DnCNN Image Denoising

Course Project, CS736: Medical Image Computing
Anagha Savit, Ishan Kapnadak, Vineet Gala

Introduction

- Image denoising is an indispensable step in many practical applications. The goal of image denoising is to recover a clean image x from a noisy observation y which follows an image degradation model $y = x + v$.
- We investigate the construction of an end-to-end trainable deep CNN for Gaussian denoising.
- Specifically, residual learning and batch normalization are utilized to speed up the training process as well as boost the denoising performance.
- In other words, the proposed DnCNN implicitly removes the latent clean image with the operations in the hidden layers.
- The batch normalization technique is further introduced to stabilize and enhance the training performance of DnCNN.

Existing approaches

- It has been claimed earlier that CNNs have better representation power than MRF models, hence CNNs might be better for denoising as well
- TNRD (trainable non-linear reaction diffusion) model has been proposed which can be expressed as a feed-forward deep network
- BM3D (Block-matching and 3D filtering) is another model which is a nonlocal self-similarity model and is considered as state-of-the-art denoising method
- TNRD has been shown to match the performance of BM3D
- Their performances are inherently restricted to the specified forms of the prior model

A higher level approach

- When the noise level σ is unknown, the denoising method should enable the user to adaptively make a trade-off between noise suppression and texture protection. The fast and flexible denoising convolutional neural network (FFDNet) was introduced to satisfy these desirable characteristics.
- FFDNet builds upon the work done for DnCNN with the inclusion of preprocessing and postprocessing layers before and after the same nonlinear mapping of DnCNN.
- FFDNet accomplishes:

  (i) the ability to handle a wide range of noise levels (i.e., [0, 75]) effectively with a single network

  (ii) the ability to remove spatially variant noise by specifying a non-uniform noise level map

Methodology

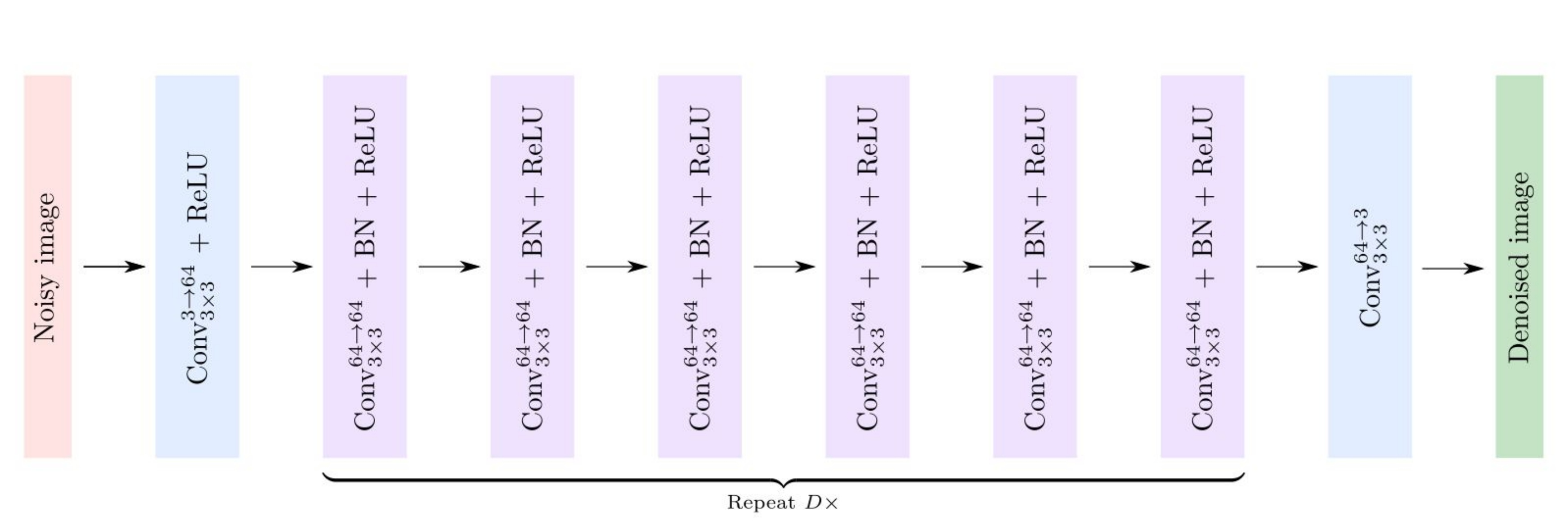
Residual Learning and Batch Normalization

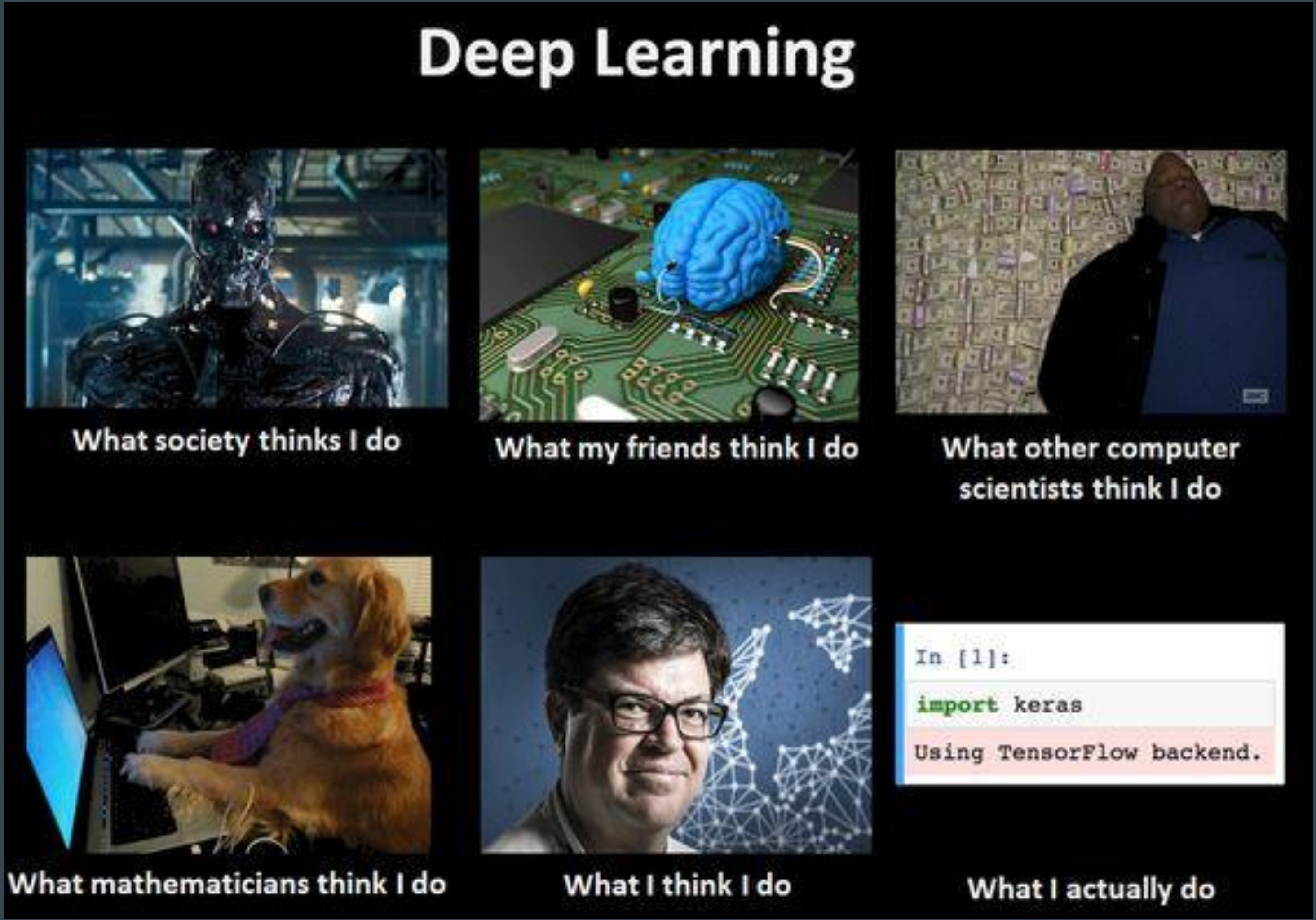
Assuming the input to our DnCNN is the noisy observation $\mathbf{y} = \mathbf{x} + \mathbf{v}$, where $\mathbf{x}$ is the noise free image and $\mathbf{v}$ is the added noise. We train the model to estimate the residual mapping $\mathcal{R}(\mathbf{y}) \approx \mathbf{v}$, and we can use that to extract the denoised prediction for $\mathbf{x}$. The loss function for training on N image pairs, parameterized by $\mathbf{\Theta}$, is modeled as:

$$\ell(\mathbf{\Theta}) = \frac{1}{2N} \sum_{i=1}^{N} \|\mathcal{R}(\mathbf{y}_i; \mathbf{\Theta}) - (\mathbf{y}_i - \mathbf{x}_i)\|_F^2$$

Batch Normalization is used to reduce training time, reduce the internal covariate shift, and better the denoising.

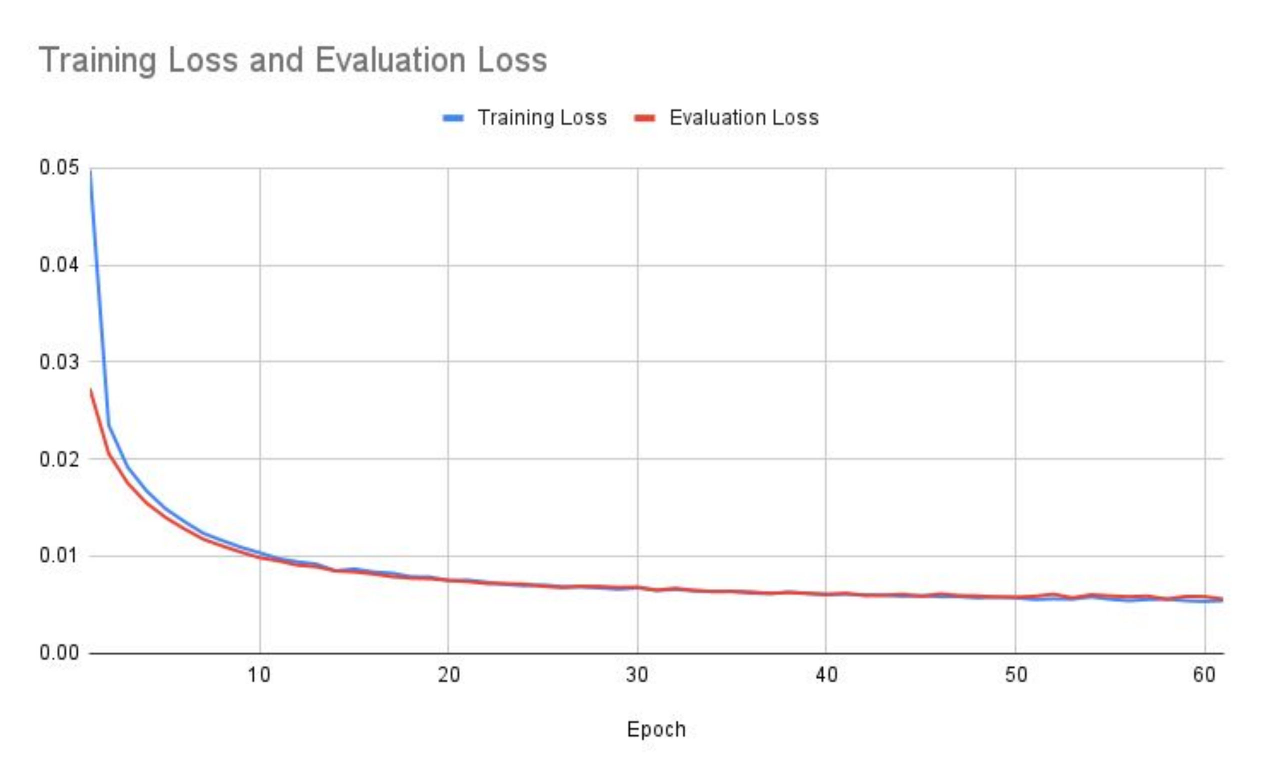
CNN Architecture




Deep Learning
What society thinks I do
What my friends think I do
What other computer scientists think I do
What mathematicians think I do
What I think I do
In [1]:
import keras
Using TensorFlow backend.
What I actually do

Results

Training and Evaluation Loss v/s Epochs



As we can see, the training and evaluation losses both decrease over epochs and have both converged after around 60 epochs of training.

Some sample results

Almost IID noise estimate

Some sample results

Not an IID noise estimate

Conclusion and Future Directions

What went right and what went wrong

Our methodology works very well on quite a few training images, such as the noisy image of the dog shown on the previous slide. Thus, our CNN model has adapted to the denoising task at hand and performs satisfactorily.

However, the model doesn’t perform as well as expected on some other images such as the image of the lady shown previously. We believe that the reason behind this degradation in performance is that the noise level present in the noisy test image is much higher than the noise level that the CNN model was trained on. Hence, the network is not able to adapt itself to denoise a noisier image.

What next?

To deal with the case where the noise level lies outside the training range, we plan to use a “Bias-free” CNN which has been shown to be able to generalise to all noise levels despite only being trained on low noise levels. A bias-free CNN removes all bias terms from the neural network. The resulting model is thus locally linear and performs quite well in denoising tasks.

The image degradation model for Gaussian denoising can be converted to a single image super-resolution (SISR) problem; analogously, the JPEG image deblocking problem can be modeled by the same image degradation model