IMAGE DE-NOISING USING DEEP NEURAL NETWORK

Jason Wayne

Department of Computer Engineering, University of Sydney, Sydney, Australia

ABSTRACT

Deep neural network as a part of deep learning algorithm is a state-of-the-art approach to find higher level representations of input data which has been introduced to many practical and challenging learning problems successfully. The primary goal of deep learning is to use large data to help solving a given task on machine learning. We propose an methodology for image de-noising project defined by this model and conduct training a large image database to get the experimental output. The result shows the robustness and efficient our our algorithm.

KEYWORDS

Deep Neural Network, Image De-noising, Machine Learning, Large-scale Database

1. INTRODUCTION

Image de-noising issue could be defined as the problem of mapping from a noisy image to a noise-free image. Various methods have been proposed for image de-noising. One approach is linear or non-linear filtering methods which are a relatively simple approach based on smoothing, such as Median filtering which replace each pixel with the median of the value of a set of neighboring pixels[1], linear smoothing and wiener filtering. Another one is methods based on wavelet or dictionary decompositions of the image. Wavelet decompositions is to transfer image signals to an alternative domain where they can be more easily separated from the noise, such as BLS-GSM[2]. The dictionary-based method is to de-noise by approximating the noisy patch using a sparse linear combination of atoms, including KSVD[3] which is an iterative algorithm that learns a dictionary on the noisy image at hand, NLSC[4] which is one of the best currently available de-noising algorithms in terms of quality of the results, but requires long computation times. The last one is methods based on global image statistics or other image properties, such as self-similarities. Typical schemes include EPLL[5] and BM3D[6] which are often considered the state-of-the-art in image de-noising.

While these models have been successfully 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. A few of deep models have also been applied to image de-noising [7-11].

Deep learning is an emerging approach within the machine learning research community [12]. Deep learning algorithms have been proposed in recent years to move machine learning systems towards the discovery of multiple levels of representation. Learning algorithms for deep architectures are centered on the learning of useful representations of data, which are better suited to the task at hand, and are organized in a hierarchy with multiple levels. There are several motivations for deep architectures: Brain inspiration (several areas of the brain are organized as a deep architecture); Cognitive arguments and engineering arguments (humans often organize ideas and concepts in a modular way, and at multiple levels.); Sharing of statistical strength for multi-task learning; Computational complexity [13]. In fact, it was found recently that the features learned in deep architectures resemble those observed in the areas V1 and V2 of visual cortex [14], and that they become more and more in variant to factors of variation in higher layers. Learning a hierarchy of features increases the ease and practicality of developing representations that are at once tailored to specific tasks, yet is able to borrow statistical strength from other related tasks. Finally, learning the feature representation can lead to higher-level (more abstract, more general) feature that are more robust to unanticipated sources of variance extant in real data.

In this paper, we present an algorithm for image restoration task that combines sparse coding and deep networks pre-trained with de-noising auto-encoder (DAE) defined by this model, and show that by training on large image databases we are able to outperform the current state-of-the-art image de-noising methods. Our algorithms will first learn a large basis functions, and then reconstruct any new input image using a weighted combination of a few of these basis functions. The weights of these basis functions then give a slightly higher-level and more succinct representation of the input; this representation can then be used in image restoration task.

2. OUR METHOD

The basic framework for our models is auto-encoder[12]. A basic auto-encoder is comprised of an encoder function $h(\cdot)$ maps an original input $x \in R^d$ to some hidden layers and get the representation of $h(x) \in R^{d_n}$, whereas a decoder $g(.)$ maps this hidden representation back to the reconstructed version of x, therefore $g(h(x)) \approx x$. The parameters of the auto-encoders are settled to minimize the reconstruction error, measured by the absolute loss defined as $L(x, g(h(x)))$. Here, we use squared error defined as:

$$L(x, g(h(x))) = \left|\left(\left|x - g(h(x))\right|\right)\right|^2 \qquad (1)$$

De-noising Auto-encoders [9] incorporate a slight modification to this setup and corrupt the inputs before mapping them into the hidden representation. They are trained to reconstruct the original input x from its corrupted version $\tilde{x}$ by minimizing:

$$L(x, g(h(x))) = \left|\left(\left|x - g(h(x))\right|\right)\right|^2 \qquad (2)$$

Typical choices of corruption include additive white Gaussian noise or binary masking noise. In this work, we use the former one with standard deviation .This is a rational and non-bias choice for natural images captured by a digital camera. The DAE architecture is shown in Figure 1.

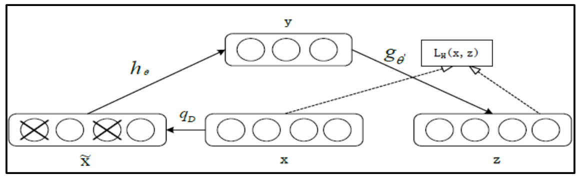

Figure 1. The DAE Architecture

The auto-encoder then maps it to y and attempts to reconstruct x via decoder, resulting in the reconstructed z . Reconstruction error is measured by loss $L_H(x,z)$. We could then use the trained model to get the representation of the original images.

3. FEATURE LEARNING

We performed all our experiments on grey-scale images, but there is no difficulty in generalizing to colored images. Image de-noising aim to map a noisy image to a cleaner version. However, the complexity of a mapping from images to images is large, so in practice we chop the image into possibly overlapping patches and learn a mapping from a noisy patch. to a clean patch. To de-noise a given image, all image patches are de-noised separately by that map. The de-noised image patches are then combined into a de-noised image.

To evaluate the model we use a set of 60, 000 images from the CIFAR-bw data set. Our system performs the following steps to feature extraction and image restoration: a. Extract random patches from training images. b. Apply a pre-processing stage to the patches. c. Learn a feature-mapping using stacked DAE learning algorithm. d. Train a image restoration algorithm. We will now describe the components of this pipe line and its parameters in more detail.

3.1. Feature Learning

We have tested our approach on a benchmark image sets, namely: CIFAR-bw: a gray-scale version of the CIFAR-100[17].The CIFAR-100 is labeled subsets of the 80 million tiny images dataset. They were collected by Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. The Sample images are shown in Figure 2.

Figure 2. Sample Images

Next, we get corrupted images. Assuming $\tilde{x}$ is the observed noisy image and x is the original noise-free image, we could then formulate the image corruption process as:

$$\tilde{x} = q(x) \tag{3}$$

Where $q: R^n \to R^n$ is an arbitrary stochastic corrupting process that corrupts the input x. For most of our experiments, we use AWG noise with $\sigma = 25$. However, we also show results for other noise levels.

Finally, as mentioned above, the system begins by extracting random sub-patches from input images. Each patch has dimension $N = n \times n$. Each $n \times n$ patch can be represented as a vector in R^N of pixel intensity values. We then construct a dataset of m randomly sampled patches, $X = \{x^{(1)}, x^{(2)}, \cdots, x^{(m)}\}$ where $x^{(i)} \in R^N$. Given this dataset, we could apply the pre-processing and unsupervised deep learning steps.

3.1.1. Image Pre-processing

Data pre-processing plays a important role in many deep learning algorithms. In practice, many methods work best after the data has been normalized. In this work, we assume that every patch is normalized by simple re-scaling, subtracting the mean and dividing by the standard deviation of its elements. For visual data, this corresponds to local brightness and contrast normalization.

3.1.2. Feature Extraction and Selection

The basic building block of our framework is a one-layer DAE. The DAE tries to learn an essential function $h_{w,b}(\tilde{x}) \approx x$, which minimizes the squared reconstruction loss:

$$J(W,b) = \frac{1}{m} \sum_{i=1}^{m} \left\| h_{w,b}\left(\tilde{x}^{(1)}\right) - x^{(i)} \right\|^2 + \frac{\lambda}{2} \sum W + \beta KL \tag{4}$$

Where

$$KL = KL\left(\rho \,\|\, \vartheta_j\right) = \rho \log \frac{\rho}{\vartheta_j} + (1-\rho) \log \frac{1-\rho}{1-\vartheta_j}, \qquad \vartheta_j = \frac{1}{m} \sum_{i=1}^{m} h_{w,b}\left(\tilde{x}^{(1)}\right) \tag{5}$$

The first term in the definition of $J(W,b)$ is an average sum-of-squares error term. The second term is a regularization term (also called a weight decay term) that tends to decrease the magnitude of the weights, and helps prevent over fitting. The third term is a sparsity penalty term then enforce the average activation of hidden is a small value close to zero. λ and β controls the weights of the penalty term. We choose to minimize the squared error since it is monotonically related to the PSNR, which is the most commonly, used measure of image quality. Thus minimizing the squared error will maximize PSNR values. One-layer DAE is a computational unit that takes as input $\tilde{x}$, and outputs

$$h(\tilde{x}) = f(wx + b) \tag{6}$$

Where $f: R \to R$ is activation function. In this work, we will choose $f(\cdot)$ to be Sigmiod function: $f(z) = \dfrac{1}{1+\exp(-z)}$. We evaluate different hidden layers, and find that it is not always beneficial to add hidden layers. A possible explanation is that SDAE with more hidden layers become more difficult to learn. Indeed, each hidden layer adds non-linearities to the model. It is therefore possible that the error landscape is complex and that stochastic gradient descent gets stuck in a poor local optimum from which it is difficult to escape. In the meantime, we try different patch sizes and find that higher noise level generally requires larger patch size.

3.2. Image Restoration

We use standard testing images that have been used to evaluate other de-noising algorithms as the testing set. To de-noise images, we decompose a given noisy image into overlapping patches. We then normalize the patches (see Section 3.1.1), de-noise each patch separately and perform the inverse normalization on the de-noised patches. The de-noised image is obtained by placing the de-noised patches at the locations of their noisy counterparts, then averaging on the overlapping regions.

After de-noising an image, we would like to know: How good is the de-noising result. A possible solution to this problem would be to rely on human evaluation of the image quality. However, this solution is too inconvenient for many applications. Hence, one is interested in automatic image quality assessment and in particular in objective image quality metrics that correlate with subjective image quality.

There are many image quality metrics, include peak signal-to-noise ration (PSNR [18]), structural similarity index (SSIM [19]), information-content weighted PSNR (IW-PSNR [20]), the information fidelity criterion (IFC[21]), DIIVINE[22], LBIQ[23], BIQI[24], etc. We employ PSNR which is the most commonly used metric[25] to quantify de-noising results. The PSNR is computed by(σ_e^2 is the mean squared error):

$$PSNR = 20\log_{10}\left(\frac{255}{\sigma_e^2}\right) \tag{7}$$

4. EXPERIMENTS

In this section, we demonstrate the results achieved by applying the above methods on several test images. Before we present de-noising result, we first show visualizations of the learned feature representations. The bases learned by SDAE are shown in Figure 4 for 8 pixel receptive fields. We have compared our method to two well-known and widely-available de-noising algorithms: KSVD [3] (a dictionary-based method) and BM3D [6] (a block matching procedure). Table 1 compares our method against KSVD and BM3D on the test set of 4 standard test images. The result from left to right is KSVD, BM3D and our method.

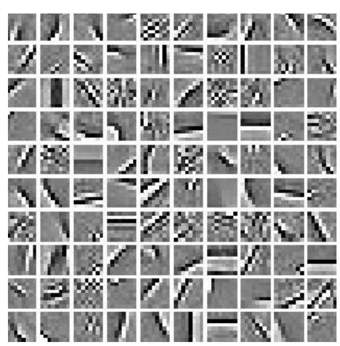

Figure 3. Randomly Selected Bases

σ	Lena			Barbara			Boats			House		
10	35.5	**35.9**	35.8	34.4	**34.9**	34.1	33.6	**33.9**	33.8	35.9	**36.7**	35.9
25	31.3	32.2	**32.2**	29.6	**30.9**	29.7	29.2	**30.0**	29.9	32.1	**32.9**	32.5
50	27.4	28.9	**29.2**	25.2	**27.2**	25.3	25.9	26.8	**26.9**	27.4	**29.7**	29.6
75	24.8	27.1	**27.6**	22.5	**25.1**	23.4	23.6	25.1	**25.8**	24.5	27.4	**27.8**

The experimental results show that BM3D perform better than other methods on average. Analyzing the outcomes of those experiments, we conclude than BM3D based on knowledge about the image to be de-noised perform well on images with regular structure (e.g., image “Barbara”), where as our methods based on knowledge about all images perform well on images with complex structures(e.g., image “Lena”)or high noise levels.

5. CONCLUSION AND FUTURE RESEARCH

In this paper, we have described an algorithm for image de-noising task defined by the deep learning framework. We have compared the results achieved by our approach against other algorithms, and show that by training on large image databases we are able to outperform the current state-of-the-art image de-noising methods.

In our future work, we would like to explore the possibility of adapting the proposed approach to various other applications such as de-noising and in painting of text and audio and also find out some math-related methods[26] to optimize the proposed algorithm. It is also meaningful to investigate into the effects of different hyper parameter settings on the learned features.

ACKNOWLEDGEMENTS

The authors would like to thank my adviser for her endless help and care.

REFERENCES

[1] F. Luisier, T. Blu and M. Unser, "A new SURE approach to image denoising: Inter scale ortho normal wavelet thres holding", Image Processing, IEEE Transactions on, vol. 16, no. 3, (2007), pp. 593-606.
[2] J. Portilla, V. Strela, M.J. Wainwright and E.P. Simoncelli, "Image denoising using scale mixtures of Gaussians in the wavelet domain", Image Processing, IEEE Transactions on, vol. 12, no. 11, (2003), pp. 1338–1351.
[3] M. Aharon, M. Elad and A. Bruckstein, "K-svd: An algorithm for designing over complete dictionaries for sparse representation", IEEE Transactions on Signal Processing (TIP), vol. 54, no. 11, (2006), pp. 4311-4322.
[4] J. Mairal, F. Bach, J. Ponce, G. Sapiro and A. Zisserman, "Non-local sparse models for image restoration", In International Conference on Computer Vision (ICCV). IEEE, (2010).
[5] D. Zoran and Y. Weiss, "From learning models of natural image patches to whole imagerestoration", In International Conference on Computer Vision (ICCV). IEEE, (2011).
[6] Dai, L., Zhang, Y., Li, Y., & Wang, H. (2014, April). MMW and THz images denoising based on adaptive CBM3D. In Sixth International Conference on Digital Image Processing (pp. 915906-915906). International Society for Optics and Photonics.
[7] Y. Tang, R. Salakhutdinov, G. Hinton and R. Boltzmann, "Machines for recognition and denoising", Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. IEEE, (2012), pp. 2264-2271.
[8] P. Vincent, H. Larochelle, Y. Bengio, et al., "Extracting and composing robust features with denoising auto encoders", Proceedings of the 25th international conference on Machine learning. ACM, (2008), pp. 1096-1103.
[9] P. Vincent, H. Larochelle, I. Lajoie, et al., "Stacked denoising auto encoders: Learning useful representations in a deep network with a local denoising criterion", The Journal of Machine Learning Research, vol. 11, (2010), pp. 3371-3408.
[10] H. C. Burger, C. J. Schuler and S. Harmeling, "Image denoising: Can plain neural networks compete with bm3d?", IEEEConf. Comput. Vision and Pattern Recognition, (2012), pp. 2392–2399.
[11] H. C. Burger, C. J. Schuler and S. Harmeling, "Image denoising with multi-layer perceptrons, part 1: comparison with existing algorithms and with bounds", arXiv, (2012), pp. 1211-1544.
[12] Y. Bengio, "Learning deep architectures for A.I.", Foundations and Trends in Machine Learning, vol. 2, no. 1, (2009), pp. 1-127.
[13] Y. Bengio and A. Courville, "Deep Learning of Representations", Handbook on Neural Information Processing, Springer Berlin Heidelberg, (2013), pp. 1-28.
[14] H. Lee, C. Ekanadham and A. Ng, "Sparse deep belief net model for visual area V2. In NIPS'07, Cambridge, MIT Press, MA, (2008), pp. 873–880.
[15] R. Raina, A. Battle, H. Lee, et al., "Self-taught learning: transfer learning from unlabeled data", Proceedings of the 24th international conference on Machine learning. (2007), pp. 759-766
[16] Y. Bengio, P. Lamblin, D. Popovici, et al., "Greedy layer-wise training of deep networks", Advances in neural information processing systems, vol. 19, (2007), pp. 153.
[17] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images", Master's thesis, Department of Computer Science, University of Toronto, (2009).
[18] Q. Huynh-Thu, M. Ghanbari, "Scope of validity of PSNR in image/video quality assessment", Electronics letters, vol. 44, no. 13, (2008), pp. 800-801.
[19] Z. Wang, A. C. Bovik, H. R. Sheikh, et al., "Image quality assessment: From error visibility to structural similarity", Image Processing, IEEE Transactions on, vol. 13, no.4, pp. 600-612.
[20] Z. Wang and Q. Li, "Information content weighting for perceptual image quality assessment", Image Processing, IEEE Transactions on, vol. 20, no. 5, (2011), pp. 1185-1198.

[21] H. R. Sheikh, A. C. Bovik, G. De Veciana, “An information fidelity criterion for image quality assessment using natural scene statistics”, Image Processing, IEEE Transactions on, vol. 14, no. 12, (2005), pp. 2117-2128.

[22] A. K. Moorthy and A. C. Bovik, “Blind image quality assessment: From natural scene statistics to perceptual quality”, Image Processing, IEEE Transactions on, vol. 20, no. 12, (2011), pp. 3350-3364.

[23] H. Tang, N. Joshi and A. Kapoor, “Learning a blind measure of perceptual image quality Computer Vision and Pattern Recognition (CVPR)”, IEEE Conference on. IEEE, (2011), pp. 305-312.

[24] A. K. Moorthy and A. C. Bovik, “Blind image quality assessment: From natural scene statistics to perceptual quality”, Image Processing, IEEE Transactions on, vol. 20, no. 12, (2011), pp. 3350-3364.

[25] Wang, Haoxiang, Ferdinand Shkjezi, and Ela Hoxha. "Distance metric learning for multi-camera people matching." In Advanced Computational Intelligence (ICACI), 2013 Sixth International Conference on, pp. 140-143. IEEE, 2013.

[26] Chen, Guang-Sheng, Hui-Yu Tang, De-Quan Zhang, Yun-Xiu Jiao, and Hao-Xiang Wang. "Existence of three solutions for a nonlocal elliptic system of-Kirchhoff type." Boundary Value Problems 2013, no. 1 (2013): 1-9.