0	22.11	27.41	27.43	30.19

TABLE 4 Scores obtained using visual quality

Image	Original	UM	DBM	BUM	NLBM
Bird	0	0.17	0.05	0.17	0.61
Burger	0	0.11	0.05	0	0.84
Flower	0	0.17	0.05	0.11	0.67
Goldhill	0.05	0.17	0.11	0	0.67
KiteS	0.05	0.17	0.11	0	0.67
Zelda	0	0.11	0.05	0	0.84
f0132	0.17	0.11	0	0.05	0.67
f0374	0.11	0.05	0.11	0.05	0.68
f0556	0.17	0.11	0	0.05	0.67
f0596	0.17	0.05	0.11	0.17	0.50
f0676	0.17	0.17	0	0.05	0.61
m0014	0	0	0.11	0.05	0.84
m0020	0	0	0.11	0.05	0.84
m0194	0	0	0.11	0.05	0.84

low and diffused illumination (a few hundred lux). A group of 18 users were asked to assign rank 1 to the best image and 0 to all other images. The observers had to stay in front of the screen but could freely change their distance during each trial before making their selection. Five different variations of each image were displayed: the original image with JPEG quality 60, and four processed versions using [3], [15], standard Unsharp Masking and NLBM, respectively. The user was permitted to examine all five versions of each image together, or in pairs, or each image individually, and to switch back and forth among the various versions. There were no time constraints. Each image was shown three times using random orderings for the placement of its versions. Notwithstanding this, all users always chose the same version of a given image all three times. The average score obtained for each image is shown in Table 4. It is observed that most users preferred the image processed using NLBM. However, in some cases a few of them preferred no processing at all, or other types of UM. Particularly, it can be observed that for natural scenes such as Goldhill or Kites and for out-of-focus blur images (images f0132-f0676 from CUHK dataset), users sometimes preferred the original compared to any kind of processing. In case of motion blur (images m0014-m0194

from CUHK dataset)), the users very strongly inclined towards NLBM. All the images we used, original and processed versions, have been uploaded in the Web site indicated below.

5.2 | Processing speed

In terms of computational load we observe that the method we propose is much faster than non-NN based techniques. NLBM takes on average 0.3 seconds per image on a core I5 CPU in a Python (PyTorch) environment that was not optimized for speed. We should also consider the overhead time for the uploading of the blur map generation network, that in our case was 3.8 seconds; since map uploading has to be done once for each processing session, this overhead should be divided by the total number of treated images in a work session.

Our method also paves the way to sophisticated enhancement techniques. Even real-time video data could be addressed [40], possibly using suitable architectures like the one in [41]. In particular, the CNN structure is well suited to code parallelization. Conversion of the current architecture in ONNX format or TensorRT format for speed optimization on low power devices will further make the technique suitable for next-generation applications requiring high throughput and low latency.

5.3 | Ablation study

We observe the performance of the NLBM while using different backbone architectures and loss functions. The results shown so far utilize VGG-16 as backbone architecture. We further demonstrate the use of Alexnet and ResNet-18 for the purpose. Also, the network was trained with a weighted L_2 norm between the target and generated blur map. Additionally, the use of a smooth absolute distance criterion to prevent exploding gradients is involved. Equation (11) is substituted with Equation (12) when the absolute difference is less than 1.

$$L_s = \sum_i |x_i - y_i| - 0.5, \quad (11)$$

$$L_s = \sum_i 0.5(x_i - y_i)^2. \quad (12)$$

In these equations, x_i is the estimated output, whereas y_i is the target. While the L_2 and L_s loss functions measure a pixel to pixel loss, a global loss in terms of perceptual or contextual loss [44] also proved to be beneficial for image quality measurement during style transfer training. To exploit this in our case, the loss is measured on the final image obtained using the blur map.

$$L_c = -\log(CX(\sigma_x, \sigma_y)), \quad (13)$$

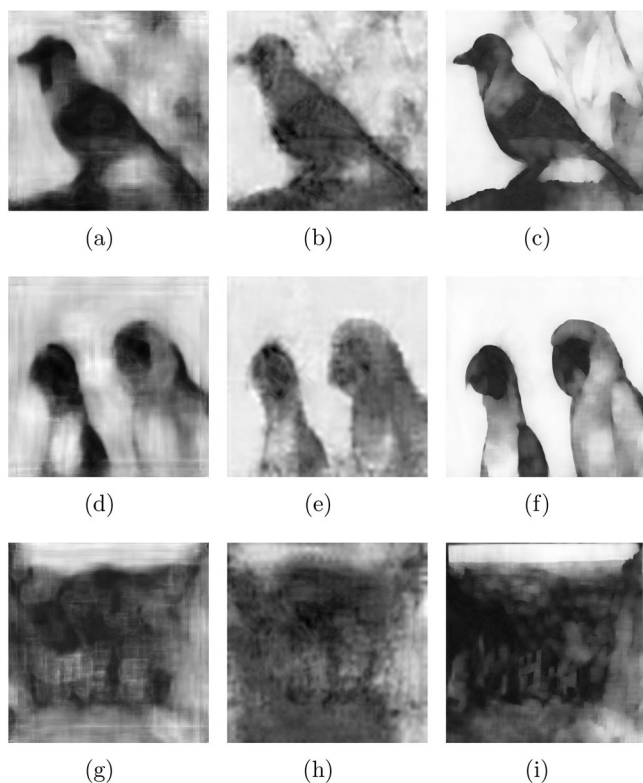
where σ_x and σ_y are VGG-16 features from the bottleneck layer, CX is the contextual similarity between the enhanced images obtained from the target map and from the generated map. We take 12 sample images which include Bird, Burger, Goldhill, Flower and 8 more from the CUHK dataset and measure

TABLE 5 BIQA scores (average and deviation) while using different Loss function for training

Backbone	Loss	Mean	StD
Alexnet	L_v+L_2	31.77	7.267
Alexnet	$L_v+L_2+L_c$	31.83	7.252
Alexnet	L_2	31.69	7.075

TABLE 6 BIQA scores (average and deviation) while using different backbone architectures for training

Backbone	Loss	Mean	StD
Alexnet	L_2	31.69	7.075
Resnet18	L_2	31.54	7.325
VGG-16	L_2	31.93	7.232

**FIGURE 22** The first column uses nonlinear resampling without the feature processor. The second column uses nonlinear resampling with a single feature processor block. The last column uses nonlinear resampling with multiple feature processor blocks

the BIQA scores after enhancing them as shown in Tables 5 and 6. It is observed that the backbone architecture does not provide much difference in the scores as long as multiple feature processing blocks with a nonlinear convolutional upsampling is utilized.

To further demonstrate the effect of these blocks and the nonlinear convolutional upsampling we have provided the obtained blur maps of a few selected images for specific

architecture modifications in Figure 22. It is observed that multiple feature processing blocks, along with the nonlinear CNN resampling, provides the visually best results.

6 | CONCLUSIONS

We present a deep learning-based blur map generation technique and we use it for blurriness-guided image enhancement. It is observed that the blur maps obtained are much more detailed when compared to concurrent techniques [20], [30]. The enhancement obtained using our technique is similar and in some cases better than those obtained in [3]. The map generation (and thus the overall process) is much faster and stable irrespective of the input image size.

One important factor to be considered in future research is the effective input image resolution, which plays an important role in selecting the size of the support for all the local operators in the algorithm. On a longer term, the flexibility of our approach could be further exploited: indeed, the blur map can be used not only together with basic UM techniques but also with any other enhancement method that needs to adapt its behaviour to the amount of local blurriness in an image. As already mentioned, video enhancement could be addressed too.

Original and processed versions of the images we show in this paper, together with more examples, are available for a more accurate inspection at <https://www.units.it/ipl/NLBM22>.

AUTHOR CONTRIBUTIONS

Jhiliik Bhattacharya: Conceptualization, formal analysis, investigation, resources, software, validation, writing - original draft. Stefano Marsi: Conceptualization, formal analysis, investigation, software, validation, writing - review and editing. Giovanni Ramponi: Conceptualization, methodology, validation, visualization, writing - original draft, writing - review and editing.

ACKNOWLEDGEMENTS

G.R. acknowledges the support of the University of Trieste and J.B. acknowledges partial funding support from DST, India (SEED/TIDE/2018/82).

CONFLICT OF INTEREST

The authors declare that they do not have any conflict of interest.

DATA AVAILABILITY STATEMENT

The data and code used in this work can be downloaded from <https://github.com/jhiliikb/NLBM>

ORCID

Jhiliik Bhattacharya  <https://orcid.org/0000-0002-9341-8582>

REFERENCES

- Mather, G.: The use of image blur as a depth cue. *Perception* 26(9), 1147–1158 (1997)

2. Qi, X., Shi, J., Liu, S., Liao, R., Jia, J.: Semantic segmentation with object clique potential. In: *IEEE International Conference on Computer Vision*, pp. 2587–2595. IEEE, Piscataway (2015)
3. Ye, W., Ma, K.-K.: Blurriness-guided Unsharp Masking. *IEEE Trans. Image Process.* 27(9), 4465–4477 (2018)
4. Wang, Z., Bovik, A.C.: Modern image quality assessment. *Synth. Lect. Image Video Multim. Process.* 2(1), 1–156 (2006)
5. Gao, F., Wang, Y., Li, P., Tan, M., Yu, J., Zhu, Y.: Deepsim: deep similarity for image quality assessment. *Neurocomputing* 257, 104–114 (2017)
6. Gao, F., Li, Z., Yu, J., Yu, J., Huang, Q.: Style-adaptive photo aesthetic rating via convolutional neural networks and multi-task learning. *Neurocomputing* 395, 247–254 (2020)
7. Dong, Z., Tian, X.: Multi-level photo quality assessment with multi-view features. *Neurocomputing* 168, 308–319 (2015)
8. Lu, W., Zeng, K., Tao, D., Yuan, Y., Gao, X.: No-reference image quality assessment in contourlet domain. *Neurocomputing* 73(4–6), 784–794 (2010)
9. Li, Y., Po, L.-M., Xu, X., Feng, L., Yuan, F., Cheung, C.-H., Cheung, K.-W.: No-reference image quality assessment with shearlet transform and deep neural networks. *Neurocomputing* 154, 94–109 (2015)
10. Dardi, F., Abate, L., Stessen, J., Ramponi, G.: Causes and subjective evaluation of blurriness in video frames. *Sig. Proc.: Image Comm.* 28(3), 209–221 (2013)
11. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards real-time object detection with region proposal networks. In: *Advances in Neural Information Processing Systems*, pp. 91–99. MIT Press, Cambridge (2015)
12. He, K., Sun, J., Tang, X.: Guided image filtering. *IEEE Trans. Patt. Anal. Mach. Intell.* 35(6), 1397–1409 (2013)
13. Tomasi, C., Manduchi, R.: Bilateral filtering for gray and color images. In: *International Conference on Computer Vision*, pp. 839–846. IEEE, Piscataway (1998)
14. Farbman, Z., Fattal, R., Lischinski, D., Szeliski, R.: Edge-preserving decompositions for multi-scale tone and detail manipulation. *ACM Trans. Graph.* 27(3), 67 (2008)
15. Ramponi, G., Strobel, N.K., Mitra, S.K., Yu, T.-H.: Nonlinear Unsharp Masking methods for image contrast enhancement. *J. Electron. Imag.* 5(3), 353–366 (1996)
16. Ramponi, G.: A cubic Unsharp Masking technique for contrast enhancement. *Signal Process.* 67(2), 211–222 (1998)
17. Polesel, A., Ramponi, G., Mathews, V.J.: Image enhancement via adaptive Unsharp Masking. *IEEE Trans. Image Process.* 9(3), 505–510 (2000)
18. Kim, S.H., Allebach, J.P.: Optimal unsharp mask for image sharpening and noise removal. *J. Electron. Imag.* 14(2), 005–023 (2005)
19. Shi, J., Xu, L., Jia, J.: Just noticeable defocus blur detection and estimation. In: *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 657–665. IEEE, Piscataway (2015)
20. Ma, K., Fu, H., Liu, T.: Deep blur mapping: exploiting high-level semantics by deep neural networks. *IEEE Trans. Image Process.* 27(10), 5155–5166 (2018)
21. Wang, Z., Simoncelli, E.P.: Local phase coherence and the perception of blur. In: *Advances in Neural Information Processing Systems*, pp. 1435–1442. MIT Press, Cambridge (2004)
22. Levin, A.: Blind motion deblurring using image statistics. In: *Advances in Neural Information Processing Systems*, vol. 19, p. 841. MIT Press, Cambridge (2006)
23. Lin, H.T., Tai, Y.-W., Brown, M.S.: Motion regularization for matting motion blurred objects. *IEEE Trans. Patt. Anal. Mach. Intell.* 33(11), 2329–2336 (2011)
24. Shi, J., Xu, L., Jia, J.: Discriminative blur detection features. In: *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2965–2972. IEEE, Piscataway (2014)
25. Liu, R., Li, Z., Jia, J.: Image partial blur detection and classification. In: *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8. IEEE, Piscataway (2008)
26. Zhang, W., Bergholm, F.: Multi-scale blur estimation and edge type classification for scene analysis. *Int. J. Comput. Vision* 24(3), 219–250 (1997)
27. Chakrabarti, A., Zickler, T., Freeman, W.T.: Analyzing spatially-varying blur. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 2512–2519. IEEE, Piscataway (2010)
28. Park, J., Tai, Y.-W., Cho, D., Kweon, I.S.: A unified approach of multi-scale deep and hand-crafted features for defocus estimation. *arXiv preprint 1704.08992*, (2017)
29. Levin, A., Lischinski, D., Weiss, Y.: A closed-form solution to natural image matting. *IEEE Trans. Pattern Anal. Mach. Intell.* 30(2), 228–242 (2008)
30. Zhang, S., Shen, X., Lin, Z., Měch, R., Costeira, J.P., Moura, J.M.F.: Learning to Understand Image Blur. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6586–6595. IEEE, Piscataway (2018)
31. Yu, N., Shen, X., Lin, Z., Mech, R., Barnes, C.: Learning to detect multiple photographic defects. *arXiv preprint 1612.01635*, (2016)
32. Lischinski, D., Farbman, Z., Uyttendaele, M., Szeliski, R.: Interactive local adjustment of tonal values. *ACM Trans. Graph.* 25(3), 646–653 (2006)
33. Kundu, D., Ghadiyaram, D., Bovik, A.C., Evans, B.L.: No-reference quality assessment of tone-Mapped HDR pictures. *IEEE Trans. Image Process.* 26(6), 2957–2971 (2017)
34. Allison, R.S., Brunnström, K., Chandler, D.M., Colett, H.R., Corriveau, P.J., Daly, S., Goel, J., Long, J.Y., Wilcox, L.M., Yaacob, Y.M., Yang, S.-H., Zhang, Y.: Perspectives on the definition of visually lossless quality for mobile and large format displays. *J. Electron. Imaging* 27(5), (2018)
35. Zhang, W., Ma, K., Yan, J., Deng, D., Wang, Z.: Blind Image Quality assessment using a deep bilinear convolutional neural network. *IEEE Trans. Circuits Syst. Video Technol. (TCSVT)* 30(1), 36–47 (2020)
36. Mittal, A., Moorthy, A.K., Bovik, A.C.: No-reference image quality assessment in the spatial domain. *IEEE Trans. Image Process.* 21(12), 4695–4708 (2012)
37. Wang, Z., Simoncelli, E.P.: Local phase coherence and the perception of blur. In: *Advances in Neural Information Processing Systems*, vol. 16. MIT Press, Cambridge, MA (2004)
38. Vu, C.T., Chandler, D.M.: S3: A spectral and spatial sharpness measure. In: *2009 First International Conference on Advances in Multimedia*, pp. 37–43. IEEE, Piscataway (2009)
39. Recommendation 500-13: Methodology for the subjective assessment of the quality of television pictures. ITU-R Rec. BT.500 (2012)
40. Jiang, X., Sun, J., Li, C., Ding, H.: Video image defogging recognition based on recurrent neural network. *IEEE Trans. Ind. Informatics* 14(7), 3281–3288 (2018)
41. Huang, S., Chen, W.: A new hardware-efficient algorithm and reconfigurable architecture for image contrast enhancement. *IEEE Trans. Image Process.* 23(10), 4426–4437 (2014)
42. Albluwi, F., Krylov, V., Dahyot, R.: Artifacts reduction in JPEG-compressed images using CNNs. *Irish Machine Vision and Image Processing Conference*. IEEE, Piscataway (2018)
43. Thongkamwitoon, T., Muammar, H., Dragotti, P.L.: An image recapture detection algorithm based on learning dictionaries of edge profiles. *IEEE Trans. Inf. Forensics Security* 10(5), 953–968 (2015)
44. Mechrez, R., Talmi, I., Zelnik-Manor, L.: The contextual loss for image transformation with non-aligned data. *arXiv preprint 1803.02077* (2018)
45. Zhao, W., Zhao, F., Wang, D., Lu, H.: Defocus blur detection via multi-stream bottom-top-bottom network. *IEEE Trans. Pattern Anal. Mach. Intell.* 42(8), 1884–1897 (2020)
46. Karaali, A., Harte, N., Jung, C.R.: Deep multi-scale feature learning for defocus blur estimation. *IEEE Trans. Image Process.* 31, 1097–1106 (2022)
47. Ma, H., Liu, S., Liao, Q., Zhang, J., Xue, J.-H.: Defocus image deblurring network with defocus map estimation as auxiliary task. *IEEE Trans. Image Process.* 31, 216–226 (2022)
48. Mahmood, M.T.: Defocus blur segmentation using genetic programming and adaptive threshold. *Comp. Mater. Continua* 70(3), 4867–4882 (2022)
49. Ali, U., Mahmood, M.T.: Defocus blur segmentation using local binary patterns with adaptive threshold. *Comp. Mater. Continua* 71(1), 1597–1611 (2022)
50. Basar, S., Waheed, A., Ali, M., Zahid, S., Zareei, M., Biswal, R.R.: An efficient defocus blur segmentation scheme based on hybrid LTP and PCNN. *Sensors* 22(7), 2724 (2022)

51. Marsi, S., Bhattacharya, J., Molina, R., Ramponi, G.: A non-linear convolution network for image processing. *Electronics* 10, 201 (2021)
52. Lv, F., Lu, F., Wu, J., Lim, C.: MBLEN: low-light image/video enhancement using CNNs. In: *British Machine Vision Conference*. Springer, London (2018)
53. Wang, R., Zhang, Q., Fu, C., Shen, X., Zheng, W-S., Jia, J.: Underexposed photo enhancement using deep illumination estimation. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 6842–6850. IEEE, Piscataway (2019)
54. Jiang, Y., Gong, X., Liu, D., Cheng, Y., Fang, C., Shen, X., Yang, J., Zhou, P., Wang, Z.: EnlightenGAN: deep light enhancement without paired supervision. *IEEE Trans. Image Process.* 30, 2340–2349 (2021)
55. Guo, C., Chongyi, L., Jichang, G., Loy, C. C., Hou, J., Kwong, S., Cong, R.: Zero-reference deep curve estimation for low-light image enhancement.

In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1777–1786. IEEE, Piscataway (2020)

How to cite this article: Bhattacharya, J., Marsi, S., Ramponi, G.: Nonlinear kernel based feature maps for blur-sensitive unsharp masking of JPEG images. *IET Image Process.* 17, 1010–1028 (2023).
<https://doi.org/10.1049/ipr2.12692>