

Benchmark for Adaptive Edge-Preserving Image Smoothing

J Faheemunnisa bi¹, B. Pushpalatha², Dr.S.A.Siva kumar³

¹P.G. Scholar, ²Guide, Assistant Professor, ³Head of the Department

^{1,2,3} BRANCH : DECS department of ECE

^{1,2,3} Dr.K.V.Subba Reddy College Of engineering for women

EMAIL:¹sjaveed87@gmail.com, ²pushpa.be90@gmail.com

Abstract

Edge-preserving image smoothing is an important step for many low-level vision problems. Though many algorithms have been proposed, there are several difficulties hindering its further development. First, most existing algorithms cannot perform well on a wide range of image contents using a single parameter setting. Second, the performance evaluation of edge preserving image smoothing remains subjective, and there is a lack of widely accepted datasets to objectively compare the different algorithms. To address these issues and further advance the state of the art, in this paper, we propose a benchmark for edge-preserving image smoothing. This benchmark includes an image dataset with ground truth image smoothing results as well as baseline algorithms that can generate competitive edge-preserving smoothing results for a wide range of image contents. A novel procedure for this problem is proposed based on local linear kernel smoothing, in which local neighbourhoods are adapted to the local smoothness of the surface measured by the observed data. The procedure can therefore remove noise correctly in continuity regions of the surface, and preserve discontinuities at the same time. Since an image can be regarded as a surface of the image intensity function and such a surface has discontinuities at the outlines of objects, this procedure can be applied directly to image denoising. Numerical studies show that it works well in applications, compared to some existing procedures.

Keywords: Edge-preserving smoothing, benchmark, image dataset, deep convolutional networks.

Introduction

Images and videos have become an integral part of our life in recent times. Applications now extend from more general documentation of an event and visual communication to more serious surveillance and medical fields. This has raised the massive demand for images with high accuracy and visual quality. However, digital images captured by modern cameras often get corrupted by noise at the time of image acquisition (digitization) and/or transmission. This form of corruption may result in degradation of visual appearance of an image. The efficiency of imaging sensors is affected by a number of factors, such as environmental conditions during image acquisition and by the quality of the sensing elements themselves. For example, in acquiring images with a CCD camera, sensor temperatures and light levels are major factors that can affect the amount of noise in the resulting image. The corruption in images may also occur during transmission. Reason being the interference in the channel used for transmission. For instance, an image transmitted through a wireless medium might be corrupted due to lighting effects or other atmospheric disturbance. Image denoising is a

well explored topic in the field of image processing where the prime objective is to improve the visual quality of an image by reducing noise from its given noisy version. Numerous image denoising techniques have been developed to minimize the effect of noise(s) occurred due to any of the above mentioned noise sources. A major challenge is to preserve the image details and local geometries while removing the undesirable noise.

In many image analysis and manipulation tasks, such as contour detection, image segmentation, and image stylization, it is important to preserve major image structures, such as salient edges and contours, while smoothing insignificant details. This can be achieved by edge-preserving image smoothing, a fundamental problem in image processing and low-level computer vision. Though a number of algorithms with diverse design philosophies have been proposed [1]–[12], there exist three problems that hinder the further development of edge-preserving image smoothing algorithms. First, the performance evaluation of edge-preserving smoothing algorithms remains subjective. At present, the prevailing method is visual inspection by subjects on the smoothed images. Such an approach is time-consuming and cannot be applied in automatic systems. There lacks an objective metric to evaluate the edge-preserving smoothing algorithms.

Second problem is that an edge-preserving smoothing algorithm is typically evaluated on a very small image set against other algorithms. There lacks a widely accepted large-scale image database for algorithm evaluation. While a smoothing algorithm produces impressive results on certain types of images, it may not perform well on other types of images. Thus, a large database for a holistic evaluation of edgepreserving smoothing algorithms is much needed. Third, smoothing algorithms typically have tunable parameters and images with different categories of contents need different parameter settings. To the best of our knowledge, no smoothing algorithms can perform reasonably well on a wide range of image contents using a single parameter setting. To address the aforementioned problems, in this paper we propose a benchmark for edge-preserving image smoothing. This benchmark includes an image dataset with “groundtruth” image smoothing results as well as baseline models that are capable of generating reasonable edge-preserving smoothing results for a wide range of image contents. Our image dataset contains 500 training and testing images with a number of visual object categories, including humans, animals, plants, indoor scenes, landscapes and vehicles. The groundtruth smoothing results in our dataset are not directly generated by handcraft approaches, but manually chosen from results generated by existing state-of-the-art edge-preserving smoothing algorithms. This is justified by two reasons. First, as discussed earlier, a single state-of-the-art smoothing algorithm is capable of producing high-quality smoothing results over a small range of image contents especially when its parameters have been fine-tuned. Therefore, a collection of smoothing algorithms are able to generate high-quality results over a wide range of contents. The only caveat is that the best results generated by these algorithms for a specific image need to be hand-picked by humans. Second, since an image has hundreds of thousands of pixels, directly annotating pixelwise smoothing results by humans is too labor-intensive and error-prone.

To establish the baseline algorithms in our benchmark, we resort to the latest deep neural networks. Deep neural networks have a large number of parameters (weights). Once these weights have been trained, they can be fixed and the resulting network has very strong generalization capability and can deal with different types of inputs. Thus, a trained deep neural network on edge-preserving smoothing dataset is expected to perform consistently

well in spite of the diverse image contents, which is the goal we want to achieve for edge-preserving image smoothing. We also note that deep learning has been broadly applied to low-level computer vision problems and has achieved state-of-the-art results. Examples include reproducing edge-preserving filters [13]–[15], image denoising [16], [17], image super-resolution [18]–[22], and JPEG deblocking [17], [23]. Specifically, we use the following two existing representative network architectures as our baseline methods, very deep convolution networks (VDCNN) and deep residual networks (ResNet). On top of these network architectures, we design novel loss functions well suited for edge-preserving image smoothing. The deep networks trained over our dataset run faster than most state-of-the-art edge-preserving smoothing algorithms, while the smoothing performance of our ResNet-based model outperforms these algorithms both qualitatively and quantitatively. Our benchmark will be publicly released.

Motivation for the work

In the last few decades, a lot of research has been conducted in the field of image denoising. However, there still remain some problems which have not been answered satisfactorily. First and foremost problem is of preserving important image features such as edges, corners and other sharp structures during the denoising process. Researchers all over the globe are working in the direction of achieving edge-preserving image denoising. Numerous approaches for denoising have been proposed in spatial and transform domain. The methods in both the domain have some problems that need to be overcome. In this work, we have identified such problems and tried to provide an effective solution to these problems. Apart from this, the study of denoising methods reveals a fact that transform domain methods such as wavelet-based approaches are found more dominant. Reason being wavelet transforms show localization in both time and frequency. Such localized nature of the wavelet transforms results in denoising with effective edge preservation [Luo (2006), Silva et al. (2012)]. Thus, all the proposed methods in this work used wavelet as a base to perform edge-preserving image denoising.

Literature Review

Reducing noise has always been one of the standard problems of the image analysis and processing community. Often though, at the same time as reducing the noise in a signal, it is important to preserve the edges. Edges are of critical importance to the visual appearance of images. Ideally denoising is all about filtering noise from the degraded image while keeping other details unchanged. Indeed, filtering is the most fundamental operation of image processing and computer vision, and it is used widely in various applications, including image smoothing and sharpening, noise removal, edge detection etc. In the broadest sense of the term “filtering”, the value of the filtered image at a given location is a function of the values of the input image in a small neighborhood of the same location [Gonzalez and Woods (2008), Jain and Tyagi (2013)].

The simplest form of filtering is an explicit Linear Translation Invariant (LTI) filtering, which can be implemented using a local neighborhood. For example, box filtering, also known as mean filtering or averaging [Gonzalez and Woods (2008)], is implemented by a local averaging operation where the value of each pixel is replaced by the average of all its neighbors. Although box filter gives the quickest filtering output, but its smoothing effect is often not sufficient. Other LTI filters that do not involve the computation of the mean of a neighborhood are also often used for smoothing. Most common of these are the Gaussian

smoothing filter [Shapiro and Stockman (2001), Gonzalez and Woods (2008)] and Wiener filter [Jain (1989), Benesty et al. (2010)]. The weights for Gaussian filter are chosen according to the shape of a Gaussian function. Gaussian filter has been proved to be a good choice for removing noise drawn from a normal distribution and the multi-scale space representation of an image can be obtained easily by Gaussian smoothing with increasing variance. Wiener filters are a class of optimum linear filters which involve linear estimation of a desired signal sequence from another related sequence.

Although LTI filtering is the simplest form of filtering and is used widely in early vision processing, it also has some drawbacks. LTI filtering is the quickest approach for smoothing the noise but some important structures are also often get blurred along with noise. To reduce these undesirable effects of linear filtering, a variety of edge-preserving filtering techniques have been proposed during the last few years.

Evolution of Edge-Preserving Image Denoising Research

Though, traditional LTI filtering techniques like mean filtering [Gonzalez and Woods (2008)], Gaussian filtering [Shapiro and Stockman (2001), Gonzalez and Woods (2008)], Wiener filtering [Jain (1989), Benesty et al. (2010)] exist for a long time for their simplicity and are able to achieve significant noise removal when the variance of noise is low, they tend to blur sharp edges, destroy lines and other fine image details. To resolve the above issues, a variety of nonlinear filters such as median [Gonzalez and Woods (2008), Pitas and Venetsanopoulos (1990)], weighted median [Yang et al. (1995)], rank conditioned rank selection [Hardie and Barner (1994)], and relaxed median [Hamza et al. (1999)] have been developed.

Apart from above nonlinear median type filters, other edge preserving denoising methods have been introduced to resolve the issues arised with linear spatial filtering during past few years.

These methods are non-linear and can preserve the image details and local geometries while removing the undesired noise, because they considers local structures and statistics during the filtering process.

Most of popular denoising techniques in this class have been developed based on Partial Differential Equations (PDEs) and variation models. The nonlinear Anisotropic Diffusion (AD) [Perona and Malik (1990), Black et al. (1998), Weickert et al. (1998)] methods were suggested to overcome blurring issues of the Gaussian filter [Shapiro and Stockman (2001), Gonzalez and Woods (2008)] by smoothing the image only in the direction orthogonal to the gradient. The regularization methods based on Total Variation (TV) [Rudin et al. (1992), Chambolle (2004)] were given to smooth the homogenous regions of the image but not its edges. Similarly, another approach based on low level processing to provide better edge preserving denoising known as the Smallest Univalued Segment Assimilating Nucleus (SUSAN) [Smith and Brady (1997)] filter has been proposed that can average all pixels in the local neighborhood which are from the same spatial region as the central pixel.

Based on the work [Aurich and Weule (1995), Smith and Brady (1997)], Tomasi and Manduchi (1998) proposed a simple, non-iterative, local filtering method known as the bilateral filter which was further modified and improved in [Elad (2002)]. Although bilateral filter was non-iterative and had simple formulation, but its direct implementation was known

to be slow. The brute force implementation has the time complexity $O(Nr^2)$, which is prohibitively high when the kernel radius r is large. To speed up the evaluation of the bilateral filter, several techniques [Durand and Dorsey (2002), Paris and Durand (2006), Porikli (2008), Yang et al. (2009), Chaudhury et al. (2011), Chaudhury (2013)] have been proposed, fast implementation of these techniques is still a challenging problem. Another issue concerning the bilateral filter [Durand and Dorsey (2002), Bae et al. (2006), Farbman et al. (2008)] is that it may have the gradient reversal artifacts in detail decomposition and High Dynamic Range (HDR) compression.

To resolve the issues raised with bilateral filter, He et al. (2010) proposed a new filter, called guided filter that can perform effective edge-preserving denoising by considering the content of a guidance image. One advantage of the guided filter over the bilateral filter was that it automatically had an $O(N)$ time exact algorithm. $O(N)$ time implies that, unlike the bilateral filter, the time complexity was independent of the window radius r , so they were free to use arbitrary kernel sizes in the applications. Unlike the bilateral filter, the guided filter avoided the gradient reversal artifacts that could appear in detail enhancement and HDR compression. The past few years have witnessed substantial developments in the area of image denoising. Buades et al. (2005a) presented an excellent survey on image denoising algorithms and also proposed an algorithm (Non-Local Means) for improvements in denoising results. The main focus of the work was, first, to define a general mathematical and experimental methodology to compare and classify the classical image denoising algorithms, second, to propose an algorithm (Non Local Means) addressing the preservation of structure in a digital image. It soon became clear that self-similarity and nonlocality are the characteristics of natural images with by far the biggest potential for image denoising. The Non-local Means (NLM) [Buades et al. (2005b)] filter is the first one which makes use of the self-similarity in the whole image. With the NLM filter, a denoised patch can be obtained by weighted averaging all other patches in the same image. It is an extension of the bilateral filter [Tomasi and Manduchi (1998)] in the sense of replacing the Euclidean distance between two pixels with the weighted Euclidean distance between two patches.

One of the most powerful and effective extensions of the non local filtering approach is the BM3D image denoising algorithm [Dabov et al. (2006)]. Maggioni et al. (2013) recently presented an extension of the BM3D algorithm, namely BM4D, to volumetric data denoising. Singular Value Decomposition (SVD) is also used in image noise filtering. Numerous approaches based on SVD filtering have been proposed in [Natarajan (1995), Konstantinides et al. (1997), Wongsawat et al. (2005), Orchard et al. (2008), Cai et al.(2010), Gu et al. (2014)]. Apart from edge-preserving filters mentioned so far, wavelets also gave superior performance in edge-preserving denoising due to properties such as sparsity and multiresolution structure.

With wavelet transform gaining popularity in the last few decades, numerous algorithms for image denoising in wavelet domain have been developed. Wavelet thresholding [Donoho and Johnstone (1994), Donoho and Johnstone (1995), Donoho (1995), Chang et al. (2000a), Chang et al. (2000b), Sendur and Selesnick (2002b), Silva et al. (2012), Jain and Tyagi (2014a)] is the key concept for wavelet domain denoising. The wavelet based methods exploit the decomposition of the data into a wavelet basis and modify the wavelet coefficients to denoise the data. The coefficients obtained through wavelet transform are modified

according to thresholding rule applied. This process of obtaining a denoised image from given noisy image using wavelet thresholding is termed as wavelet shrinkage.

Fodor and Kamath (2001) provided an empirical study on denoising using wavelet shrinkage. Inspired by the SURE-LET method [Blu and Luisier (2007)] and the guided filter [He et al. (2010)], Qiu et al. (2013) presented a novel edge-preserving smoothing filter, called LLSURE filter which is based on a local linear model and the principle of Stein's Unbiased Risk Estimate (SURE). The LLSURE filter has the edge-preserving smoothing property that can filter out noise

Existing System

Edge-preserving Image Smoothing using Block SVD

The wavelets have a strong influence on edge preserving image denoising problems. With wavelet transform, one can decompose the image signal into multiple subbands (a lowfrequency or smooth subband at coarsest scale and three detail subbands at all resolution levels) of wavelet coefficients. For most signals, energy mainly distributes in the smooth subband and energy in the detail subbands is clustered on a few large wavelet coefficients, corresponding to the edge structure of the original signal. In contrast, noise energy spreads over both the smooth subband and the detail subbands. Thus, noise can be suppressed through the thresholding of the small coefficients by using an appropriate threshold. At last, the thresholded coefficients are transformed back to the original domain to reconstruct the image. As described earlier that in wavelet domain, the noise energy dominates at lower scales, where the signal energy is highly localized and mainly distributes in a few large coefficients corresponding to the edge structures of images. Hence, wavelet transform provides a space which is quite suitable for the proposed adaptive thresholding to perform image denoising. Also, it was mentioned earlier that the proposed approach follows the divide-and-conquer strategy, i.e. it firstly divides the detail subband of wavelet domain into the blocks and then performs an edgepreserving block adaptive thresholding based on SVD for noise reduction. Although the idea of divide-and-conquer approach is not new, there exist two essential differences with respect to the conventional ones. First, conventional divide-and-conquer approach is devoted to reducing the cost of a filtering method. Whereas the divide-and-conquer approach used here is based on three aspects

- (a) natural images usually possess inhomogeneous nature;
- (b) most inhomogeneous images admit a partition into several homogeneous parts and
- (c) the SVD estimate has suitability with images showing homogeneous variations [Johnston and Silverman (1990)].

Secondly, a unique threshold is associated to all blocks in conventional divide-and-conquer approach. Since coefficients in one block might be more relevant to coefficients in other block in terms of edge details, so the global threshold is obviously not the best choice when we are considering the piecewise homogeneous characteristic of most blocks. Opposite to this, in the proposed approach, the threshold is associated with each block based on the decision about the presence of edge structure within the block. Similar to the WASVD (Adaptive SVD filtering in Wavelet domain) method [Hou (2003)], the proposed approach: divides the detail subbands of wavelet transform domain into blocks; applies SVD-based procedure to each block for checking the presence of edge structure in that block; associates different thresholds with edge present and edge absent blocks. However, unlike the WASVD, the proposed approach: instead of using fixed thresholds $\lambda_{nonedge}$ for all edge absent blocks and λ_{edge} for all edge present blocks, determines these thresholds locally for each block by considering the coefficients within the block and noise variance; instead of using the fixed

value of the noise variance which is used in the computation of threshold, the noise variance is estimated separately at each resolution level by taking the noise strength of that level into consideration; instead of using SVD filtering, performs the noise suppression through wavelet thresholding.

This scheme includes the computation of noise level and estimation of thresholding parameters independently for each block. The main stages of the proposed denoising method are illustrated in Fig. 1.

Proposed System

Adaptive Patch-based Edge-preserving Image Smoothing

Image denoising is one of the most diversified research areas in the field of image processing and computer vision. It is highly desirable for a denoising technique to preserve important image features e.g. edges, after denoising. Wavelet transforms show excellent proficiency in providing efficient edge-preserving image denoising, due to their capability of separating noise signals from image signals.

In this, we have presented a new technique for noise reduction using wavelet transforms. A Locally Adaptive Patch-Based (LAPB) thresholding which involves estimation of thresholding parameters in a local neighborhood and relies on the aggregation of multiple thresholded estimates of a wavelet coefficient, is employed to effectively suppress Gaussian noise while preserving relevant features of the original image. The proposed denoising approach is motivated by the spatial domain edge-preserving filtering method LLSURE [Qiu et al. (2013)]. Similar to their approach, our method also relies on aggregating the multiple filtered results of a pixel obtained due to its participation in overlapping regions of local neighborhoods. However, we have used this mechanism in wavelet domain which shows improvement over the LLSURE method. Since the proposed method itself is a wavelet based denoising method, so its main motive is to point out the existing limitations of the wavelet-based methods considered here and improving them. The following differences make the proposed method superior to the other wavelet-based denoising methods considered for comparison:

In other methods, the noise variance used for the threshold computation is estimated from the coefficients of HH1 subband (i.e. highest frequency subband) by using a robust median estimator [Donoho and Johnstone (1994)], and is kept fixed during the thresholding process through all resolution scales. However, the noise strength decreases with the increment in the resolution scale. Therefore, instead of using fixed noise variance, it should be estimated separately at each resolution scale from the coefficients of HH subband of that scale. The proposed method follows this way of estimating the noise variance. Unlike the other considered wavelet-based methods, the proposed method utilizes the sliding neighborhood mechanism to perform thresholding of wavelet coefficients. Each coefficient within the range of the sliding neighborhood, which is centered on a pixel location, is thresholded using the threshold computed locally using the member coefficients of the neighborhood.

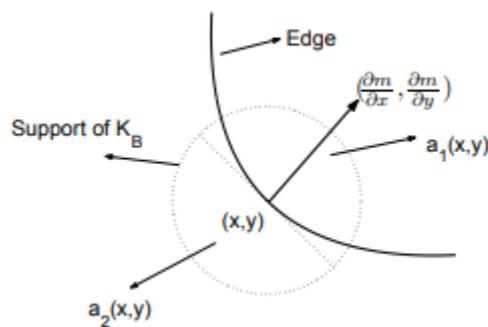
In locally adaptive thresholding schemes, each pixel (coefficient) definitely is a member of all the neighborhoods around every member coefficient in its neighborhood and accordingly participates in the computation of thresholds corresponding to all the member coefficients in its neighborhood. This fact leads us to consider the cumulative effect of all these thresholds to obtain the desired thresholded value of the coefficient. The proposed denoising method in the next section is based on this hypothesis. A new locally adaptive patch-based (LAPB)

thresholding scheme is proposed to threshold the small wavelet coefficients which are considered to be noise while preserving edges. This sort of thresholding involves computation of multiple thresholded estimates of a wavelet coefficient in a subband and the aggregation of such multiple estimates to obtain the desired thresholded value of that particular coefficient. In addition to this, it includes the calculation of noise level and estimation of thresholding parameters in a local neighborhood.

Edge-preserving surface estimation

An edge can be defined as a curve in the (X, Y) plane, along which the surface is discontinuous. Of course, the conventional estimator $\text{bac}(x, y)$ is biased for estimating $m(x, y)$, if there is an edge in the neighbourhood of (x, y) . Next, we present a solution to overcome this limitation.

By its definition, the gradient $(\partial m/\partial x, \partial m/\partial y)$ indicates the direction of the maximal increase in m around (x, y) . If the point (x, y) is on an edge segment, then the gradient direction would be asymptotically perpendicular to the tangent direction of the edge segment. The support of the kernel function K is then divided into two parts by a line passing the point (x, y) and perpendicular to the gradient direction $(\partial m/\partial x, \partial m/\partial y)$.



In the two parts, we define two one-sided local linear kernel estimators as follows:

$$(\hat{a}_j(x, y), \hat{a}_{j,x}(x, y), \hat{a}_{j,y}(x, y))$$

$$= \arg \min_{a,b,c} \sum_{i=1}^n (Z_i - a - b(X_i - x) - c(Y_i - y))^2 \cdot K_B^{(j)}((X_i - x), (Y_i - y)),$$

for $j = 1, 2$. In (3), $K^{(1)}B$ and $K^{(2)}B$ are the same as K_B in (2), except that their supports have been restricted to the two half-circles, as demonstrated. Then, $a_1(x, y)$ and $a_2(x, y)$ provide two one-sided estimators of $m(x, y)$.

By now, we have obtained three estimators for $m(x, y)$: the conventional estimator $a_c(x, y)$ and two one-sided estimators $a_1(x, y)$ and $a_2(x, y)$. If there are no edge pixels in the neighbourhood of (x, y) , then $a_c(x, y)$ should be selected for estimating $m(x, y)$, because it averages more observations around the point (x, y) and thus is more powerful in removing noise. If there is an edge segment around (x, y) , then the conventional estimator $a_c(x, y)$ is not a good estimator of $m(x, y)$ any more. In such a case, however, one of the two one-sided estimators $a_1(x, y)$ and $a_2(x, y)$ should still estimate the surface well, because most observations used by this estimator are located on a single side of the edge segment, guaranteed by the statistical properties of the estimated gradient direction from (2).

In practice, the edge locations are usually unknown; so, we need to choose among the three estimators a_c, a_1 and a_2 in a data-driven way, which is discussed below. The quality of the

three estimators a_c , a_1 and a_2 can be measured by the Weighted Residual Mean Squares (WRMS) of the related fitted surfaces, defined by:

$$\begin{aligned} & \text{WRMS}_c(x, y) \\ &= \frac{1}{\sum_i K_B(i)} \sum_i [Z_i - \hat{a}_c(x, y) - \hat{a}_{c,x}(x, y)(X_i - x) - \hat{a}_{c,y}(x, y)(Y_i - x)]^2 K_B(i), \end{aligned}$$

$$\begin{aligned} & \text{WRMS}_j(x, y) \\ &= \frac{1}{\sum_i K_B^{(j)}(i)} \sum_i [Z_i - \hat{a}_j(x, y) - \hat{a}_{j,x}(x, y)(X_i - x) - \hat{a}_{j,y}(x, y)(Y_i - x)]^2 K_B^{(j)}(i), \end{aligned}$$

The behaviour of these quantities depends on whether there are edge pixels in the neighbourhood of the point (x, y) . If there are no edge pixels in the neighbourhood, then all WRMS's are good estimators of the noise variance σ^2 . Otherwise, those WRMS's who use data points on both sides of edge segments would be biased for estimating σ^2 , and the bias would depend on the jump size and the Euclidean distance between the point (x, y) and the edge segments.

Based on these results, our edge-preserving surface estimator is defined by:

$$\hat{m}(x, y) = \begin{cases} \hat{a}_c(x, y) & \text{if } \text{diff}(x, y) \leq u \\ \hat{a}_1(x, y) & \text{if } \text{diff}(x, y) > u \text{ and } \text{WRMS}_1(x, y) < \text{WRMS}_2(x, y) \\ \hat{a}_2(x, y) & \text{if } \text{diff}(x, y) > u \text{ and } \text{WRMS}_1(x, y) > \text{WRMS}_2(x, y) \\ \frac{\hat{a}_1(x, y) + \hat{a}_2(x, y)}{2} & \text{if } \text{diff}(x, y) > u \text{ and } \text{WRMS}_1(x, y) = \text{WRMS}_2(x, y), \end{cases}$$

where u is a threshold value a

$$\text{diff}_{(x, y)} = \max \{ \text{WRMS}_c(x, y) - \text{WRMS}_1(x, y), \text{WRMS}_c(x, y) - \text{WRMS}_2(x, y) \}$$

So our surface estimator $\hat{m}(x, y)$ is defined by one of the three estimators: $a_c(x, y)$, $a_1(x, y)$ and $a_2(x, y)$, depending on whether there are edge pixels around (x, y) , judged by the WRMS values. If we are in a continuity region of the surface, then all three WRMS's are close to σ^2 , so that $\text{diff}_{(x, y)}$ is close to zero. On the other hand, if we are close to an edge segment, then one of the two one-sided WRMS's would be smaller than WRMS_c , and thus $\text{diff}_{(x, y)}$ would be relatively large. Therefore, $\text{diff}_{(x, y)}$ can be used to judge whether there are edge pixels around (x, y) . In (5), the case $\text{WRMS}_1(x, y) = \text{WRMS}_2(x, y)$ has, for n tending to infinity, probability zero to occur under some regularity conditions. It is included just for completeness. The explicit formulation of the estimator (5) is helpful when investigating theoretical properties of the estimator.

Edge Preservation Properties

Edges are fundamental features of images. They often contain valuable information and are important for human visual perception. In addition, edge information is used for image analysis and object classification. Median filters have good edge preservation properties. In fact, the median filter adapts to the signal characteristics in the sense that it behaves like a lowpass filter in the homogeneous regions for suppressing noise components, while it exhibits highpass behaviour close to the edges for preserving them. Images may contain horizontal, vertical and/or diagonal edges. The effectiveness of the median filter in preserving these edges depends on the geometry of the filter window. Cross-shaped windows are good at preserving horizontal and vertical edges, while the X-shaped filters are preferred for

preserving the diagonal edges. Square windows do not have preference for the edge direction and are found to preserve the edges of all orientations fairly well. Square-shaped filters are commonly used, since most of the images contain edges along all directions.

Noise Removing Filters

In the early development of signal and image processing, linear filters were the primary tools. Their mathematical simplicity and the existence of some desirable properties made them easy to design and implement. Moreover linear filters offered satisfactory performance in many applications. However linear filters have poor performance in the presence of noise that is not additive as well as in problems where system nonlinearities or nonGaussian statistics are encountered. In addition, various criteria such as the maximum entropy criterion lead to nonlinear solutions. In image processing applications, linear filters tend to blur the edges and do not remove impulsive noise effectively. They do not perform well in the presence of signal dependent noise. It is obvious that when the exact characteristics of our visual system are not well understood. Experimental results indicate that the first processing levels of our visual system possess nonlinear characteristics. For such reasons, nonlinear filtering techniques for signal/image processing were considered as early as 1958. There is a tremendous and dynamic development in the field of nonlinear filtering since then. Research in the field showcases its popularity. There is a widespread use of nonlinear digital filters in a variety of applications, notably in telecommunications, image processing and geophysical signal processing. Most of the currently available image processing software packages includes nonlinear filters (e.g. median filters and morphological filters).

Edge-preserving algorithms

Edge-preserving smoothing filters are much more suitable for feature extraction. Some examples of this filter class are:

- Median Filter
- Symmetrical Nearest Neighbour Filter (SNN)
- Maximum Homogeneity Neighbour Filter (MHN)
- Conditional Averaging Filter

These non linear algorithms are calculating the filtered gray value in dependence of the content of a defined neighbourhood. From the list of the neighbourhood pixels, only these are taken for the averaging, which have similar gray values compared to the pixel in consideration. Each edge-preserving filter has its own specific algorithm, but they all have in common, that the effect of this smoothing strategy is to preserve edges. Unfortunately, these smoothing filters have the characteristic not to smooth satisfyingly, because small gray value fluctuations existing in the really homogeneous areas are emphasized and not reduced. In addition, the Symmetrical Nearest Neighbour Filter is unable to produce reliable results in case of small areas.

Adaptive Median Filter

Some of the nonlinear filters are normally optimised for specific type of noise and specific type of images. Images are modelled as 2-D stochastic processes whose statistics vary from application to application. The noise statistics vary in various regions of the image. The types of noise vary in the application from one image to another. Since the image and noise statistics are unknown, non-adaptive filters cannot perform well. Adaptive filters can be expected to perform better. General Median filters often exhibit blurring for large window sizes, or insufficient noise suppression for small window sizes. Preservation of signal features and elimination of noise are two different issues in signal and image processing. To

overcome these limitations Adaptive Median Filter is designed and offers a better approach to achieve good noise filtering and fine details preservation. The Signal Adaptive Median filter, reported by, adjusts its window length automatically depending on the local signal to noise ratio and on the nature of the signal, that is, whether it is an edge or a flat region. The filter allows simultaneous removal of a combination of signal dependent and additive random noise in addition to mixed impulse noise in images.

The median filter and its variants such as weighted median filter, center weighted median filter performs well, as long as the spatial density of the impulsive noise is not large. A decision based Adaptive Median Filter (AMF) (Hwang and Haddad 1995) has been proposed to remove impulse noise with variable window size. Adaptive median filters can handle impulse noise with higher probabilities and preserve image sharpness. The adaptive median filter works in a rectangular window area that increases in size during the filtering operation, depending on density of the noise. The problem is that the standard median filter replaces every point in an image by the median of the corresponding neighbourhood. This causes unnecessary loss of image details. If the median value of the pixels in the window too is an impulse, adaptive median filter increases the window size and replaces the pixel under test by the median value of the new window. The AMF is superior to Lin's adaptive scheme because it is simpler and better performing in removing the high density of impulsive noise as well as nonimpulsive noise and in preserving fine details. In the case of high density impulse noise, the adaptive algorithm performed quite well. The choice of maximum allowed window size depends on the application, but a reasonable starting value can be estimated by experimenting with various sizes of the standard median filter first.

A novel robust estimation, based filter is proposed to remove low to high density salt and pepper noise effectively. The robust formulation aims at eliminating the noise outliers while preserving the edge structures in the restored image. Many of the existing filters, such as, Adaptive Median Filter (Hwang and Haddad 1995), Progressive Switching Median Filter (Wang and Zhang 1999), Boundary Discriminative Noise Detector (Pei-Eng Ng and Ma 2006), Srimi-Ebenezer method (Srinivasan and Ebenezer 2007) are removing high density salt and pepper noise effectively. However, they fail to restore the edges and fine details when the noise density increases above 70%. Hence, the aim of the chapter is, to propose a new nonlinear algorithm to remove high density impulse noise with edge and detail preservation, up to a noise level of 90%. The function of the proposed filter is to detect the outlier pixels and restore the original value, using robust estimation. The restoration results are compared with the standard median filter, weighted median filter, progressive switching median filter, adaptive median filter and a recently proposed Srimi-Ebenezer method and BDND method and MTND method. Experimental results show that the proposed filter removes low to high density salt and pepper noise and preserves edges and fine details very satisfactorily upto a noise density as high as 90%

Let $\{x(i)\}$ and $\{y(i)\}$ denote the input and output sequences respectively. A one-dimensional median filter slides a $2N+1$ point wide window.

over $\{x(i)\}$. At each point the samples inside the window are sorted out and the median or middle value is used as the filter output. The median output is associated with the time sample at the center of the window. The filtering procedure can be expressed as:

$$y(i) = \text{median}(x(i-N), x(i-N+1), \dots, x(i), \dots, x(i+N)), i \in Z$$

The Equation is also called moving median or running median. For the window to reach the front and rear ends of the input signal sequence, N number of samples are appended both at the beginning and at the end. The front endpoints take the value of the first sample while the rear endpoints take the value of the last sample. An example of one-dimensional median

filtering with window length is illustrated in Figure 2. The two-dimensional median filtering process has the following definition:

$$y(i, j) = \text{median}(x(i+r, j+s)), (r, s) \in A \text{ and } (i, j) \in Z^2$$

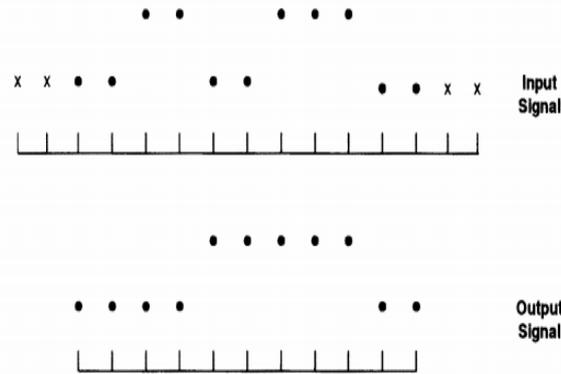


Figure 2 Illustration of one-dimensional median filtering with window size 5

The set $A \subset Z^2$ defines the neighbourhood of the central pixel (i, j) . It is called the filter window. The commonly used window structures in two-dimensional median filtering are shown in Figure 3. The border samples of twodimensional signals are processed by replicating them as done in onedimensional median filtering. The median filters have been used with success in speech (Jayant 1976 and Rabiner et al 1975) and image (Pratt 1991 and Pitas and Venetsanopoulos 1990 and Perlman et al 1987) processing applications. In addition, fast algorithms and hardware implementation for median filtering have been developed (Ataman et al 1980, Huang et al 1979, Ahmad and Sundararajan 1987 and Oflazer 1983).

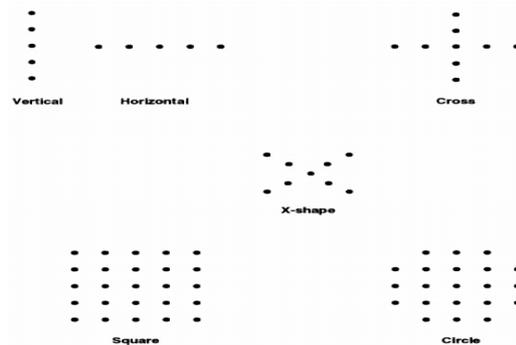


Figure 3 Window structures used for two-dimensional median filtering

Properties of Median Filters

Conventional tools such as frequency response and impulse response cannot be used for analyzing median filters as they do not come within the scope of linear system theory. As a result, new tools had to be developed to analyze and characterize the behaviour of these nonlinear filters deterministically and statistically (Gallagher, Jr. and Wise 1981, Nodes and Gallagher, Jr. 1984, Ataman et al 1981, Kuhlmann and Wise 1981, Bovik 1987a, Fitch et al 1985, Wendt et al 1986, Arce and Gallagher 1982, Astola et al 1987, Zeng 1994, Eberly et al 1991 and Mao and Gan 1993). The deterministic properties describe the effect of filtering on the structure of the signal. On the other hand, the statistical analysis shows how effective the filter is in removing the different types of noise.

Deterministic Properties of Median Filters

The deterministic properties of median filters are described by their root signal set. The root signal set is defined as a set of signals, which remain invariant to further filtering. The

concept of root signal with reference to the median filter is explained as follows. Under steady state conditions, when a sinusoidal signal is passed through a linear system, the frequency of the sinusoid is not changed; only its phase and amplitude are altered. This fact is not valid for median filters, because they are basically nonlinear systems. However, proved that if any signal of finite-length is repeatedly median filtered using the same window, then the resultant signal becomes invariant to further filtering at one point. Such a signal is called the root signal. That is, for a median filter of length $k = 2N+1$, this means that:

$$x(i) = \text{median}(x(i-N), x(i-N+1), \dots, x(i), \dots, x(i+N))$$

If the above condition is satisfied for all i , then $\{x(i)\}$ is called the root signal of that particular median filter. The meaning of root signals for median filtering is analogous to the meaning of sinusoids in the passband of linear filters. If the original signal is of length L points (without counting the appended points at the beginning and the end), then

$$3 \left(\frac{L-2}{2(N+2)} \right)$$

filter passes are the maximum required to reach a root. The fact that the root signals are invariant to further filtering offers interesting possibilities. For example, in image filtering, a common approach is to design a median filter such that certain prescribed features, such as lines are root signals and thus not disturbed by the filtering operation. Root signals of median filters have been used for speech and image coding

Statistical Properties of Median Filters

Median is the best location estimator in the L_1 sense, because it minimizes:

$$\sum_{i=1}^n |x(i) - T_n| = \min$$

The median filter is robust in the presence of long tailed noise. The effect of an outlier (impulse) on the performance of an estimator can be studied by a function called Influence Function (IF). The IF of the estimator T , at the distribution F for those $x \in X$, is denoted as $IF(x; T, F)$, where X is the sample space. The IF of the mean and median estimators at the Gaussian distribution of zero mean and unit variance is shown in Figure 4. It can be seen from Figure 4 that the influence of an outlier on the mean estimator keeps increasing with the magnitude of x , while it gets bounded at $1/(2\phi(0))$ on the median estimator, irrespective of the magnitude of x

The most important measure of robustness based on the IF is the gross error sensitivity v^* of T at the distribution F

$$v^* = \sup_x |IF(x; T, F)|$$

where \sup denotes supremum. The gross error sensitivity measures the worst effect of contamination at any point $x \in X$. If v is finite, T is called the robust estimator. From the Equation and Figure 4, it is evident that v^* is unbounded for the mean estimator for unbounded values of x . Therefore, the mean is not a robust estimator; even one distant outlier can cause catastrophic effects on the arithmetic mean. The gross error sensitivity of the median estimator for zero mean Gaussian distribution of unit variance is :

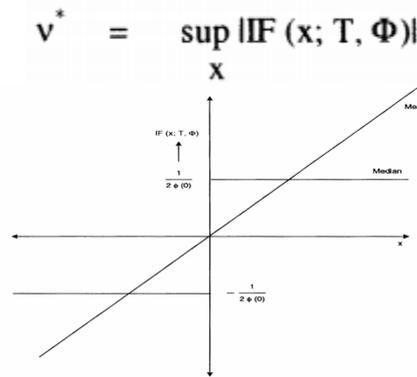


Figure 4 Influence function of the mean and median estimators for zero mean Gaussian distribution of unit variance

$$= \frac{1}{2\phi(0)}$$

$$= 1.2533$$

The finite value of v^* indicates that the median is a robust estimator.

To evaluate the median as an estimator of location, a measure called Asymptotic Relative Efficiency $ARE(T, S)$ of two estimators $T(F)$ and $S(F)$ is used and it is defined as:

$$ARE(T, S) = \frac{V(S, F)}{V(T, F)}$$

where $V(S, F)$ and $V(T, F)$, respectively, are the asymptotic variance of estimators S and T at the distribution F . To obtain an idea of the performance of the median, it is compared with the arithmetic mean \bar{x} for different distributions. The asymptotic relative efficiency of the median estimator with respect to the mean at the distribution F is:

$$ARE(\text{median}(x(i)), \bar{x}) = \frac{V(\bar{x}, F)}{V(\text{median}(x(i)), F)}$$

When $ARE(\text{median}(x(i)), \bar{x})$ is greater than one, the median performs better, that is, it exhibits lower output variance than the arithmetic mean. $ARE(\text{median}(x(i)), \bar{x})$ values evaluated at different distributions are summarized in Table 1

Noise Probability Density Function	$ARE(\text{median}(x(i)), \bar{x})$
Uniform	1/3
Gaussian	2/π
Laplacian	2

Table 1 Asymptotic Relative Efficiency of Median Estimator with respect to Mean Estimator

The median performs at its worst for the short tailed uniform distribution and performs at its best for the long tailed laplacian distribution.

Advantages

The proposed method has several desirable features due to which it achieves an effective edgepreserving denoising. First, the approach of divide-and-conquer used in our method adapts to the inhomogeneous nature of natural images. Second, block-dependent thresholding

which relies on the estimation of thresholding parameters locally for each block, enhances the denoising performance as it can characterize the local features better than a subband-dependent thresholding. Third, instead of having fixed noise variance, estimating it locally at each resolution scale makes it more beneficial as it takes the noise strength at that scale into consideration. Fourth, edge adaptive thresholding is applied to each block which considers the edge strength of that block for better edge-preservation.

Experimental Results

The results shown in tables demonstrate that the proposed method is superior to all the methods in the left half section regarding the PSNR and SSIM measures. However, when only wavelet-based approaches are considered, the proposed method is at times contested by the SURELET method in respect to all available performance measures. Although the proposed method is not always superior to the other methods in terms of all three performance measures.

A more detailed visual comparison between the deep models and the $L1$ smoothing algorithm [12] is given in Figure 4 considering the fact that the $L1$ smoothing algorithm is the most frequently chosen algorithm and it achieves the lowest WRMSE and WMAE among existing state-of-the-art algorithms. From Figure 4 we can see that the $L1$ smoothing algorithm wrongly increases the color contrast between two flattened regions on the airplane. The results from VDCNN and ResNet models do not have such artifacts. As mentioned earlier, we do not aim to reproduce individual filters like [13]–[15], [27]. By utilizing the constructed dataset, our baseline algorithm aims to train a deep CNN model that can produce reasonable edge-preserving smoothing results for a wide range of image contents without further tuning parameters. To the best of our knowledge, existing smoothing algorithms cannot perform consistently well on a wide range of image contents using a single parameter setting. We can see that the $L0$ smoothing algorithm needs to set different parameters for the ‘Racing car’ and the ‘Gloves’ images. If we set $\lambda = 0.03$ for the ‘Racing Car’ image, the edge between grass and road will blur. $\lambda = 0.01$ is the proper setting for the ‘Racing Car’ image. However, if we set $\lambda = 0.01$ for the ‘Gloves’ image, there still remain undesirable noises. In contrast, our ResNet model produces robust visual results on different images without tuning parameters. More results can be found in the supplementary file.

Tone Mapping

Tone mapping is a popular technique to map one set of colors to another to reproduce the appearance of a high dynamic range (HDR) image on a low dynamic range (LDR) display. The state-of-the-art tone mappers commonly adopt a layer decomposition scheme to decompose the HDR image into low- and high-frequency layers and then process them separately. In particular, the low frequency layer is estimated by applying an edge-preserving filter to the original HDR image. The edge-preserving property is very important for avoiding halo artifact and achieving naturalness in the tonemapped images. Thus, a stable and effective edge-preserving filter is highly desirable to improve the tone mapping performance.

To avoid halo artifact, an edge-preserving filter should be able to preserve the strong edge regions and flatten other regions in the image, regardless of the image contents and types. Our ResNet baseline model can handle this task well, because it is trained on our dataset which is constructed with such criteria. We use the tone mapping framework in [33] by replacing the original bilateral filter by our ResNet model. We compare the tone mapped results with several state-of-the-art tone mappers, including bilateral filter method (BF) [33],

visual adaptation (VAD) [34], and local edge-preserving filter (LEP) [35]. BF-based tone mapper [33] may not be as effective as the recently proposed approaches, but BF is widely adopted in different image processing tasks. On the other hand, VAD [34] and LEP [35] are selected because they obtain state-of-the-art performance. We do not compare with [41] because saliency is beyond the scope of this work. Fig. 5 shows our tone mapping results compared with these tone mappers. We can see that our tone mapper with ResNet model reaches an excellent balance between halo removal and naturalness preservation. Other tone mappers suffers from either halo artifact or over-enhancement problems.

Peak Signal to Noise Ratio

The peak signal to noise ratio often abbreviated as PSNR, is the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation. Because many signals have a very wide dynamic range, PSNR is usually expressed in terms of the logarithmic decibel scale. The PSNR is most commonly used as a measure of quality of reconstruction in image denoising and image restoration. It is easily defined via the Mean Square Error (MSE). For 2D $M \times N$ monochrome images, the formula for PSNR calculation is given by equation

$$PSNR = 10 \log_{10} \left(\frac{255^2}{MSE} \right) = 10 \log_{10} \left(\frac{MAX^2}{MSE} \right)$$

Where MAX is the maximum pixel value of the image. When the pixels are represented using 8 bits per sample, this is 255. Higher the PSNR better is the quality

Mean Square Error (MSE)

Mean Square Error indicates average error of the pixels throughout the image. A definition of a MSE does not indicate that the denoised image suffers more errors instead it refers to a greater difference between the original and denoised image. This means that there is a significant noise reduction. The formula for the MSE calculation is given by Equation

$$MSE = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N (r_{ij} - x_{ij})^2$$

where r_{ij} is the original image, x_{ij} is the restored image

Structural Similarity Index (SSIM)

The structural similarity index is a method for measuring the similarity between two images. The index can be viewed as a quality measure of one of the images being compared provided the other image is regarded as of perfect quality. The SSIM metric is calculated on various windows of an image. The measure between two windows x and y of common size $N \times N$ is:

$$MSSIM(X, Y) = \frac{(2\mu_x \mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

where μ_x the average of X ; μ_y the average of Y ; σ_x the variance of X ; σ_y the variance of Y ; σ_{xy} the covariance of X and Y ; $c_1 = (k_1 L)^2$, $c_2 = (k_2 L)^2$ two variables to stabilize the division with weak denominator; L the dynamic range of the pixel-values.

CONCLUSIONS

We presented a benchmark for edge-preserving image smoothing for the purpose of quantitative performance evaluation and further advancing the state-of-the-art. This

benchmark consists of 500 source images and their “groundtruth” image smoothing results as well as baseline learning models. The baseline models are representative deep convolutional network architectures, on top of which we design novel loss functions well suited for edge-preserving image smoothing. Our trained deep networks run fast at test time while their smoothing results outperform state-of-the-art smoothing algorithms both quantitatively and qualitatively.

Future Work

In the future, work can be done to provide more effective solutions for the identified problems. In addition the presented image denoising techniques can be tested on other grayscale image datasets for further establishing their efficacy. Also the work can be extended to denoising of color images as well as video sequences. The proposed techniques in tetralet domain are well suited for denoise square natural grayscale images with dimensions in the exponential order of two. However, if image is not a square then it has to be extended to make it a suitable input image. After denoising, the image is cropped to get the original size. But such adjustment may severely affect the denoising performance. Also choosing an appropriate number of tetromino coverings being averaged is a crucial point. In future, the work can be done to resolve these issues.

All the denoising techniques presented in this thesis have developed using Discrete Wavelet Transforms (DWT) which, due to its filter-bank implementation, offers a high flexibility in implementation and the possibility of using a wide number of wavelet families. Besides all the benefits of this transform, it has also a series of limitations such as its shift sensitivity and its poor directional selectivity. These limitations can be somewhat overcome by using some of DWT’s extensions, such as the Undecimated Discrete Wavelet Transform (UDWT) which is translation invariant or the Discrete Wavelet Packet Transform, that offers a better directional selectivity. Another way of overcoming these limitations is given by the use of Complex Wavelet Transforms (CWT). These variants of wavelet transforms can be used for future enhancement of denoising techniques.

References

- [1] C. Tomasi and R. Manduchi, “Bilateral filtering for gray and color images,” in *Proc. 6th Int. Conf. Comput. Vis.*, Jan. 1998, pp. 839–846.
- [2] P. Perona and J. Malik, “Scale-space and edge detection using anisotropic diffusion,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 7, pp. 629–639, Jul. 1990.
- [3] Z. Farbman, D. Lischinski, and R. Szeliski, “Edge-preserving decompositions for multi-scale tone and detail manipulation,” *Trans. Graph.*, vol. 27, no. 3, p. 67, Aug. 2008.
- [4] R. Fattal, “Edge-avoiding wavelets and their applications,” *ACM Trans. Graph.*, vol. 28, no. 3, Aug. 2009, Art. no. 22.
- [5] Q. Zhang, X. Shen, L. Xu, and J. Jia, “Rolling guidance filter,” in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2014, pp. 815–830.
- [6] B. Ham, M. Cho, and J. Ponce, “Robust guided image filtering using nonconvex potentials,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 1, pp. 192–207, Jan. 2017.
- [7] L. Xu, C. Lu, Y. Xu, and J. Jia, “Image smoothing via L_0 gradient minimization,” *ACM Trans. Graph.*, vol. 30, no. 6, Dec. 2011, Art. no. 174.
- [8] D. Min, S. Choi, J. Lu, B. Ham, K. Sohn, and M. Do, “Fast global image smoothing based on weighted least squares,” *IEEE Trans. Image Process.*, vol. 23, no. 12, pp. 5638–5653, Dec. 2014.

- [9] L. Bao, Y. Song, Q. Yang, H. Yuan, and G. Wang, "Tree filtering: Efficient structure-preserving smoothing with a minimum spanning tree," *IEEE Trans. Image Process.*, vol. 23, no. 2, pp. 555–569, Feb. 2014.
- [10] Q. Zhang, L. Xu, and J. Jia, "100+ times faster weighted median filter (WMF)," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2830–2837.
- [11] S. Paris, S. W. Hasinoff, and J. Kautz, "Local Laplacian filters: Edgeaware image processing with a Laplacian pyramid," *ACM Trans. Graph.*, vol. 30, no. 4, pp. 1–68, Jul. 2011.
- [12] S. Bi, X. Han, and Y. Yu, "An L_1 image transform for edgepreserving smoothing and scene-level intrinsic decomposition," *ACM Trans. Graph.*, vol. 34, no. 4, Aug. 2015, Art. no. 78.
- [13] L. Xu, J. Ren, Q. Yan, R. Liao, and J. Jia, "Deep edge-aware filters," in *Proc. 32nd Int. Conf. Mach. Learn. (ICML)*, 2015, pp. 1669–1678.
- [14] S. Liu, J. Pan, and M.-H. Yang, "Learning recursive filters for low-level vision via a hybrid neural network," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, Sep. 2016, pp. 560–576.
- [15] Y. Li, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Deep joint image filtering," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 154–169.
- [16] X. Mao, C. Shen, and Y.-B. Yang, "Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 2802–2810.
- [17] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3142–3155, Jul. 2017.
- [18] J. Kim, J. K. Lee, and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1646–1654.
- [19] C. Ledig, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017.
- [20] J. Kim, J. K. Lee, and K. M. Lee, "Deeply-recursive convolutional network for image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 1637–1645.
- [21] Y. Tai, J. Yang, and X. Liu, "Image super-resolution via deep recursive residual network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2790–2798.
- [22] Y. Tai, J. Yang, X. Liu, and C. Xu, "Memnet: A persistent memory network for image restoration," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4539–4547.
- [23] C. Dong, Y. Deng, C. C. Loy, and X. Tang, "Compression artifacts reduction by a deep convolutional network," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 576–584.
- [24] 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 *Proc. 8th IEEE Int. Conf. Comput. Vis. (ICCV)*, vol. 2, Jul. 2001, pp. 416–423.
- [25] B. Ham, M. Cho, and J. Ponce, "Robust image filtering using joint static and dynamic guidance," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 4823–4831.

- [26] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 898–916, May 2011.
- [27] Q. Fan, J. Yang, G. Hua, B. Chen, and D. Wipf, "A generic deep architecture for single image reflection removal and image smoothing," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 3238–3247.
- [28] K. Ma *et al.*, "Group MAD competition? A new methodology to compare objective image quality models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1664–1673.
- [29] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced deep residual networks for single image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR) Workshops*, Jul. 2017, pp. 136–144.
- [30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
- [31] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1026–1034.
- [32] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, Jan. 2015, pp. 1–15.
- [33] F. Durand and J. Dorsey, "Fast bilateral filtering for the display of highdynamic-range images," *ACM Trans. Graph.*, vol. 21, no. 3, pp. 257–266, 2002.
- [34] S. Ferradans, M. Bertalmio, E. Provenzi, and V. Caselles, "An analysis of visual adaptation and contrast perception for tone mapping," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 10, pp. 2002–2012, Oct. 2011.
- [35] B. Gu, W. Li, M. Zhu, and M. Wang, "Local edge-preserving multiscale decomposition for high dynamic range image tone mapping," *IEEE Trans. Image Process.*, vol. 22, no. 1, pp. 70–79, Jan. 2013.
- [36] X. Fu, D. Zeng, Y. Huang, X.-P. Zhang, and X. Ding, "A weighted variational model for simultaneous reflectance and illumination estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2782–2790.
- [37] Z. Liang, W. Liu, and R. Yao, "Contrast enhancement by nonlinear diffusion filtering," *IEEE Trans. Image Process.*, vol. 25, no. 2, pp. 673–686, Feb. 2016.
- [38] K. He, J. Sun, and X. Tang, "Guided image filtering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 6, pp. 1397–1409, Jun. 2013.
- [39] H. Yeganeh and Z. Wang, "Objective quality assessment of tone-mapped images," *IEEE Trans. Image Process.*, vol. 22, no. 2, pp. 657–667, Feb. 2013.
- [40] V. Jaya and R. Gopikakumari, "IEM: A new image enhancement metric for contrast and sharpness measurements," *Int. J. Comput. Appl.*, vol. 79, no. 9, pp. 1–9, Oct. 2013.
- [41] Z. Li and J. Zheng, "Visual-saliency-based tone mapping for high dynamic range images," *IEEE Trans. Ind. Electron.*, vol. 61, no. 12, pp. 7076–7082, Dec. 2014.
- [42] H. Yue, J. Yang, X. Sun, F. Wu, and C. Hou, "Contrast enhancement based on intrinsic image decomposition," *IEEE Trans. Image Process.*, vol. 26, no. 8, pp. 3981–3994, Aug. 2017.