

A Novel Approach to Image Denoising and Image Inpainting

R.Revathi., MCA.,M.Phil.,M.E.,
Assistant Professor, Department of Computer Science,
Jawaharlal Nehru College for Women, Ulundurpet.
Email: sriramrr2007@gmail.com

ABSTRACT

Image denoising is an important image processing task, both as a process itself, and as a component in other processes. Very many ways to denoise an image or a set of data exists. The main properties of a good image denoising model are that it will remove noise while preserving edges. Traditionally, linear models have been used. One common approach is to use a Gaussian filter, or equivalently solving the heat-equation with the noisy image as input-data, i.e. a linear, 2nd order PDE-model. For some purposes this kind of denoising is adequate. One big advantage of linear noise removal models is the speed. But a back draw of the linear models is that they are not able to preserve edges in a good manner: edges, which are recognized as discontinuities in the image, are smeared out. Here I am using a novel approach to image denoising that is level set approach is employed. Level Set Methods offer an appealing approach to noise removal. In particular, they exploit the fact that curves moving under their curvature smooth out and disappear. Since the method evolves contours, boundaries remain essentially sharp and do not blur. Second, a "min/max" switch is used to control whether or not curvature flow is applied; this results in an algorithm that stops automatically once the smallest features are removed.

Keywords - Gaussian denoising, single image super-resolution (SISR) and JPEG image deblocking,DnCNN, AWGN.

Date of Submission: Dec 28, 2017

Date of Acceptance: Jan 17, 2018

I. INTRODUCTION

Discriminative model learning for image denoising has been recently attracting considerable attentions due to its favorable denoising performance. In this paper, we take one step forward by investigating the construction of feed-forward denoising convolutional neural networks (DnCNNs) to embrace the progress in very deep architecture, learning algorithm, and regularization method into image denoising. Specifically, residual learning and batch normalization are utilized to speed up the training process as well as boost the denoising performance. Different from the existing discriminative denoising models which usually train a specific model for additive white Gaussian noise (AWGN) at a certain noise level, our DnCNN model is able to handle Gaussian denoising with unknown noise level (i.e., blind Gaussian denoising). With the residual learning strategy, DnCNN implicitly removes the latent clean image in the hidden layers. This property motivates us to train a single DnCNN model to tackle with several general image denoising tasks such as Gaussian denoising, single image super-resolution and JPEG image deblocking. Our extensive experiments demonstrate that our DnCNN model can not only exhibit high effectiveness in several general image denoising tasks, but also be efficiently implemented by benefiting from GPU computing. The idea is to view the pixel values as a topographic map; the intensity (somewhere between white and black) at each pixel is the height of the surface at that point. Suppose we then let each contour undergo motion by curvature. Then very small contours, corresponding to spikes of noise, will disappear quickly. Better yet, the boundaries will remain sharp, since they will not blur

under this motion, and instead only move according to their curvature.

II. EXISTING WORK

Observed image signals are often corrupted by acquisition channel or artificial editing. The goal of image restoration techniques is to restore the original image from a noisy observation of it. Image denoising and inpainting are common image restoration problems that are both useful by themselves and important preprocessing steps of many other applications. Image denoising problems arise when an image is corrupted by additive white Gaussian noise which is common result of many acquisition channels, whereas image inpainting problems occur when some pixel values are missing or when we want to remove more sophisticated patterns, like superimposed text or other objects, from the image. This paper focuses on image denoising and blind inpainting. Various methods have been proposed for image denoising. One approach is to transfer image signals to an alternative domain where they can be more easily separated from the noise [1, 2, 3]. For example, Bayes Least Squares with a Gaussian Scale-Mixture (BLS-GSM), which was proposed by Portilla et al, is based on the transformation to wavelet domain [2]. Another approach is to capture image statistics directly in the image domain. Following this strategy, A family of models exploiting the (linear) sparse coding technique have drawn increasing attention recently [4, 5, 6, 7, 8, 9]. Sparse coding methods reconstruct images from a sparse linear combination of an over-complete dictionary. In recent research, the dictionary is learned from data instead of hand crafted as before. This learning step improves the

performance of sparse coding significantly. One example of these methods is the KSVD sparse coding algorithm proposed in [6].

III. IMAGE INPAINTING

Image inpainting methods can be divided into two categories: non-blind inpainting and blind inpainting. In non-blind inpainting, the regions that need to be filled in are provided to the algorithm a priori, whereas in blind inpainting, no information about the locations of the corrupted pixels is given and the algorithm must automatically identify the pixels that require inpainting. The state-of-the-art non-blind inpainting algorithms can perform very well on removing text, doodle, or even very large objects [10, 11, 12]. Some image denoising methods, after modification, can also be applied to non-blind image inpainting with state-of-the-art results [7]. Blind inpainting, however, is a much harder problem. To the best of our knowledge, existing algorithms can only address i.i.d. or simply structured impulse noise [13, 14, 15]. Although sparse coding models perform well in practice, they share a shallow linear structure. Recent research suggests, however, that non-linear, deep models can achieve superior performance in various real world problems. One typical category of deep models are multi-layer neural networks. In [16], Jain et al. proposed to denoise images with convolutional neural networks. In this paper, we propose to combine the advantageous “sparse” and “deep” principles of sparse coding and deep networks to solve the image denoising and blind inpainting problems.

The sparse variants of deep neural network are expected to perform especially well in vision problems because they have a similar structure to human visual cortex [17]. Deep neural networks with many hidden layers were generally considered hard to train before a new training scheme was proposed which is to adopt greedy layer-wise pre-training to give better initialization of network parameters before traditional back-propagation training [18, 19]. There exist several methods for pre-training, including Restricted Boltzmann Machine (RBM) and Denoising Auto-encoder (DA) [20, 21]. We employ DA to perform pre-training in our method because it naturally lends itself to denoising and inpainting tasks.

3.1 Denoising Auto-Encoder

DA is a two-layer neural network that tries to reconstruct the original input from a noisy version of it. The structure of a DA is shown in Fig.1a. A series of DAs can be stacked to form a deep network called Stacked Denoising Auto-encoders (SDA) by using the hidden layer activation of the previous layer as input of the next layer. SDA is widely used for unsupervised pre-training and feature learning [21]. In these settings, only the clean data is provided while the noisy version of it is generated during training by adding random Gaussian or Salt-and-Pepper noise to the clean data. After training of one layer, only the clean data is passed on to the network to produce the clean training data for the next layer while the noisy data is discarded.

The noisy training data for the next layer is similarly constructed by randomly corrupting the generated clean training data. For the image denoising and inpainting tasks, however, the choices of clean and noisy input are natural: they are set to be the desired image after denoising or inpainting and the observed noisy image respectively. Therefore, we propose a new training scheme that trains the DA to reconstruct the clean image from the corresponding noisy observation. After training of the first layer, the hidden layer activations of both the noisy input and the clean input are calculated to serve as the training data of the second layer.

Our experiments on the image denoising and inpainting tasks demonstrate that SDA is able to learn features that adapt to specific noises from white Gaussian noise to superimposed text. Inspired by SDA’s ability to learn noise specific features in denoising tasks, we argue that in unsupervised feature learning problems the type of noise used can also affect the performance. Specifically, instead of corrupting the input with arbitrarily chosen noise, more sophisticated corruption process that agrees to the true noise distribution in the data can improve the quality of the learned features. For example, when learning audio features, the variations of noise on different frequencies are usually different and sometimes correlated. Hence instead of corrupting the training data with simple i.i.d. Gaussian noise, Gaussian noise with more realistic parameters that are either estimated from data or suggested by theory should be a better choice

IV. MODEL DESCRIPTION

In this section, we first introduce the problem formulation and some basic notations. Then we briefly give preliminaries about Denoising Auto-encoder (DA), which is a fundamental building block of our proposed method.

4.1 Problem Formulation

Assuming x is the observed noisy image and y is the original noise free image, we can formulate the image corruption process as

$x = \eta(y)$. (1) where $\eta : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is an arbitrary stochastic corrupting process that corrupts the input. Then, the denoising task’s learning objective becomes:

$$f = \operatorname{argmin}_f E_y \|kf(x) - y\|_k$$

From this formulation, we can see that the task here is to find a function f that best approximates η^{-1} . We can now treat the image denoising and inpainting problems in a unified framework by choosing appropriate η in different situations. 2.2 Denoising Auto-encoder Let y_i be the original data for $i = 1, 2, \dots, N$ and x_i be the corrupted version of corresponding y_i . DA is defined as shown in Fig.1a: $h(x_i) = \sigma(Wx_i + b)$ (3) $\hat{y}(x_i) = \sigma(W_0h(x_i) + b_0)$ (4) where $\sigma(x) = (1 + \exp(-x))^{-1}$ is the sigmoid activation function which is applied element-wise to vectors, h_i is the hidden layer activation, $\hat{y}(x_i)$ is an approximation of y_i and $\Theta = \{W, b, W_0, b_0\}$ represents the weights and biases. DA can be trained with various optimization methods to minimize the reconstruction loss: $\theta = \operatorname{argmin}_\theta \sum_{i=1}^N \|y_i - \hat{y}(x_i)\|_k$. (5) After finish training a DA, we can move on to training the next layer by using the hidden layer activation of the first layer as the input of the next

layer. This is called stacked denoising auto encoder (SDA) [21].

4.2 Denoising Gaussian Noise

Denoising White Gaussian Noise We first corrupt images with additive white Gaussian noise of various standard deviations. For the proposed method, one SSDA model is trained for each noise level. We evaluate different hyperparameter combinations and report the best result. We set K to 2 for all cases because adding more layers may slightly improve the performance but require much more training time. In the meantime, we try different patch sizes and find that higher noise level generally requires larger patch size

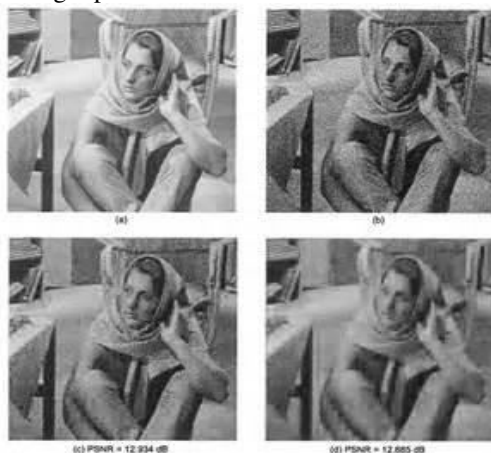
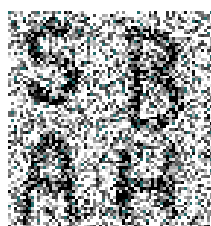


Fig.1. Noise Reduction

4.3 Image Inpainting

For the image inpainting task, we test our model on the text removal problem. Both the training and testing set compose of images with super-imposed text of various fonts and sizes from 18-pix to 36-pix. Due to the lack of comparable blind inpainting algorithms, We compare our method to the non-blind KSVD inpainting algorithm [7], which significantly simplifies the problem by requiring the knowledge of which pixels are corrupted and require inpainting. A visual comparison is shown in Fig.3. We find that SSDA is able to eliminate text of small fonts completely while text of larger fonts is dimmed. The proposed method, being blind, generates results comparable to KSVD's even though KSVD is a non-blind algorithm. Non-blind inpainting is a well developed technology that works decently on the removal of small



(236K)



(102K)

objects. Blind inpainting, however, is much harder since it demands automatic identification of the patterns that

requires inpainting, which, by itself is a very challenging problem. To the best of our knowledge, former methods are only capable of removing i.i.d. or simply structured impulse noise [9, 10, 5]. SSDA's capability of blind inpainting of complex patterns is one of this paper's major contributions.

V. A LEVEL SET APPROACH TO NOISE REMOVAL

Level Set Methods offer an appealing approach to noise removal. In particular, they exploit the fact that curves moving under their curvature smooth out and disappear. The idea is to view the pixel values as a topographic map; the intensity (somewhere between white and black) at each pixel is the height of the surface at that point. Suppose we then let each contour undergo motion by curvature. Then very small contours, corresponding to spikes of noise, will disappear quickly. Better yet, the boundaries will remain sharp, since they will not blur under this motion, and instead only move according to their curvature. Nonlinear models on the other hand can handle edges in a much better way than linear models can. One popular model for nonlinear image denoising is the Total Variation (TV)-filter, introduced by Rudin, Osher and Fatemi. This filter is very good at preserving edges, but smoothly varying regions in the input image are transformed into piecewise constant regions in the output image. Using the TV-filter as a denoiser leads to solving a 2nd order nonlinear PDE. Since smooth regions are transformed into piecewise constant regions when using the TV-filter, it is desirable to create a model for which smoothly varying regions are transformed into smoothly varying regions, and yet the edges are preserved. This can be done for instance by solving a 4th order PDE instead of the 2nd order PDE from the TV-filter. Results show that the 4th order filter produces much better results in smooth regions, and still preserves edges in a very good way. Some results showing the behavior of the 4th order model is shown:

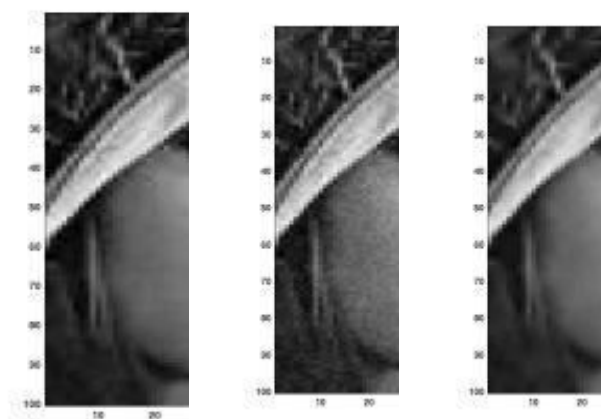


Fig2. Movies of Noise Removal under a Level Set Approach

VI. EXPLANATION/PREVIEW

One illustration of interface methods is the removal of noise from an image. Consider a gray-scale image, made

up of pixels which have some value between white (0) and black (255). To make life easy for a second, imagine a black letter on a white background. We'll adopt the usual convention, so that each pixel has a value of either 0 or 255. Now, let's imagine a lot of noise in the image; by noise, we mean pixels that are supposed to be black or white, but in fact have corrupted values somewhere between 0 and 255. As an example, see the figure above.

The idea is to view the pixel values as a topographic map; the intensity (somewhere between white and black) at each pixel is the height of the surface at that point. Suppose we then let each contour undergo motion by curvature. Then very small contours, corresponding to spikes of noise, will disappear quickly. Better yet, the boundaries will remain sharp, since they will not blur under this motion, and instead only move according to their curvature.

Of course, if you let the contours flow under the curvature, Grayson's theorem says that eventually everything will shrink and disappear. Instead we use a min/max flow; which turns the curvature flow on or off depending on the scale of the noise you want to remove. Some advantages of this approach are that:

- It stops automatically; if you apply it forever, it will clean the image, and then do nothing.
- It requires only local operations on pixels; that means, each pixel value is cleaned or left alone depending only on the basis of the neighboring pixels.

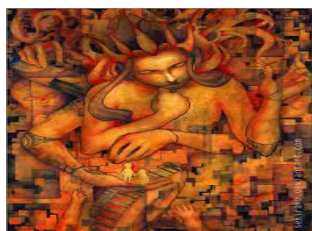


Fig 4. Image inpainting.

VII. PRIOR VS. LEARNED STRUCTURE

Unlike models relying on structural priors, our method's denoising ability comes from learning. Some models, for example BLS-GSM, have carefully designed structures that can give surprisingly good results with random parameter settings [23]. However, randomly initialized SSDA obviously can not produce any meaningful results. Therefore SSDA's ability to denoise and inpaint images is mostly the result of training. Whereas models that rely on structural priors usually have very limited scope of applications, our model can be adapted to other tasks more conveniently. With some modifications, it is possible to denoise audio signals or complete missing data (as a data preprocessing step) with SSDA. 4.2 Advantages and Limitations Traditionally, for complicated inpainting tasks, an inpainting mask that tells the algorithm which pixels correspond to noise and require inpainting is supplied a priori. However, in various situations this is time consuming or sometimes even impossible. Our approach, being blind, has significant advantages in such circumstances. This makes our method a suitable choice for fully automatic and noise pattern specific image

processing. The limitation of our method is also obvious: SSDA strongly relies on supervised training.

VIII. CONCLUSIONS

In our experiment, we find that SSDA can generalize to unseen, but similar noise patterns. Generally speaking, however, SSDA can remove only the noise patterns it has seen in the training data.

ACKNOWLEDGMENTS

I would also like to show my gratitude to Dr.K.Jayaraman, for sharing his pearls of wisdom with me.

I thank "anonymous" reviewers for their so called insights. I thank our Principal Dr.S.Arjunan, who provided insight and expertise that greatly assisted the research.

I thank Thiru S.Maniselvan for comments that greatly improved the manuscript.

REFERENCES

- [1] J. Xu, K. Zhang, M. Xu, and Z. Zhou. An adaptive threshold method for image denoising based on wavelet domain. Proceedings of SPIE, the International Society for Optical Engineering, 7495:165, 2009.
- [2] X. Lu, H. Yuan, P. Yan, Y. Yuan, L. Li, and X. Li. Image denoising via improved sparse coding. Proceedings of the British Machine Vision Conference, pages 74–1, 2011.
- [3] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online dictionary learning for sparse coding. Proceedings of the 26th Annual International Conference on Machine Learning, pages 689–696, 2009.
- [4] Malladi, R., and Sethian, J.A., Image Processing via Level Set Curvature Flow, Proceedings of the National Academy of Sciences, Vol. 92(15), pp. 7046--7050, July 1995.
- [5] <http://ieeexplore.ieee.org/document/6320763>
- [6] <http://ieeexplore.ieee.org/xpl/RecentIssue.jsp?reload=true&punumber=83>
- [7] A. Dosovitskiy and T. Brox. Inverting visual representations with convolutional networks. arXiv preprint arXiv:1506.02753, 2015.
- [8] L. Gatys, A. S. Ecker, and M. Bethge. Texture synthesis using convolutional neural networks. In NIPS, 2015.
- [9] J. S. Ren, L. Xu, Q. Yan, and W. Sun. Shepard convolutional neural networks. In NIPS, 2015.
- [10] Mahendran and A. Vedaldi. Understanding deep image representations by inverting them. In CVPR, 2015.
- [11] C. Dong, C. C. Loy, K. He, and X. Tang. Image super-resolution using deep convolutional networks. IEEE Trans. Pattern Anal. Mach. Intell., 38(2):295–307, 2016.
- [12] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In Proc. IEEE Conf. Comp. Vis. Patt. Recogn., volume abs/1512.03385, 2016.