Image Denoising Using DNN

Unnikrishnan N

Indian Institute of Science

unnikrishnann@iisc.ac.in

August 10, 2019

Overview

Image Denoising - Classical Methods

Signal Model

$$\mathbf{Y} = \mathbf{X} + \mathbf{N}$$

where $\mathbf{X}$ is an image corrupted with additive noise N

Objective

Recover X from noisy measurements Y

Techniques for Recovery

1. Spatial domain techniques
2. Transform domain techniques

Dictionary Learning

Signal Model

$$\mathbf{Y} = \mathbf{AX}$$

where Y is a set of measurements, A is dictionary and X is sparse representation of Y

Problem formulation

$$\min_{\mathbf{A},\mathbf{X}} \|\mathbf{Y} - \mathbf{AX}\|_{\mathbf{F}}^{2} \quad \text{such that } \|\mathbf{X_j}\|_{\mathbf{0}} \leq \mathbf{k} \quad \forall \mathbf{j} \in [\mathbf{l}]$$

Find both **dictionary** and **sparse representation** from measurements

where $\mathbf{Y} \in \Re^{mxl}$ is l no of the measured vectors
$\mathbf{A} \in \Re^{mxn}$ Dictionary with unit norm columns with $m < n$
$\mathbf{X} \in \Re^{nxl}$ with sparsity of each columns $\leq k(k < m < n)$

Dictionary Learning Algorithms

Popular Iterative Algorithms

1. MOD
2. K-SVD

- Once the dictionary is learned, use any of the compressive sensing recovery algorithms to recover the Denoised image.
- If the dictionary is constructed from multiple images, that can be widely used for denoising.

Quality Metrics

PSNR - Peak SNR

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [X(i,j) - Y(i,j)]^2$$

where X and Y are true and estimated images

m, n are no of rows and columns of images X and Y

$$\mathbf{PSNR} = \mathbf{10} \cdot \log_{\mathbf{10}} \left(\frac{\mathbf{MAX_x^2}}{\mathbf{MSE}} \right)$$

- Higher the value of PSNR $\Rightarrow$ better the reconstruction.

Quality Metrics -Continues

SSIM (Structural SIMilarity) Index

A Combination of luminance, contrast and structure comparisons are used

$$l(x,y) = \frac{2\mu_x\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1}$$

$$c(x,y) = \frac{2\sigma_x\sigma_y + c_2}{\sigma_x^2 + \sigma_y^2 + c_2}$$

$$s(x,y) = \frac{\sigma_{xy} + c_3}{\sigma_x\sigma_y + c_3}$$

$$\text{SSIM}(x,y) = \left[l(x,y)^{\alpha} \cdot c(x,y)^{\beta} \cdot s(x,y)^{\gamma} \right]$$

$\alpha, \beta, \gamma > 0$ parameters are chosen based on the importance of each metric.
where μ_x, μ_y are average intensity of the images x and y.
σ_x, σ_x and σ_{xy} standard deviation of x, y and cross covariance of x and y
c1, c2, c3 are non zero constants used for avoiding instability.

DNN Based Image Denoising

Universal Approximation Theorem

Theorem

Let $\phi : \Re \to \Re$ be non-constant, bounded and continuous function. Let I_m denotes m dimensional hypercube $[0,1]^m$. Let $C(I_m)$ be space of continuous real valued functions on I_m. Then given any $\epsilon > 0$ and any function $f \in C(I_m)$, $\exists\ n \in N$, real constants $v_i, b_i \in \Re$ and real vectors $w_i \in \Re^m\ \forall i \in [n]$ such that we may define:

$$F(\mathbf{x}) = \sum_{i=1}^{n} v_i \varphi\left(\mathbf{w_i^T x} + b_i\right)$$

as an approximate realization of the function f

$$|F(\mathbf{x}) - f(\mathbf{x})| < \varepsilon \quad \forall \mathbf{x} \in I_m$$

Different Types Of Learning

Supervised Learning

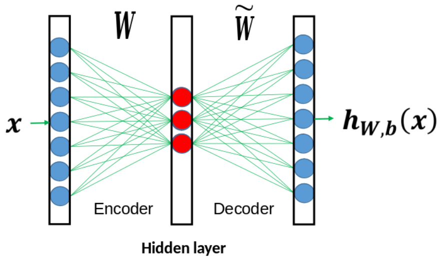
Here data is given in the form of examples with labels. Allowing the algorithm to predict the label for each example, and giving it feedback as to whether it predicted it correctly or not.

Unsupervised Learning

Here data is given in the form of examples without labels. This mechanism learns properties of the data. From there it can learn to group or organise the data.

Reinforcement Learning

Autoencoder -Unsupervised



where **W** and **b** are weight matrix and bias

z and **x** are input and output of each layer

ϕ is activation function

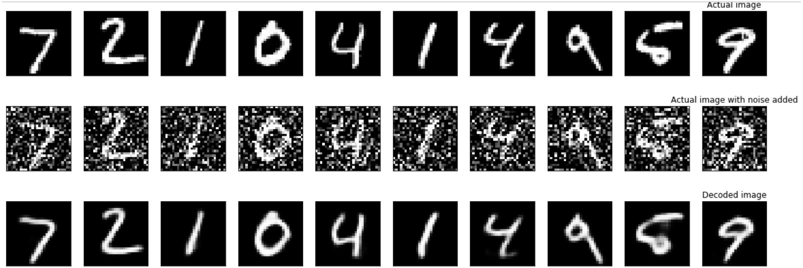
Neural Network

$\mathbf{z} = \phi(\mathbf{Wx} + \mathbf{b})$ Overall output $\mathbf{h}_{\mathbf{W},\mathbf{b}}(\mathbf{x})$

Objective

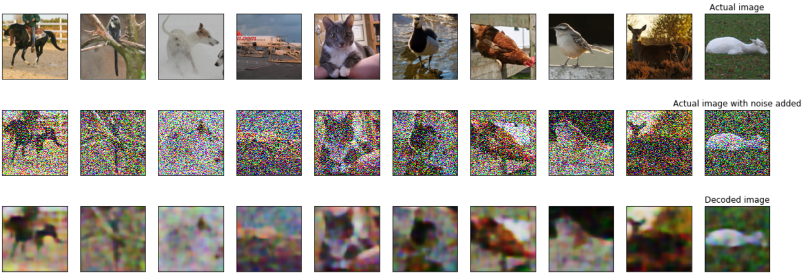
Minimise average cost $\mathbf{J}(\mathbf{W}, \mathbf{b})$ by iteratively update weights using back propagation algorithms

With Convolutional Neural Network.



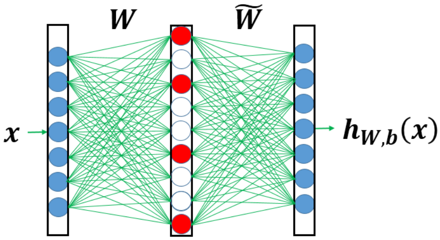
1. PSNR (avg) = 20dB
2. SSIM (avg) = 0.88

Results - Unsupervised continue



1. PSNR (avg) = 17dB
2. SSIM (avg) = 0.55

Sparse Autoencoder - Regularized Autoencoder

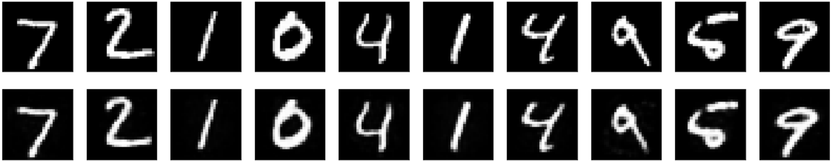


Overall cost function with L1 regularizer

$$J(W,b) = \left[\frac{1}{m}\sum_{i=1}^{m}\left(\frac{1}{2}\left\|h_{W,b}\left(x^{(i)}\right)-y^{(i)}\right\|^{2}\right)\right]+\frac{\lambda}{2}\sum_{l=1}^{n_l-1}\sum_{i=1}^{s_l}\sum_{j=1}^{s_{l+1}}\left|W_{ji}^{(l)}\right|$$

where m is no of training inputs
n_l, s_l are no of layers and no of neurons in l^{th} layer
λ is the decay factor.

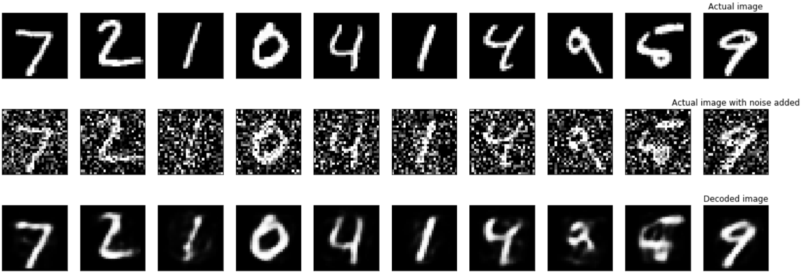
Sparse Autoencoder - Results (without noise)



Acivation function used = sigmoid.

1. PSNR (avg) = 30dB
2. SSIM (avg) = 0.94

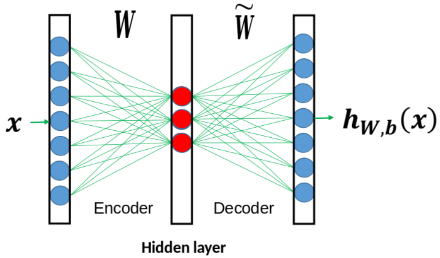
Sparse Autoencoder - Results (with noise)



Acivation function used = sigmoid.

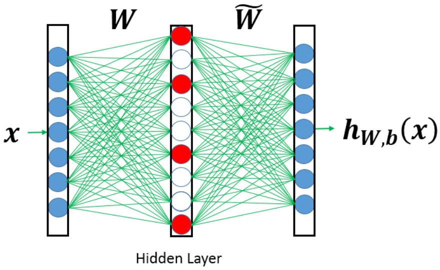
1. PSNR (avg) = 19dB
2. SSIM (avg) = 0.79

Dictionary Learning From Sparse Autoencoder



Autoencoder

If we use linear activation function and cost function as MSE then $\widetilde{\mathbf{W}}$ becomes **PCA**



Sparse Autoencoder

If we use linear activation function with unit norm constraint then $\widetilde{\mathbf{W}}$ becomes **dictionary**

Conclusion And Future Scope

Conclusion

1. Image denoising can be done using autoencoders and sparse autoencoders
2. Autoencoders learn approximate PCA
3. Sparse autoencoders learn approximate dictionary

Future Scope

1. Add Unit norm constraint to weights along with regularization criteria in sparse autoencoder

References

- Elad, Michael,"Sparse and Redundant Representations - From Theory to Applications in Signal and Image Processing",2010
- Wang; Zhou; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. (2004). "Image quality assessment: from error visibility to structural similarity". IEEE Transactions on Image Processing. 13 (4).
- Junyuan Xie, Linli Xu, Enhong Chen, "Image Denoising and Inpainting with Deep NeuralNetworks", NIPS 2012
- Andrew N G, CS294A Lecture notes-Sparse Autoencoder, 2011
- https://blog.keras.io/building-autoencoders-in-keras.html
- Snigdha Tariyal, Angshul Majumdar,Richa Singh and Mayank Vatsa,"Deep Dictionary Learning", IEEE Access date of publication December 9, 2016
- https://en.wikipedia.org/wiki/Universal approximation theorem

The End