BAYESIAN ENSEMBLE LEARNING FOR MEDICAL IMAGE DENOISING

by

Hyuntaek Oh
University of Kansas, 2012

Submitted to the graduate degree program in Bioengineering and the
Graduate Faculty of the University of Kansas
In partial fulfillment of the requirements for the degree of
Master of Science.

_____

Chairperson: Dr. Brian Potetz

_____

Dr. Luke Huan

_____

Dr. Shannon Blunt

_____

Dr. Arvin Agah

Date Defended: 2012/ 08/ 15

The Thesis Committee for Hyuntaek Oh

certifies that this is the approved version of the following thesis:

BAYESIAN ENSEMBLE LEARNING FOR MEDICAL IMAGE DENOISING

_____

Chairperson Dr. Brian Potetz

Date approved: 2012/ 08/ 15

Abstract

Medical images are often affected by random noise because of both image acquisition from the medical modalities and image transmission from modalities to workspace in the main computer. Medical image denoising removes noise from the CT or MR images and it is an essential step that makes diagnosing more efficient. Many denoising algorithms have been introduced such as Non-local Means, Fields of Experts, and BM3D.

In this thesis, we implement the Bayesian ensemble learning for not only natural image denoising but also medical image denoising. The Bayesian ensemble models are Non-local Means and Fields of Experts, the very successful recent algorithms. The Non-local Means presumes that the image contains an extensive amount of self-similarity. The approach of the Fields of Experts model extends traditional Markov Random Field model by learning potential functions over extended pixel neighborhoods. The two models are implemented, and image denoising is performed on both natural images and MR images. For MR images, we used two noise distributions, Gaussian and Rician. The experimental results obtained are used to compare with the single algorithm, and discuss the ensemble learning and their approaches.

Acknowledgement

First of all, I would like to express my deepest gratitude to my advisor, Dr. Brian Potetz, for the continuous support of my Master degree and research, for this patience, motivation, and immense knowledge. His guidance helped me in all the time of research and writing this thesis. It would not have been possible to write this thesis without all this support and guidance.

Besides my advisor, I would like to thank the rest of my thesis committee: Dr. Luke Huan, Dr. Shannon Blunt and Dr. Arvin Agah for serving on my thesis committee.

Last but not the least, I would like to thank my family for supporting me throughout all my studies at the University of Kansas.

Table of Contents

List of Figures

List of Table

1 Introduction

1.1 Medical images

Medical images obtained from Computed Tomography (CT) and Magnetic Resonance Imaging (MRI) are the most common medical images for diagnosis. There are other types of medical images such as ultrasound, angiography, and Positron Emission Tomography (PET). Medical images are often affected by noise because of both image acquisition from the medical modalities and image transmission from modalities to workspace in the main computer system. This noisy usually affects the visual quality of the original images so image denoising always has been issued in the medical image processing.

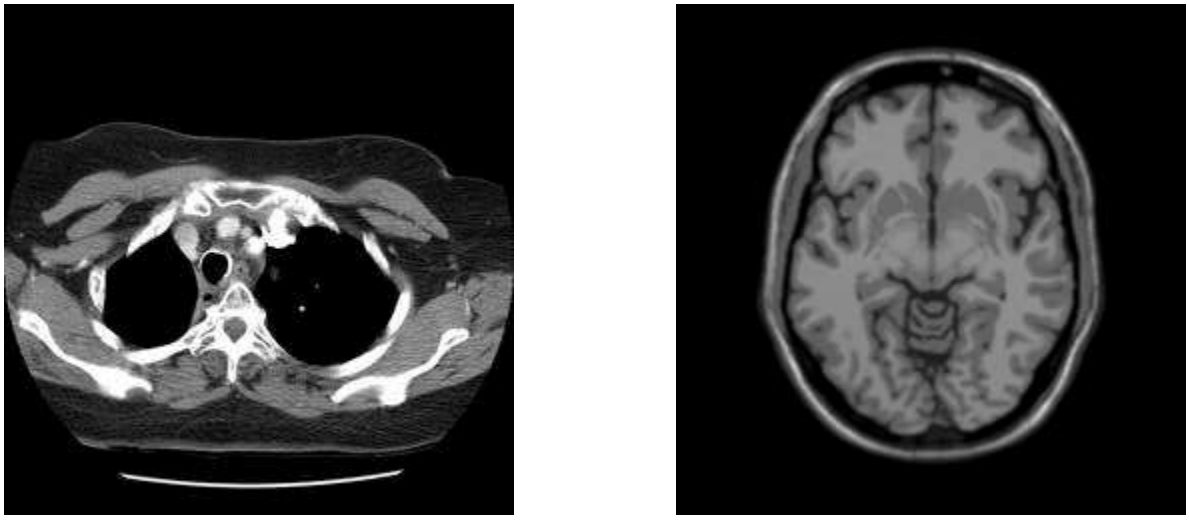


Figure 1.1. Sample medical images
(Left: Chest CT image from NBIA (National Biomedical Imaging Archive),
Right: Brain MRI image from BrainWeb Database)

1.2 Motivation

Medical image denoising removes noise from the CT, MRI, or ultrasound images and it is essential step that makes diagnosing more efficient for doctors. Several denoising algorithms have been introduced such as total variation minimization [1], Wiener filtering [2], Sparse coding [3], etc. Recently, Non-local means [4] , Fields of Experts [5] [6] and BM3D [7] have been proposed for promising denoising algorithm. One striking aspect of image denoising research is that a wide array of denoising strategies have remained popular, and in spite of vastly different approaches, many of these algorithms produce reasonably similar performance in terms of peak signal-to-noise ratio (PSNR). For example, Fields of Experts (FoE) pursues an entirely parametric approach, by training Markov random fields with large 5x5 cliques to capture the statistics of small image patches. Over a set of six canonical images (Barbara, Lena, etc.), FoE attained a PSNR of 30.24dB for Gaussian noise with $\sigma = 20$. A Gaussian scale mixture also uses a parametric approach, and captures the joint statistics of neighboring Gabor filter coefficients. Over the same set of six images, mean PSNR was 30.78 for $\sigma = 20$. Another method that exploits patterns found within the noisy input image is Non-local Means (NLM). However, NLM uses a wholly non-parametric approach, by identifying similar patches within the noisy input image and averaging these together, weighed according to similarity and proximity. NLM achieves a PSNR of 30.37dB for $\sigma = 20$ on the same set of images. BM3D is an algorithm with a similar strategy, but uses more sophisticated methods to combine similar image patches.

This brief list of algorithms includes some that are parametric and other non-parametric, some that focus on matching natural scene statistics and others that focus on utilizing patterns from within the noisy input image, and some that use generatively trained probabilistic models, some discriminatively trained probabilistic models, and others that do not use probabilistic models at all. Arrays of additional

differences are evident between the implementation details of each approach. In spite of these significant differences in strategy, performance is reasonably similar between these varied algorithms. Continual improvements to denoising algorithms regularly change the dominant approach, and no category of denoising strategies has produced a clear enough victor to discourage further research in any other category.

On the surface, this observation may suggest that image denoising algorithms are converging to some upper-bound on denoising performance. However, it is worth noting that each method is regarded as having different advantages and disadvantages. Strategies that perform best for low noise levels may not perform as well for high noise levels. Input images that contain many regular textures or patterns often benefit from NLM, while images with less internal regularity may benefit from algorithms trained from large suites of natural images. Additionally, one important quality of denoising methods is the ability to preserve sharp edges while removing noise. Methods that fail in this regard produce output that appear over-blurred. NLM often perform well at maintaining sharp edges, as demonstrated by their residual images (the denoised image minus the true image). These residual images show that methods like NLM perform similarly near edges as they do near smooth regions. Other methods such as FoE show higher residual error near edges. Since FoE achieves a similar overall PSNR, it suggests that performance within smooth regions is higher for FoE.

When multiple regression algorithms produce similar performance using significantly different approaches, and with distinct advantages and disadvantages, those algorithms are highly suitable for combination using ensemble learning methods. Ensemble learning is a method of combining multiple (possibly weak) predictors to produce one unified predictor of greater accuracy. While ensemble learning is a common and successful technique in machine learning, it has not been applied to image denoising.

Ensemble learning methods benefit when the constituent algorithms are significantly different from one another. In this thesis, we apply Bayesian ensemble learning methods to combine two of the most distinct denoising methods: Fields of Experts and NL-means.

We use 40 natural images from the Berkeley Segmentation Benchmark for training [8]. Another set of 80 natural images from Berkeley database are used of testing along with 6 canonical images. For each level of input noise, the ensemble method achieved statistically significant improvement over both FoE and NLM. The application of FoE or NLM has been considered mostly with natural images. Because the noise on the medical images is different from the natural images (e.g. MRI has the Rician noise distribution), image denoising algorithm based on the Gaussian white noise should be reconsidered for medical image denoising. Only a few studies were done to medical image denoising [9] [10] [11]. NLM was adapted to denoised MR images but they considered not only the Gaussian white noise but also the Rician noise [12]. There are two kinds of images in the Computed Tomography (CT), low dose and high dose images. High dose images could provide better resolution than low dose images but low dose images are required in recent years due to patient safety [13]. Low dose images have more noise than high dose images so a denoising algorithm of CT images should be considered. The algorithms for image denoising should contain not only efficient peak signal-to-noise ratio (PSNR) or Structural Similarity Index (SSIM) [14] but also no limitations from natural images to medical images. Therefore, we also tested the ensemble learning on MR images from BrainWeb [15]. We used 30 MR images for training and tested on another 20 MR images. To make noisy MR images, we used both Gaussian and Rician noise distribution. Because an assumption of FoE is based on natural images which use Gaussian distribution, the ensemble learning achieved outstanding improvement over both FoE and NLM with Gaussian noise. The ensemble

learning results did not perform well on Rician noise, but the results still showed that the ensemble learning outperformed NLM and FoE, or similar with NLM and FoE.

1.3 Thesis Statement

The goal of this thesis is to introduce an alternative approach focusing on the implementation of high performance algorithm based on the two models, Non-local means and Fields of Experts, not only for natural image denoising but also for medical image denoising. One benefit of exploring ensemble learning for denoising is that it is more than just a single denoising algorithm – it is methodology that can continue to be helpful even after Non-local Means and Fields of Experts become obsolete, since other denoising methods can be used instead of Non-local Means. And any probabilistic method can be used in place of Fields of Experts. Combining two methods, Non-local Means and Fields of Experts, by using the ensemble learning would have both advantages from Non-local Means and Fields of Experts, and improve the disadvantages from each other's.

1.4 Thesis Organization

Chapter 1: Introduction - A brief introduction to the medical images and the motivation of current challenge to the medical image denoising

Chapter 2: Background - A description of image denoising, different kinds of noise such as Gaussian and Rician noise, Non-local means, Fields of Experts, and how to calculate the PSNR with two different images

Chapter 3: Application of the Ensemble Learning - A description of Bayesian ensemble learning model with Non-local means and Fields of Experts

Chapter 4: Experimentation and Results - A description of the experimentations and results from our proposed denoising algorithm

Chapter 5: Discussion and Conclusion - A summarization with a discussion of our proposed denoising algorithm and the future work.

2 Background

2.1 Image denoising

The goal of image denoising is to reconstruct the original image from the noisy image,

$$\mathrm{y}(i) = \mathrm{x}(i) + n(i) \tag{2.1}$$

where $\mathrm{y}(i)$ is the observed image, $\mathrm{x}(i)$ is the original image and $n(i)$ is the noise value at pixel i. Adding a Gaussian white noise is the simple way to make a model of natural noisy images, and adding a Rician noise distribution is added to make a model of medical noisy images. The ideal denoising algorithm is to remove the noise, $n(i)$, and recover the original image, $\mathrm{x}(i)$.

Previous methods such as Gaussian [16] or Wiener filtering [2] attempt to separate the image into the two parts which are the smooth and oscillatory part by removing the high frequency from the low frequency. This would result in a loss of fine edges in the denoised image. Low frequency noise will remain in the image even after denoising. Therefore, new algorithms have been introduced recently such as Non-local means [4], Fields of Experts [5], or BM3D [7].

2.2 Gaussian and Rician Noise

2.2.1 Gaussian Noise

Gaussian white noise is widely used in natural images for image denoising. In (2.1), $n(i)$ is the Gaussian white noise values with known variance σ^2 and zero mean. The Gaussian white noise models are made by adding random values to the original images. Modeling noisy model for the medical images should be reconsidered because the noise in MR images has a Rician distribution [17].

2.2.2 Rician Noise

The signal of MR images is detected through a quadrature detector that has the real and the imaginary parts. The magnitude of MR image can be computed from the square root of the sum of the squares of the real and imaginary Gaussian distributions. The noise in the MR image follows a Rician distribution [17]. The Rician distributed noisy MR image will be defined as follows,

$$I_n = \sqrt{(I_o + \sigma)^2 + \sigma^2} \tag{2.2}$$

where $I_n$is the noisy MR image, $I_o$is the ground truth image, σ is the Gaussian noise standard deviation [18]. The Rician noise damages a quality of the MR images so MRI denoising filter always has been an issue in medical imaging society [9] [10]. The following figure is a sample noisy MR image with the Gaussian and Rician distribution and sigma value is 10.

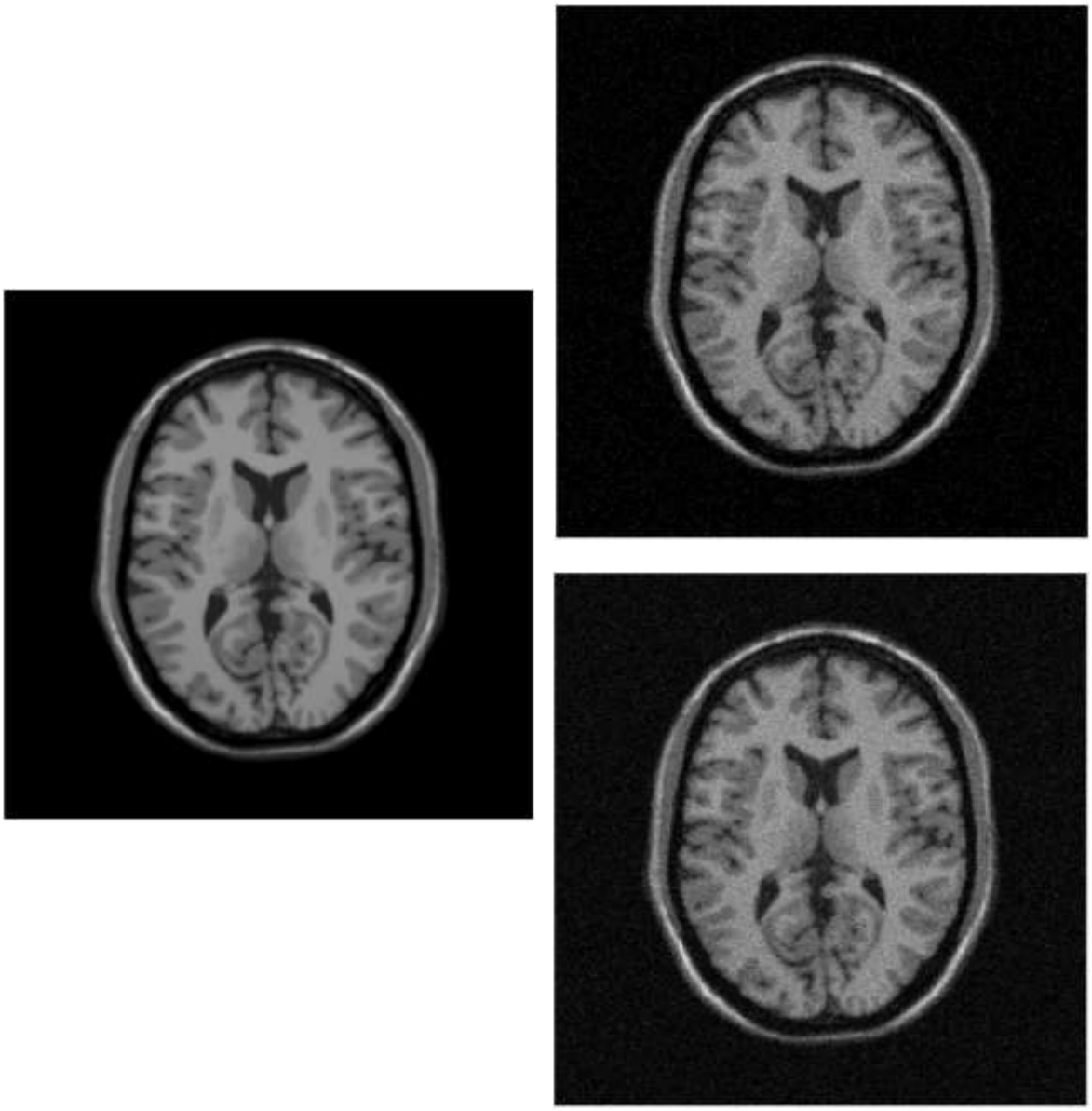


Figure 2.1. Sample noisy MR images with σ = 10
(Left: Original image, Top Right: Gaussian noisy image, Bottom Right: Rician noisy image)

2.3 Non-local means

Non-local means (NLM) image denoising algorithm was suggested by Antoni Buades, Bartomeu Coll, and Jean-Michael Morel. NLM presumes that the image contains an extensive amount of self-similarity [4]. Efros and Leung originally developed the concept of self-similarity for texture synthesis [19].

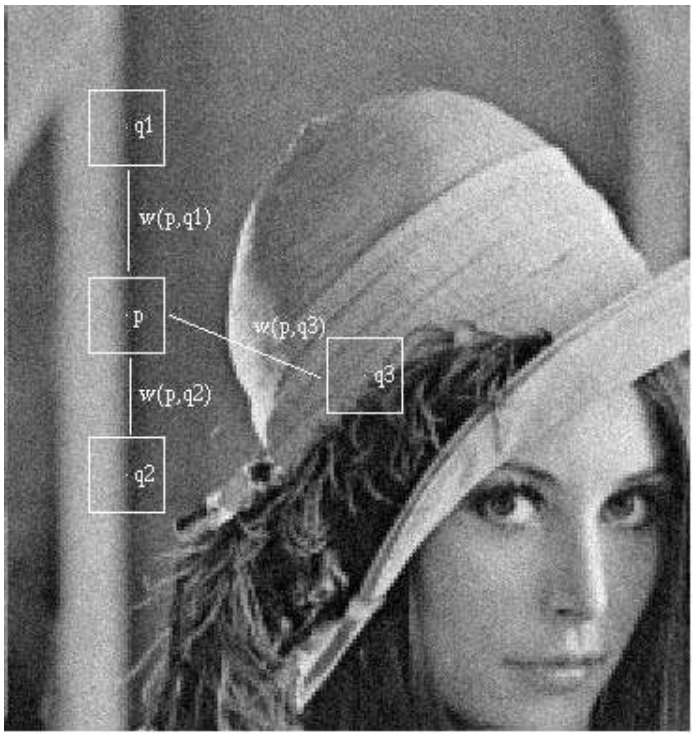


Figure 2.2. Scheme of NLM strategy

Figure 2.2 shows the scheme of NLM strategy. The figure shows four different pixels, p, q1, q2, and q3. Similar neighborhoods to p's neighborhood could be found in most pixels in the same column of p. Similar pixel neighborhoods give a large weight, w(p, q1) and w(p, q2), while much different neighborhoods give a small weight w(p, q3).

To compute each pixel I of the NLM denoising image, the following formula was used [4]:

$$\mathrm{NL}[v](i) = \sum_{j \in I} w(i,j) v(j) \tag{2.3}$$

where v is the noisy image $v = \{v(i) | i \in I\}$, and weights $w(i,j)$, which rely on the similarity between pixel i and j, meet the following conditions $0 \leq w(i,j) \leq 1$ and $\sum_{j \in I} w(i,j) = 1$. A neighborhood should be defined to compute the similarity. The weighted Euclidean distance is used to measure the similarity and the following formula is implemented to calculate the Euclidean distance of noisy neighborhoods:

$$\mathrm{E}\left\| v(N_i) - v\left(N_j\right) \right\|_{2,a}^2 = \left\| u(N_i) - u\left(N_j\right) \right\|_{2,a}^2 + 2\sigma^2 \tag{2.4}$$

where $v(N_i)$ and $v\left(N_j\right)$ are the gray scale vectors, N_i and N_j are the square neighborhood of fixed size and centered around a pixel i and j, respectively, and a is the Gaussian kernel's standard deviation. The weights $w(i,j)$ can be calculated with the following formula:

$$\mathrm{w}(i,j) = \frac{1}{Z(i)} e^{-\frac{\left\| v(N_i) - v\left(N_j\right) \right\|_{2,a}^2}{h^2}} \tag{2.5}$$

where $Z(i)$ is the normalizing constant

$$Z(i) = \sum_j e^{-\frac{\left\| v(N_i) - v\left(N_j\right) \right\|_{2,a}^2}{h^2}} \tag{2.6}$$

and the parameter h satisfies as a filtering degree.

The original NLM applied to 2D natural images and also continued to 3D images especially MR images [11][20]. For a MR image case, the 3D neighborhoods of the two voxels are compared to calculate the similarity of two voxels.

Figure 2.3. Left: Original image
Top Right: Noisy image with σ = 20, Bottom Right: Denoised image by using NLM

2.4 Fields of Experts

Fields of Experts was proposed by Stefan Roth and Michael J. Black [21]. The goal of the Fields of Experts is to develop a framework for learning rich, generic prior models of natural images. To learn potential functions through extended neighboring pixels, Markov Random Field model was used in the Fields of Experts. The key in the Fields of Experts is to extend Markov Random Field by modeling the local field potentials with learned filters [21]. To do this, Products of Experts were used [5]. In comparison with prior Markov Random Field approaches, all parameters in the Fields of Experts model are learned from a set of training data [21]. Those models prior probability of images can be calculated with the following formula:

$$\mathrm{P}(\vec{I}) \propto \prod_C \prod_{i=1}^{K} \left(1 + \frac{1}{2}\left(\overrightarrow{I_c} \cdot \overrightarrow{J_c}\right)^2\right)^{-\alpha_i} \tag{2.7}$$

where I_c is 5x5 image patch and filter J_c represents especially unlikely image patches obtained by training the Fields of Experts model on an general image database. About 20,000 image patches are selected randomly from the Berkeley Segmentation database and the image patches are used for the training data [8]. Figure 2.4 shows a selection of the 24 filters learned from the training Fields of Experts model on 5x5 pixels.

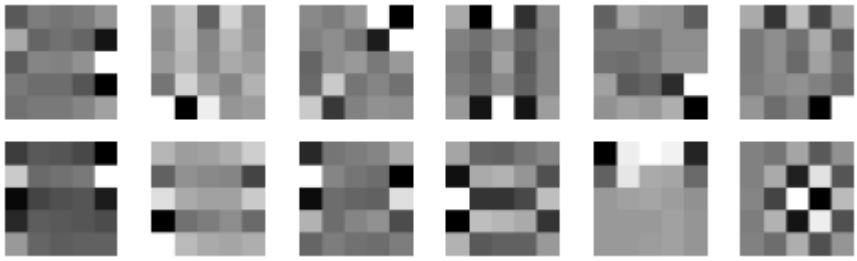


Figure 2.4. Selection of the 5x5 filters learned from the training data

Inference: For the denoising problem, the goal is to infer the most likely correction for the image given the prior and the noisy image. Given a noisy image N, we can find the denoised image D that maximizes the prior probability:

$$\mathrm{p}(D|N) \propto p(N|D)p(D) \tag{2.8}$$

We can write the $p(N|D)$ as:

$$\mathrm{p}(N|D) \propto \prod_j \exp\left(-\frac{1}{2\sigma^2}\left(D_j - N_j\right)^2\right) \tag{2.9}$$

where σ is known standard deviation and D_j and N_j are the denoised and noisy image at pixel j, respectively. $p(N|D)$ is the Gaussian distribution and $p(D)$ is the Fields of Experts model which is in formula (2.7). In this study, we use the Fields of Experts algorithm MATLAB code provided by the authors and use the similar parameters to get the same results of the paper.

Figure 2.5. Left: Original image
Top Right: Noisy image with σ = 20, Bottom Right: Denoised image by using FoE

2.5 Peak Signal to Noise Ratio

The peak signal-to-noise ratio is widely used for the ratio between two different images such as original and noisy image. The PSNR is usually measured in the logarithmic decibel scale and expressed in the Mean Square Error, MSE. The MSE which has x·y size of two different images, f and g (one of the images is noisy image) is defined as:

$$\text{MSE} = \frac{1}{xy}\sum_{i=0}^{x-1}\sum_{j=0}^{y-1}\big(f(x,y) - g(x,y)\big)^2 \tag{2.10}$$

The PSNR is expressed as:

$$\text{PSNR} = 10 \cdot \log_{10}\left(\frac{max_I^2}{MSE}\right) = 20 \cdot \log_{10}\left(\frac{max_I}{MSE}\right) \tag{2.11}$$

where max_I is the peak value of the image and this value is equal to 255 when the pixels have 8 bits sampling rate. When the two different images, f and g, do not have differences, the MSE will be almost 0 so the PSNR will have an infinite value.

Figure 2.6. Example PSNR values (Left: Original Image,
Center: Noisy image with σ = 20 (PSNR: 22.11dB),
Right: Denoised image by using the Fields of Experts (PSNR: 29.63dB))

3 Application of the Ensemble Learning

3.1 The Ensemble Learning

As discussed in the introduction, in spite of relatively comparable performance, image denoising techniques like FoE and NLM use very different methodologies and underlying philosophies for image denoising. This would seem to make these two algorithms strong candidates for ensemble learning methods which may be able to produce a single denoising algorithm that retains the advantages of both FoE and NLM. However, this problem differs from traditional ensemble learning problems in several important ways, which result in both advantages and disadvantages in comparison with standard ensemble learning scenarios.

One significant difference is that ensemble learning is typically used to combine multiple models of the data or the predicted output, where each constituent algorithm was trained on labeled data to optimize the parameters of the algorithm. In contrast, many image denoising methods utilize no training at all. Methods like NLM are derived primarily from a theoretical analysis of the image denoising problem, rather than machine learning techniques. NLM does have some parameters, such as the size of the search window, but these parameters are not expected to affect the behavior of the algorithm significantly, and so training is not emphasized.

The lack of training in the constituent algorithms makes image denoising ineligible for some of the most powerful aspects of ensemble learning. Many ensemble learning methods, such as boosting [22] or

bootstrap aggregation [23], achieve higher performance by training each constituent algorithm on different subsets of available training data. In that way, one algorithm becomes an "expert" on one type of input, while another algorithm specializes in another type of input. Because NLM does not rely on training or make use of labeled data, such techniques are not possible here.

Another important difference is that the bulk of ensemble learning methods are designed for classification problems, where the predicted output is either binary, or at least discrete. For classification problems, the outputs of each constituent algorithm are typically combined using a voting scheme: whichever class receives the most votes (possibly weighted by the performance of the constituent algorithm) is the output of the ensemble learning algorithm. Ensemble learning for continuous-valued output (known as regression) is not entirely uncommon [24]. In these cases, outputs of the constituent algorithms are usually combined by averaging. Unfortunately, averaging multiple denoised images may lead to undesirable artifacts. Each denoising algorithm must hypothesize the most probable image structure that was obscured by noise in the input image. When these hypotheses disagree, averaging them together can result in two faint but duplicate differing image structures, rather than a single consistent denoised image. For example, consider the case where noise has obscured the precise location of an edge between two regions. If NLM hypothesizes an edge in one location, and FoE hypothesizes an edge in a different location, an averaging procedure will produce two faint edges, which is generally an unlikely result. We would prefer an ensemble technique that could select the single edge location that appears most likely. Ideally, an ensemble method should be capable of selecting a single coherent denoising hypothesis rather than combining multiple outputs naively. This advantage of traditional ensemble methods may provide one explanation as to why ensemble methods have not been applied to image denoising in the past.

If a large number of constituent algorithms are available, more sophisticated schemes for combining algorithm output might be possible, where outlier responses were given less weight. Such techniques might avoid the pitfalls of averaging methods listed above. Unfortunately, only a handful of competitive image denoising algorithms exist.

Finally, a third major difference is that some constituent denoising algorithms (in particular, FoE) provide not only a single hypothesis denoised image, but also provide a probability distribution over the space of all possible denoised images. Recall that FoE outputs not only a denoised image D_{FoE}, but also provides a probability distribution over denoised images $P_{FoE}(D|N)$, where N is the input noisy image. This distribution allows us to quantify the uncertainty in the FoE solution, or potentially, to measure the likelihood of the outputs of other denoising algorithms within the FoE model. As we describe below, the availability of a probabilistic model of the output space is a great advantage for applying ensemble method to the image denoising problem, because it provides a possible solution to the disadvantages of averaging described above.

If all constituent algorithms provide a distribution over the space of hypothesis, then Bayesian ensemble learning methods can be used to combine each distribution into a single distribution. In particular, the choice of which model is superior for a particular input can be treated as a latent variable. Specifically, suppose we wish to predict output y given input x, and we have a database of labeled training examples Z. Also, suppose we have M different probabilistic predictors which each provide a distribution $P_m(y|x)$ over possible outputs. Then we can write the probability of possible outputs y as:

$$P(y|x,Z) = \sum_{m=1}^{M} P(y,m|x,Z) \quad (3.1)$$

$$= \sum_{m=1}^{M} P(y|m,x,Z)P(m|x,Z) \quad (3.2)$$

$$= \sum_{m=1}^{M} P_m(y|x,Z)P(m|Z) \quad (3.3)$$

Thus, the Bayesian ensemble distribution is simply the weighted sum of each constituent distribution, weighted by performance over the training data Z. The downside of Bayesian ensemble methods is that the ensemble distribution $\sum_{m=1}^{M} P_m(y|x,Z)P(m|Z)$ is often computationally demanding to optimize. However, the advantage is that outputs y that score highly in the ensemble distribution must be considered probable according to all of the constituent algorithms, especially those that were most successful on the training data. If Bayesian ensemble methods were applied to the image denoising problem, this would be eliminate the pitfall of averaging two denoising outputs together. The mean of two plausible solutions may not itself be plausible, but the optimum of a Bayesian ensemble distribution largely satisfied both models simultaneously.

Unfortunately, Bayesian ensemble learning methods cannot be applied in the denoising problem, because not all of the constituent denoising algorithms provide a probability distribution over denoised images. Instead, we must combine a mixture of probabilistic and non-probabilistic models. To our knowledge, this circumstance has not been studied explicitly in past ensemble learning methods. Our goal is to retain the advantage of purely Bayesian ensemble approaches: the ensemble method should produce an output that internally consistent, considered highly plausible by the FoE probability distribution, while simultaneously resembles the NLM output. Additionally, we need to find a method that is efficient.

Bayesian ensembles are often computationally intensive to optimize, in part because, being summations, they do not factorize.

Our approach is to treat the NLM output as a known, given quantity. Thus, we want to model $P(I|N, D_{NL})$, where I is the hypothetical original image, N is the noisy image, and D_{NL} is the output of NLM. By applying Bayes rule, we can write

$$P(I|N, D_{NL}) = P(N|I, D_{NL})P(I|D_{NL})/P(N|D_{NL}) \tag{3.4}$$

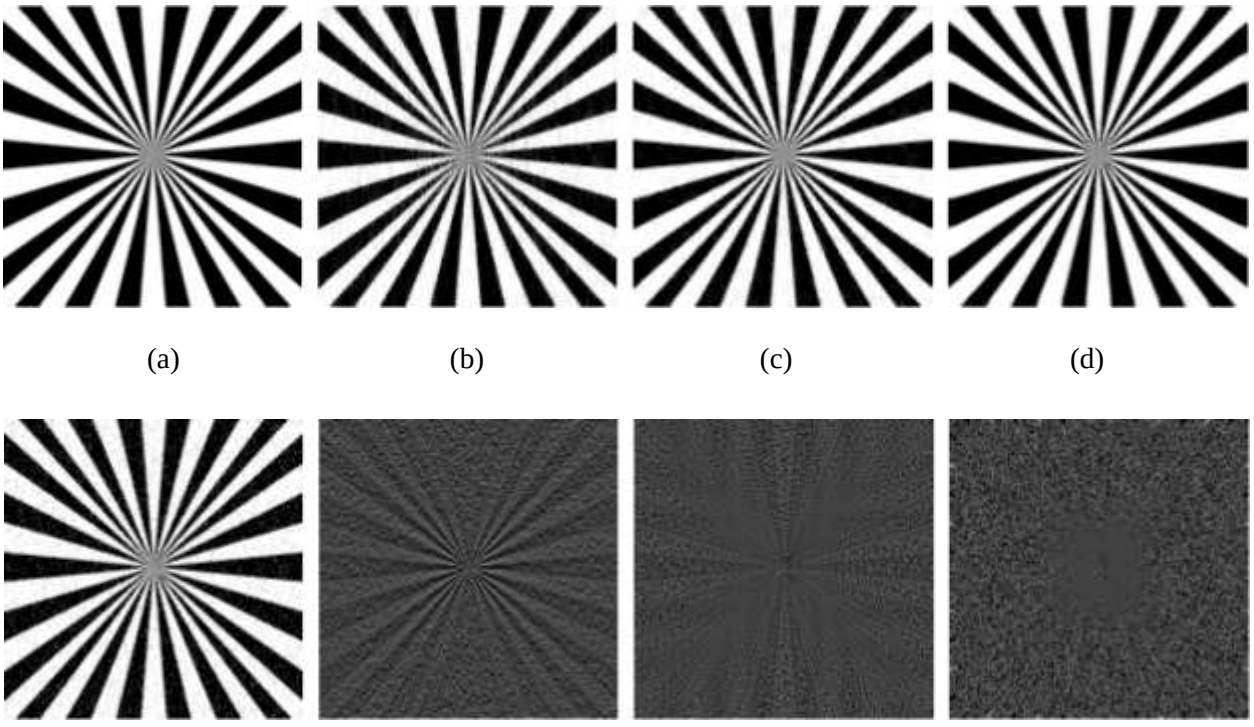
$$= \frac{P(N|I)P(D_{NL}|I)P(I)}{P(N|D_{NL})P(D_{NL})} \tag{3.5}$$

$$\propto P(N|I)P(D_{NL}|I)P(I) \tag{3.6}$$

where terms that do not depend on I can be ignored as constants. The term $P(N|I)$ is simply the noise model, which is a Gaussian of mean I and variance given by the strength of the image noise. The prior over noiseless images, $P(I)$, can be taken from the FoE model, given by equation (2.7).

To complete the ensemble model, we must choose a model for $P(D_{NL}|I)$. One known strength of the NLM method is that the image residual, defined as $I - D_{NL}$, shows little image structure [4]. In other words, edges and features that are visible in the noisy image are faint or not visible in the residual. In comparison, other leading denoising images often produce residuals that retain structure from the noisy image. This advantage of NLM is believed to stem from the methodology used by NLM. Trained methods like FoE base their models for image structure such as edges entirely from databases of natural images. In contrast, NLM acquires statistics of image structure from the noisy input image itself. For example, FoE may misjudge the spatial scale, or sharpness, of edges if evidence of the edge is weak within the image, and other false spatial of edges solely by comparing against similar edges within the

noisy image, and the edges with less common scales are less likely to be biased. For these reasons, strong structure visible in the noisy input image is less likely to be visible in the residuals of NLM outputs.



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 3.1. The image residuals (reproduced from [25]): σ = 20. (a) original image, (b) PDE-based [26] (c) Wavelet-based [27], (d) NLM, (e) noisy image, (f)-(g) the residual of each image (b)-(d) respectively

One consequence of this observation is that the residual of NLM can be approximated as white noise. In particular, the residual of NLM is known to closely resemble the statistical structure of the additive noise. This allows us to define $P(D_{NL}|I)$ accordingly. In the case of input images with additive Gaussian noise, $P(D_{NL}|I)$ is Gaussian centered at I, with some $\sigma_{NL} < \sigma$:

$$p(D_{NL}|I) \propto \exp\left(-\frac{\sum_{x,y}\left(I(x,y)-D_{NL}(x,y)\right)^2}{2\sigma_{NL}^2}\right) \tag{3.7}$$

Likewise, when the input image has additive Rician noise, $p(D_{NL}|I)$ should be Rician.

This completes the definition of our ensemble model. Note that this model completes our objective: images that optimize the ensemble distribution are simultaneously

- plausible according to the Fields of Experts model
- similar to the Non-local Means output
- close to the noisy input image

We also must ensure that our model can be optimized efficiently. Here, we observe that our model can be simplified into a form that is very similar to the FoE probabilistic model [5]. We can start with the formula (3.6).

$$p(N|I)p(D_{NL}|I) \propto \exp\left(\frac{(N-I)^2}{2\sigma_N^2}+\frac{(D_{NL}-I)^2}{2\sigma_{NL}^2}\right) \propto \exp\left(\frac{\left(N_{pseudo}-I\right)^2}{2\sigma_{pseudo}^2}\right) \tag{3.8}$$

In formula (3.8), the left side shows a component of the ensemble learning model, and the right side shows the same model re-arranged to have the same structure as FoE [5]. This new form introduces new

variables such as N_{pseudo} and σ_{pseudo}. We can get these parameters σ_{pseudo} and N_{pseudo} by comparing the left and right side. In the left side, the parameters such as N, I, and D_{NL} are all functions.

$$\frac{(N^2-2\cdot N\cdot I+I^2)}{2\sigma_N^2}+\frac{(D_{NL}^2-2\cdot D_{NL}\cdot I+I^2)}{2\sigma_{NL}^2}=\frac{(N_{pseudo}^2-2\cdot N_{pseudo}\cdot I+I^2)}{2\sigma_{pseudo}^2} \tag{3.9}$$

We can combine I^2 and I part of left and right side to get the parameters σ_{pseudo} and N_{pseudo}. The coefficient for the term I^2 must be same for the right and left side of the formula (3.9). The formula (3.10) shows the I^2 part of left and right side. The formula (3.11) shows the I part.:

$$I^2\cdot\left(\frac{1}{2\sigma_N^2}+\frac{1}{2\sigma_{NL}^2}\right)=\ I^2\cdot\left(\frac{1}{2\sigma_{pseudo}^2}\right) \tag{3.10}$$

$$I\cdot\left(-2N\cdot\frac{1}{2\sigma_N^2}-2D_{NL}\cdot\frac{1}{2\sigma_{NL}^2}\right)=I\cdot\left(-2N_{pseudo}\cdot\frac{1}{2\sigma_{pseudo}^2}\right) \tag{3.11}$$

From the formula (3.10), we can get the parameter σ_{pseudo}. We can get the parameter, N_{pseudo}, from the formula (3.11). Therefore, the parameters σ_{pseudo} and N_{pseudo} are

$$\begin{aligned}{\sigma_{pseudo}}^2 &= 1/\{(1/\sigma_N^2)+(1/\sigma_{NL}^2)\}\\ N_{pseudo}\ &= \mathrm{N}\cdot\alpha+D_{NL}\cdot\beta\end{aligned} \tag{3.12}$$

where

$$\begin{aligned}\alpha &= (1/\sigma_N^2)/\{(1/\sigma_N^2)+(1/\sigma_{NL}^2)\}\\ \beta &= (1/\sigma_{NL}^2)/\{(1/\sigma_N^2)+(1/\sigma_{NL}^2)\}\end{aligned} \tag{3.13}$$

Because the ensemble learning is usually computationally intensive, choosing the method of combining two algorithms, such an NLM and FoE in this case, allows us to do with extra computational effort and extra coding to build the ensemble learning.

In theory, σ_{NL} could be estimated empirically by finding the standard deviation of the residuals of NLM outputs on natural images for each noise level σ. In practice, we found that superior performance was attained by learning σ_{NL} from a set of training images.

3.2 The process of the Ensemble Learning

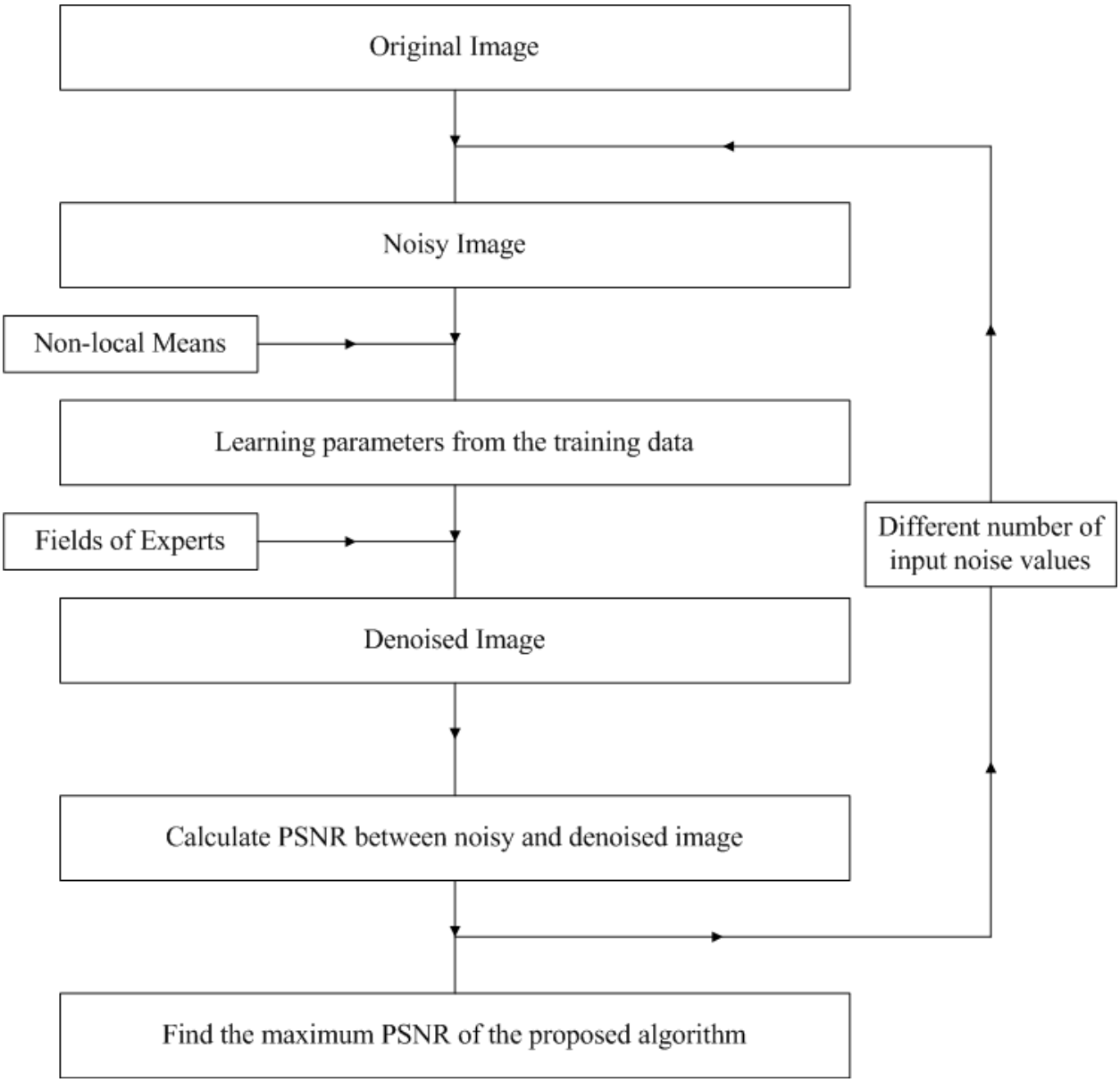


Figure 3.2. Block diagram of the ensemble learning

Figure 3.2 shows the block diagram of the ensemble learning. First, a noisy image could be made with the original image by using the Gaussian or Rician distribution. The same noisy images were used for NLM and FoE in this model.

The first denoising progress was performed by using the Non-local means. The parameters σ_{pseudo} and N_{pseudo} were calculated with the denoised image, D_{NL}, which was done by the NLM, σ_N , the original input Gaussian or Rician sigma value and σ_{NL} ,another sigma value from the NLM denoised image. The second denoising progress with FoE algorithm was adapted to the NLM denoised image, D_{NL}, with the learned parameters, σ_{pseudo} and N_{pseudo}. The PSNR could be computed by using the original images and denoised images after performing the FoE algorithm. Same process was executed with different number of the input Gaussian or Rician noise values, σ_N. The maximum PSNR of the ensemble learning could be accomplished by comparing all the PSNR values.

For natural image denoising, 40 images from Berkeley Segmentation Benchmark were used for training σ_{pseudo}. The selection of σ_{pseudo} is very important in our prosed ensemble learning. Based on the best selection of σ_{pseudo}, we used 80 natural images from Berkeley database and 6 canonical images for testing. For medical image denoising, we separate to two parts, noisy image with Gaussian and Rician. 20 MR images from BrainWeb were used for training σ_{pseudo} and we tested on other 30 MR images. This process had done to both Gaussian and Rician noise.

4 Experimentation and Results

4.1 Experimentation and Results of the natural images

4.1.1 Training dataset

First, we performed the ensemble learning on the 40 natural images randomly selected from the Berkeley Segmentation Benchmark to get the proper σ_{pseudo} for the best performance [8]. Different numbers of input Gaussian noise, σ, were added to the each original image. We used the provided MATLAB code of NLM and FoE algorithm from the author's website and built the ensemble learning code with several lines of MATLAB codes. All the codes were run through the Bioinformatics Cluster (Linux cluster with 64 dual processor 3.2Ghz Xeon processors and 64 dual core 2.8Ghz Xeon processors for a total of 384 processors 37 TB of on-line storage Reconfigurable floating-point gate arrays) at the Information and Telecommunication Technology Center at the University of Kansas [28]. 80 other image sets from the Berkeley Segmentation Benchmark and the six natural images (barbra, boat, fingerprint, house, lena, and peppers) were evaluated with single NLM, FoE and the ensemble learning to verify the performance of the ensemble learning. The ensemble learning denoising performance was evaluated by the Peak Signal-to-noise ratio (PSNR).

The noisy image was obtained from the original image by adding with different numbers of input noise value, σ = 10, 15, 20, 30, 40, 50, 75 and 100. The NLM algorithm was used to get the $D_{NL}$and several

numbers of sigma values from the NLM denoised image were used, σ_{NL} = 1, 2, 3, 5, 10, 20, 30, 40, 50, 100, 250, and 500. We used all these sigma values, NLM denoised images and FoE algorithm to get the ensemble learning denoised images. The 5x5 filter of Fields of Experts was used to obtain the denoised images. We used 5,000 iteration numbers to implement for FoE [5].

All of these processes were applied to different noisy images with different numbers of input noise values. PSNR was calculated with the original images and denoised images which were acquired from the ensemble learning. And then the average of their PSNR values with different input sigma values and other sigma values, σ_{NL}, from the NLM denoised images were calculated.

Figure 4.1. Subset of the training images for the proper selection of σ_{NL}

Figure 4.1 shows a subset of the 40 training images for the proper selection of σ_{NL}. The training images contain images of architecture, landscape, people, flowers, etc. The proper selection of σ_{NL} for the best performance could be learned from these image dataset. The parameters, σ_{pseudo}, could be learned from the selected σ_{NL} and the given input sigma value, σ , by using the formula (3.12).

Table 4.1: The average PSNR (dB) results from the 40 natural images for training σ_{pseudo}

sigma	10	15	20	30	40	50	75	100
NLM	31.85	30.20	28.88	26.95	25.66	24.69	23.04	21.82
0001	31.85	30.21	28.91	26.99	25.69	24.72	23.06	21.84
0002	31.97	30.26	28.95	27.06	25.85	24.92	23.22	21.95
0003	32.09	30.28	28.95	27.07	25.85	24.92	23.22	21.95
0005	32.40	30.34	28.94	27.04	25.84	24.96	23.48	22.20
0010	33.04	30.53	28.93	26.90	25.70	24.85	23.54	22.61
0020	33.47	30.96	29.26	26.99	25.67	24.79	23.50	22.65
0030	33.50	31.14	29.51	27.16	25.74	24.79	23.49	22.68
0040	33.50	31.18	29.60	27.29	25.74	24.75	23.46	22.71
0050	33.49	31.17	29.61	27.34	25.93	24.72	23.42	22.71
0100	33.48	31.13	29.57	27.38	25.73	24.64	23.23	22.50
0250	33.47	31.11	29.54	27.36	25.73	24.61	23.06	22.25
0500	33.47	31.11	29.53	27.35	25.74	24.61	23.03	21.15
FoE	33.36	31.07	29.53	27.35	25.74	24.61	23.02	19.37

Table 4.1 shows the average PSNR results of the NLM, FoE and the ensemble learning for training σ_{pseudo} with different number of input sigma and other σ_{NL} values from the NLM denoised images. From the table 4.1, the ensemble learning showed the better result comparing to the single NLM and FoE. When the input sigma is 20, the average PSNR of NLM is 28.88dB and FoE is 29.53dB. The ensemble learning is 29.61dB at that point with σ_{NL}=50. The average PSNR of ensemble learning with small σ_{NL} value such as 1 showed a similar result with the NLM algorithm because of N_{pseudo} from the formula (3.12) and (3.13). When the σ_{NL} is close to very small number, the β component in the N_{pseudo} is much bigger than the α component so the N_{pseudo} is most likely to the D_{NL}, which is the denoised image from the NLM algorithm. On the contrary to this, when the σ_{NL} has a large value such as 500, the N_{pseudo} is very similar to the N, which is the original noisy image, and the denoised result of this N_{pseudo} is almost same as the FoE algorithm.

We could find that the denoising performance of ensemble learning showed a better result when the average PSNR of NLM and FoE is similar. For example, when the input sigma is 50, the PSNR of the NLM is 24.69dB and the FoE is 24.61dB. In this case, the PSNR of the ensemble learning is 24.96dB, and most PSNR values of the ensemble learning are better than the NLM and FoE algorithm except when the σ_{NL} are 100, 250 and 500. Based on our result, the ensemble learning algorithm showed an improvement performance with the high input sigma value such as 50 and 75, which have a similar PSNR value between the NLM and FoE. The ensemble learning outperforms with the σ_{NL} = 30, 40, 50, 100, 50, 5, 10, 40 when the σ = 10, 15, 20, 30, 40, 50, 75, 100, respectively. σ_{pseudo} and N_{pseudo} could be calculated with the selected σ_{NL} and σ by using the formula (3.12) and (3.13). For example, when the σ is 20, the best selection of σ_{NL} is 50. Therefore, σ_{pseudo}, α, and β have 18.57, 0.86, and 0.14, respectively. N_{pseudo} could be obtained with the calculated α and β.

4.1.2 Results of the natural images

Results of the natural images were attained from two image datasets. The first result was done with the 6 canonical images (Barbara, Boat, etc). Table 4.2 shows the PSNR results for 6 natural images denoised by using the ensemble learning with different number of input noise sigma and the selected σ_{NL}. Bold and underlining numbers in the Table 4.2 indicate the maximum PSNR results among the three denoising algorithms. There are three results, NLM, FoE and Ensemble learning (from the left) for each image. The most of the results of the ensemble learning showed an improvement in image denoising comparing to the NLM and FoE. Especially, the ensemble learning showed a significant improvement when the input noise value increased. For example, when the input sigma is 75, the ensemble learning outperformed NLM and FoE in all cases. When the input sigma of the Barbara is 50, the PSNR result of the ensemble learning, NLM, and FoE are 24.78dB, 24.70dB, and 23.13dB, respectively. Figure 4.2 shows the results of denoising images by using the NLM, FoE and ensemble learning. Comparing to these results, the ensemble learning outperformed both NLM and FoE quantitatively. With selected σ_{NL}, the PSNR result of the ensemble learning was 27.62dB. The PSNR results of NLM and FoE were 27.14dB and 26.92dB, respectively.

The denoising results of NLM usually have an advantage on some edges comparing with the FoE. However, cyclic borders could be found on the denoising results of NLM. The denoising results of FoE sometimes have a disadvantage on some edges, especially when the input noisy sigma value is high. The ensemble learning is available to keep some edges, which is an advantage from NLM and a disadvantage from FoE, and remove the cyclic borders. Therefore, some disadvantages from NLM and FoE could be overwhelmed by the ensemble learning.

Table 4.2: The PSNR (dB) results for natural images denoised with the Ensemble learning (From the left: Non-local Means, Fields of Experts and Ensemble learning)

σ	Barbara			Boat			Fingerprint		
10	**33.72**	32.93	33.19	32.78	33.27	**33.27**	31.03	32.11	**32.17**
15	**31.80**	30.25	30.6	30.97	31.40	**31.42**	29.08	29.64	**29.78**
20	**30.19**	28.41	28.77	29.54	**30.05**	30.04	27.44	28.03	**28.16**
30	**27.72**	25.88	26.03	27.56	**27.96**	27.95	25.09	25.77	**25.79**
40	25.96	24.17	**26.04**	26.11	26.18	**26.27**	23.22	**23.58**	23.28
50	24.70	23.13	**24.78**	25.04	24.93	**25.28**	21.80	21.47	**21.88**
75	22.91	22.00	**22.92**	23.27	23.13	**23.73**	19.49	18.23	**19.54**
100	21.70	18.97	**21.86**	22.04	20.44	**22.76**	**18.20**	17.63	17.57

σ	House			Lena			Peppers		
10	**35.47**	35.22	35.27	**35.17**	35.11	35.16	33.38	34.16	**34.21**
15	**33.92**	33.62	33.74	33.36	33.32	**33.41**	31.74	32.05	**32.2**
20	**32.61**	32.34	32.50	31.97	32.03	**32.11**	30.49	30.57	**30.76**
30	30.05	30.34	**30.40**	29.84	29.88	**29.95**	28.11	28.10	**28.2**
40	28.23	**28.74**	28.57	28.26	28.13	**28.58**	26.64	26.47	**26.83**
50	26.71	**27.28**	27.17	27.14	26.92	**27.62**	25.16	24.95	**25.43**
75	24.34	24.68	**25.15**	25.03	24.94	**25.88**	22.91	22.55	**23.35**
100	22.85	19.75	**24.03**	23.46	21.05	**24.8**	21.5	18.84	**22.04**

(1) (2)

(3) (4)

Figure 4.2. Denoising results. (1) Image with Gaussian noise, σ = 50 (PSNR = 14.18dB), (2) Denoised image using the NLM (PSNR =27.14dB), (3) Denoised image using the FoE (PSNR = 26.92dB), (4) Denoised image using the ensemble learning (PSNR = 27.62dB) with σ_{NL} = 5

Figure 4.3. Close-up denoising results. From the left, NLM, FoE, and ensemble learning

Figure 4.3 shows the close-up denoising results. From the Figure 4.3, we could find the advantages and disadvantages of NLM and FoE. Some edges could be found in the result of NLM, but not in the result of FoE. Cyclic borders could be detected in the result of NLM. Our ensemble learning model could not only have some good edges but also have less cyclic borders. In other words, the ensemble learning could be overcome some disadvantages from NLM and FoE.

The second result was done with a subset of 80 images from Berkeley database [8]. Figure 4.4 shows the subset of 80 tested images. A dataset of 80 tested images contains images of people, landscape, architecture, etc. This dataset was selected randomly from Berkeley database. Figure 4.5 shows noisy images and results of ensemble learning with σ =30.

Figure 4.4. A subset of tested 80 images from Berkeley database

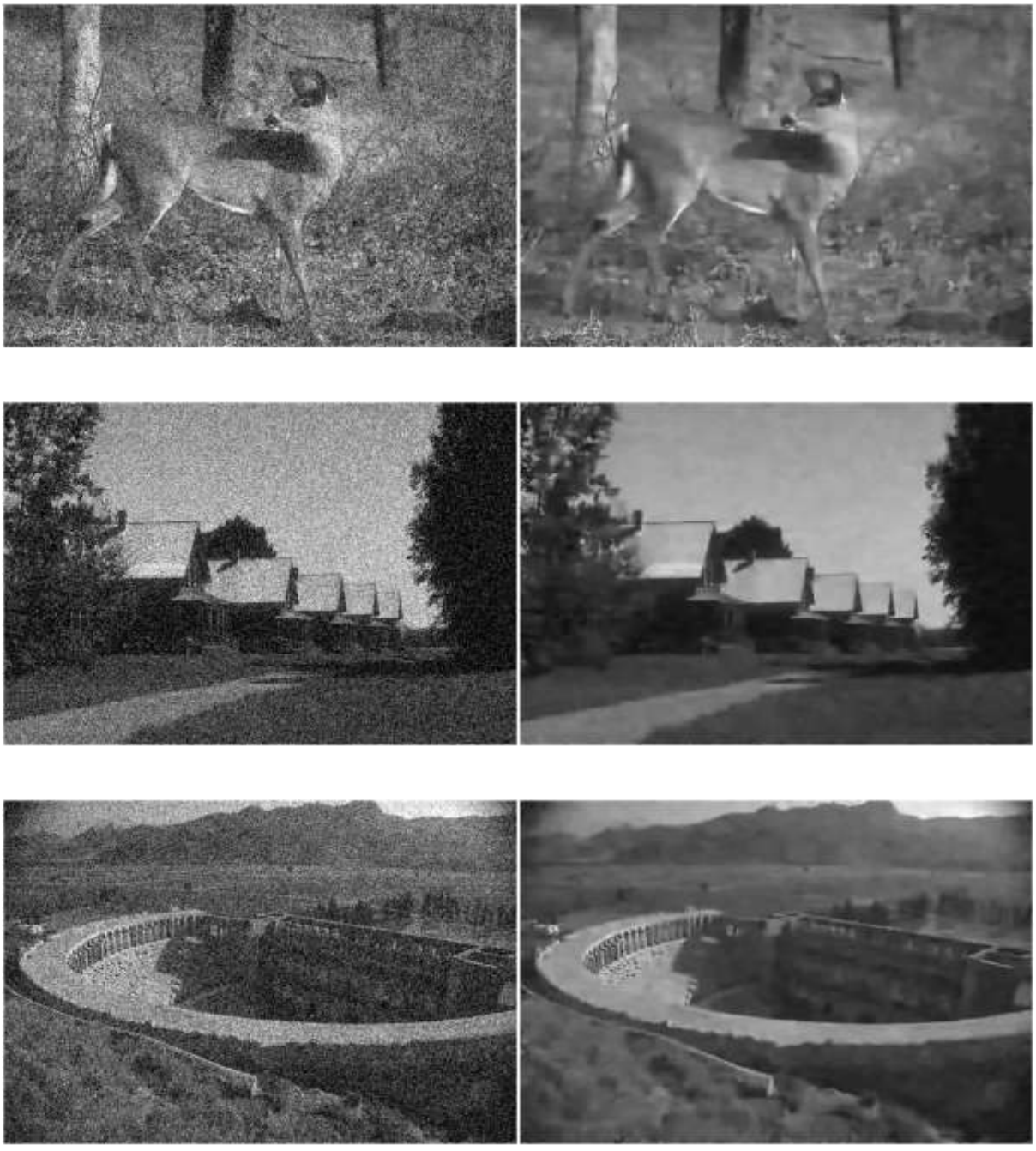


Figure 4.5. Ensemble learning results with σ = 30 (Left: Noisy image, Right: Denoised image)

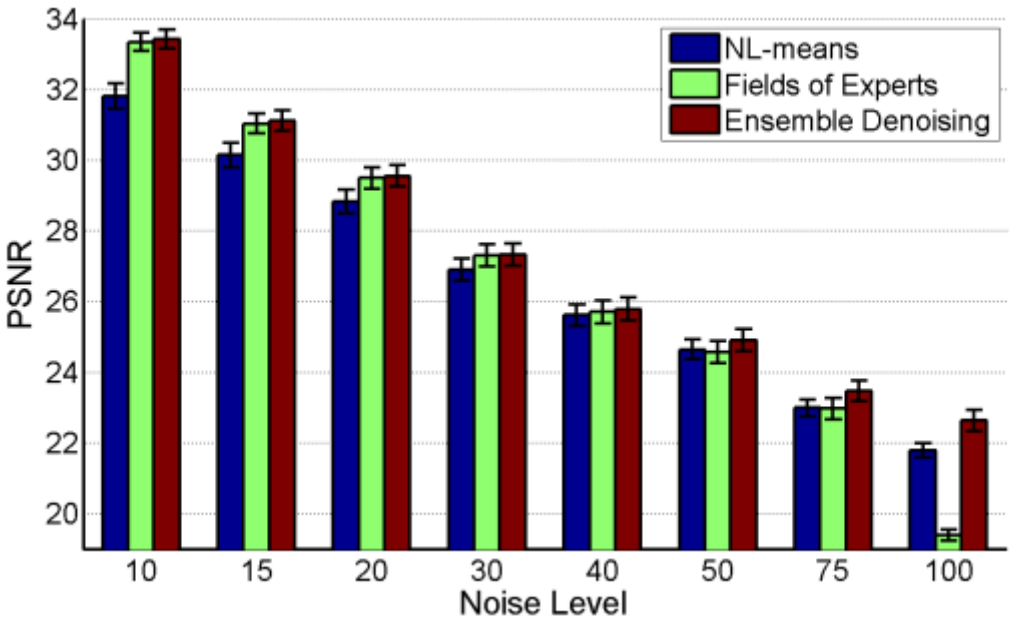


Figure 4.6. Average denoising PSNR results of 80 natural images.
(From left to right, Non-local means, Fields of Experts, and Ensemble learning)

The result of average PSNR values at different input noise level is displayed in Figure 4.6. The standard error is shown as the error bar. The ensemble learning outperformed the NLM and FoE in most cases. The ensemble learning showed a better improvement, especially at the high levels of the input noise. When the input noise is 100, the average PSNR results of ensemble learning, NLM, and FoE are 22.75dB, 21.83dB, and 19.47dB, respectively. All the σ_{NL} numbers which are needed to calculate σ_{pseudo} are selected from the results of the training dataset. Figure 4.7 shows an improvement of the ensemble learning comparing with the best performance of FoE and NLM. The improvement of the ensemble learning is increasing gradually while the input noise level goes high.

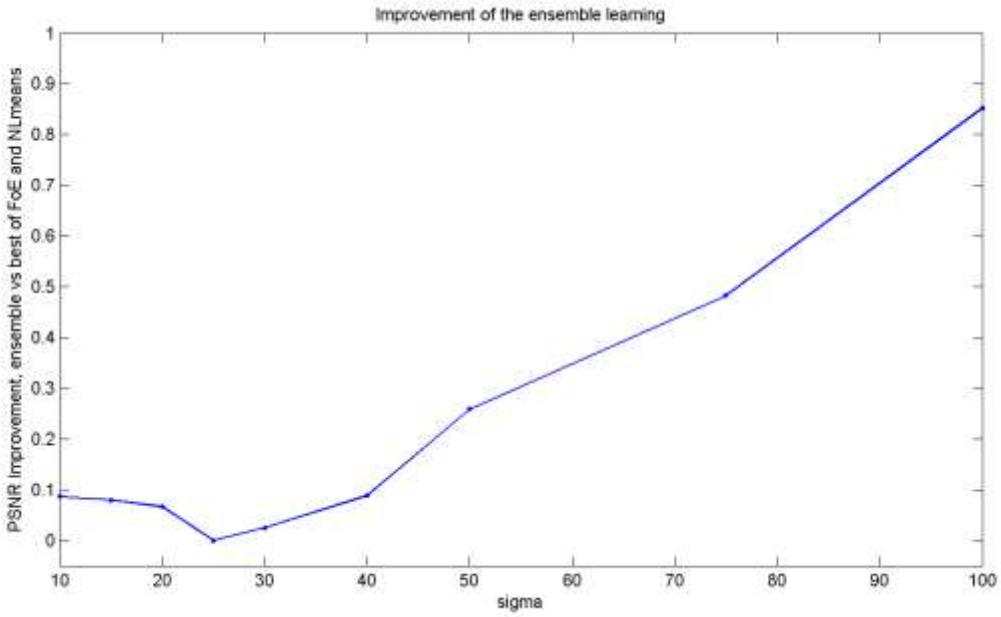


Figure 4.7. Improvement of the ensemble learning of the natural images comparing with the best result of FoE and NLM

4.2 Experimentation and Results of the medical images

4.2.1 Medical images with Gaussian noise

The ensemble learning was evaluated on medical images from the BrainWeb [15]. The dataset of MR images consists of 50 MR images and was randomly obtained from the BrainWeb. First, different numbers of input Gaussian noise, σ, were added to the original images. σ_{pseudo} was calculated with the selected σ_{NL} which is learned from the training dataset. N_{pseudo} was computed with the selected σ_{NL} and σ by using the formula (3.12) and (3.13). All the processes were the same as what we did on the natural images. 30 MR images were used for training σ_{pseudo}, and 20 MR images were used for testing. From the training result, we could get the best selection of σ_{pseudo}. The ensemble learning outperforms with the σ_{NL} = 10, 20, 30, 40, 50, 100, 100, 100 when the σ = 10, 15, 20, 30, 40, 50, 75, 100, respectively. Table 4.3 shows the average PSNR result of the 20 MR images denoised by using the ensemble learning with the selected σ_{NL}. Bold numbers were used to demonstrate the maximum PSNR results.

Table 4.3. The average PSNR (dB) result of the 20 MR images with Gaussian noise

σ	NLM	FoE	Ensemble learning
10	38.80	38.19	**39.23**
15	36.05	35.85	**36.52**
20	33.98	34.03	**34.52**
30	31.12	31.56	**31.83**
40	29.15	29.72	**29.99**
50	27.61	28.13	**28.52**
75	24.96	24.9	**25.95**
100	23.21	19.92	**24.42**

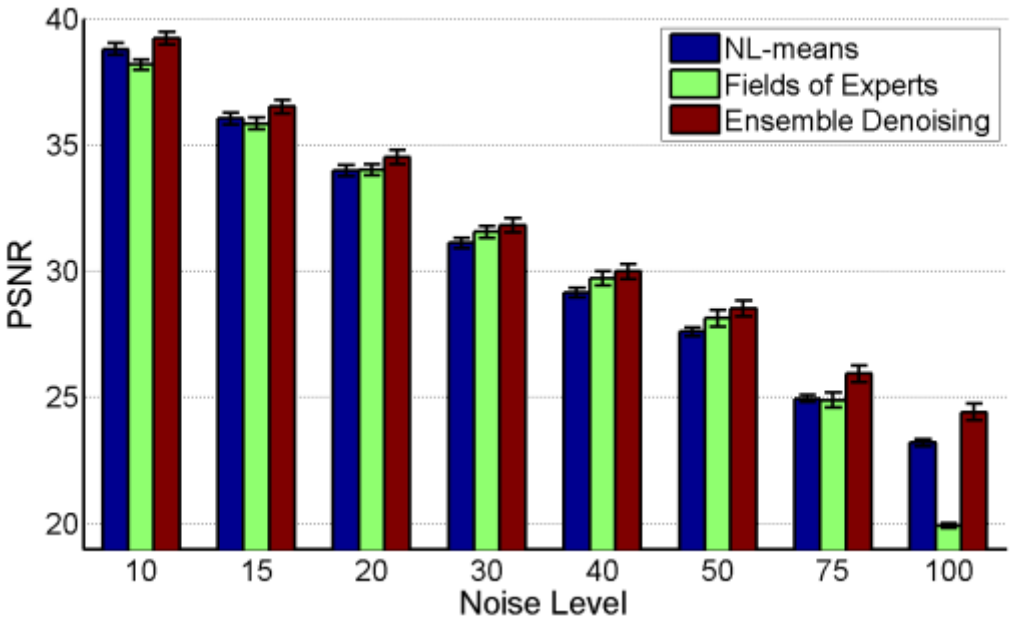


Figure 4.8. Average denoising PSNR results of 20 MR images with Gaussian noise. (From left to right, Non-local means, Fields of Experts, and Ensemble learning)

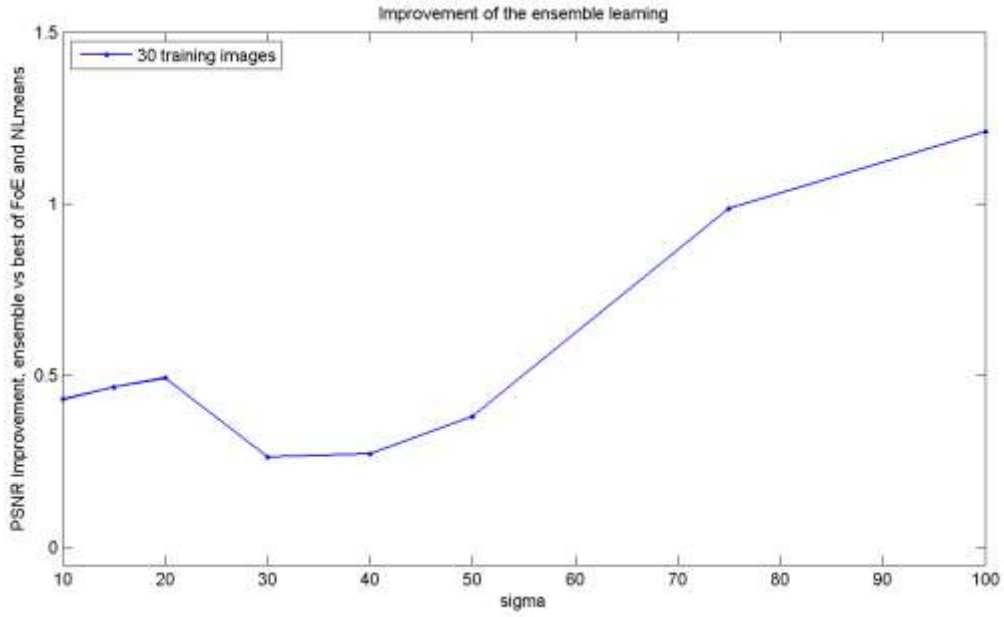
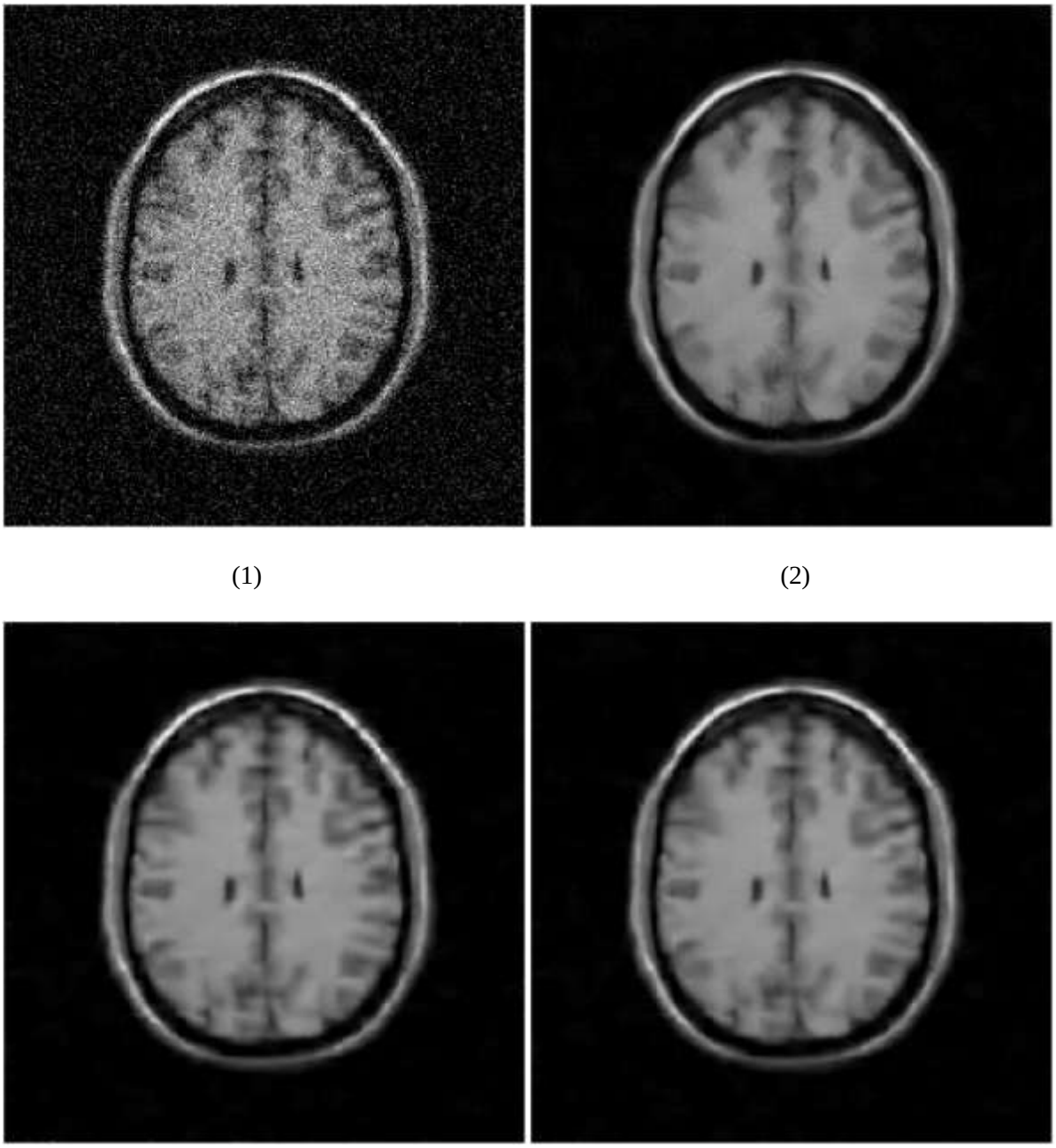


Figure 4.9. Improvement of the ensemble learning of the MR images with Gaussian noise comparing with the best result of FoE and NLM

Figure 4.8 shows the average denoising PSNR results of 20 MR images with the standard error. From the Table 4.3 and Figure 4.8, we could confirm that the ensemble learning outperformed the NLM and FoE in all the cases. Like the result of natural images, the ensemble learning showed a better improvement when the input noise increased. Also, the ensemble learning showed a better performance when the result of the NLM and FoE were similar. For example, when the input sigma is 75, the PSNR result of the NLM and FoE are 24.96dB and 24.90dB, respectively. The PSNR result of the ensemble learning is 25.95dB. Figure 4.9 shows an improvement of the ensemble learning of the MR images with Gaussian noise comparing with the best result of FoE and NLM. Figure 4.10 shows noisy images with Gaussian noise and results of ensemble learning with σ =30.



(1) (2)

(3) (4)

Figure 4.10. Denoising results. (1) Image with Gaussian noise, σ = 30 (PSNR = 18.61dB), (2) Denoised image using the NLM (PSNR = 31.27dB), (3) Denoised image using the FoE (PSNR = 31.93dB), (4) Denoised image using the ensemble learning (PSNR = 32.07dB) with σ_{NL} = 40

4.2.2 Medical images with Rician noise

Most denoising algorithms for medical images are built based on Rician noise distribution [10][29]. Unlike natural images, MR images can be obtained through a quadrature detector which that has the real and the imaginary parts. Therefore, Rician noise distribution is used for MR images. The ensemble learning for the medical image with Rician noise was evaluated on the same MR image dataset from the BrainWeb [15]. First, different numbers of input Rician noise were added to the original images instead of Gaussian noise. Rician noisy images can be made by using the formula (2.2). All the processes were the same as what we did on the natural images. Selecting the size of training and testing dataset was the same as what we did on the medical images with Gaussian noise. The dataset for training σ_{pseudo} consists of 30 MR images, and the dataset for testing is equivalent to 20 MR images. The ensemble learning outperformed with the σ_{NL} = 5, 20, 250, 500, 500, 500, 10, 10 when the σ = 10, 15, 20, 30, 40, 50, 75, 100, respectively. Table 4.4 shows the average PSNR result of the 20 MR images denoised by using the ensemble learning with the selected σ_{NL}. Bold numbers display the maximum PSNR results.

Table 4.4. The average PSNR (dB) result of the 20 MR images with Rician noise

σ	NLM	FoE	Ensemble learning
10	34.28	34.04	**34.35**
15	**31.15**	31.04	**31.15**
20	28.84	**28.88**	28.87
30	25.41	**25.66**	**25.66**
40	23.00	**23.29**	**23.29**
50	21.27	**21.42**	**21.42**
75	18.67	18.17	**18.71**
100	17.34	16.77	**17.38**

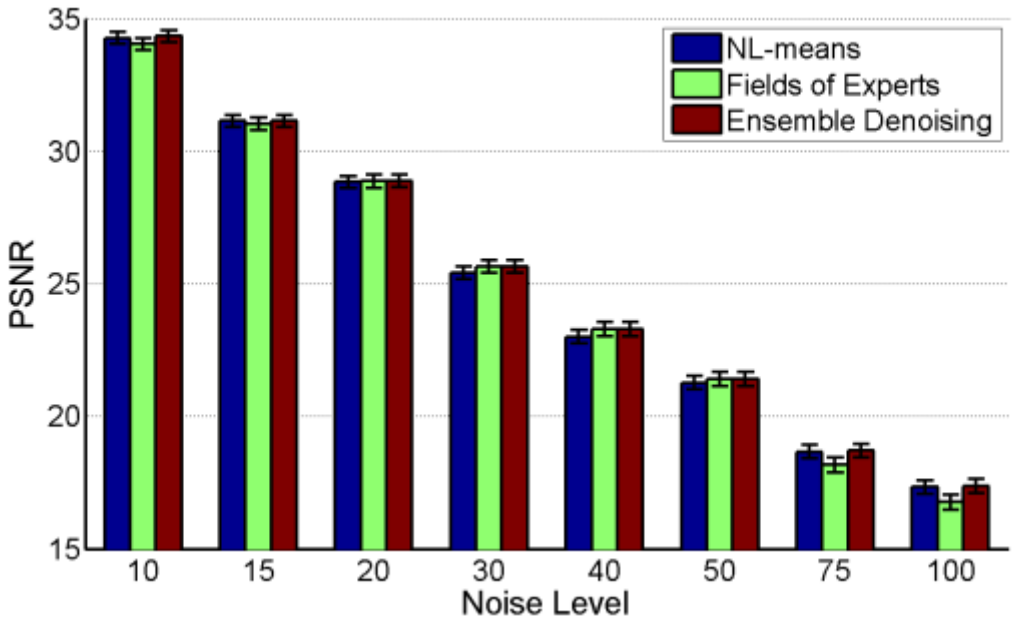


Figure 4.11. Average denoising PSNR results of 20 MR images with Rician noise. (From left to right, Non-local means, Fields of Experts, and Ensemble learning)

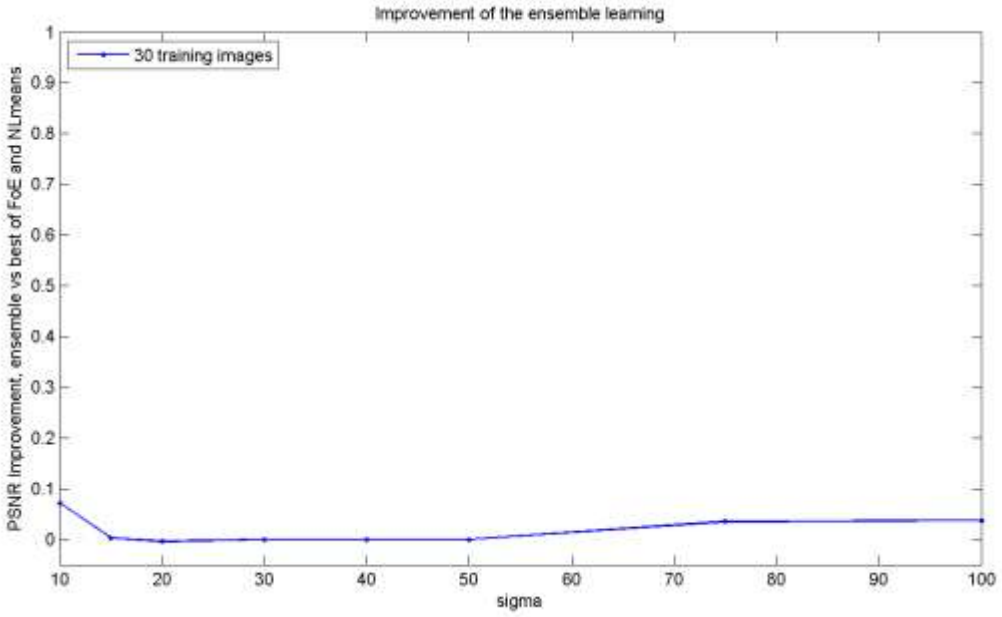
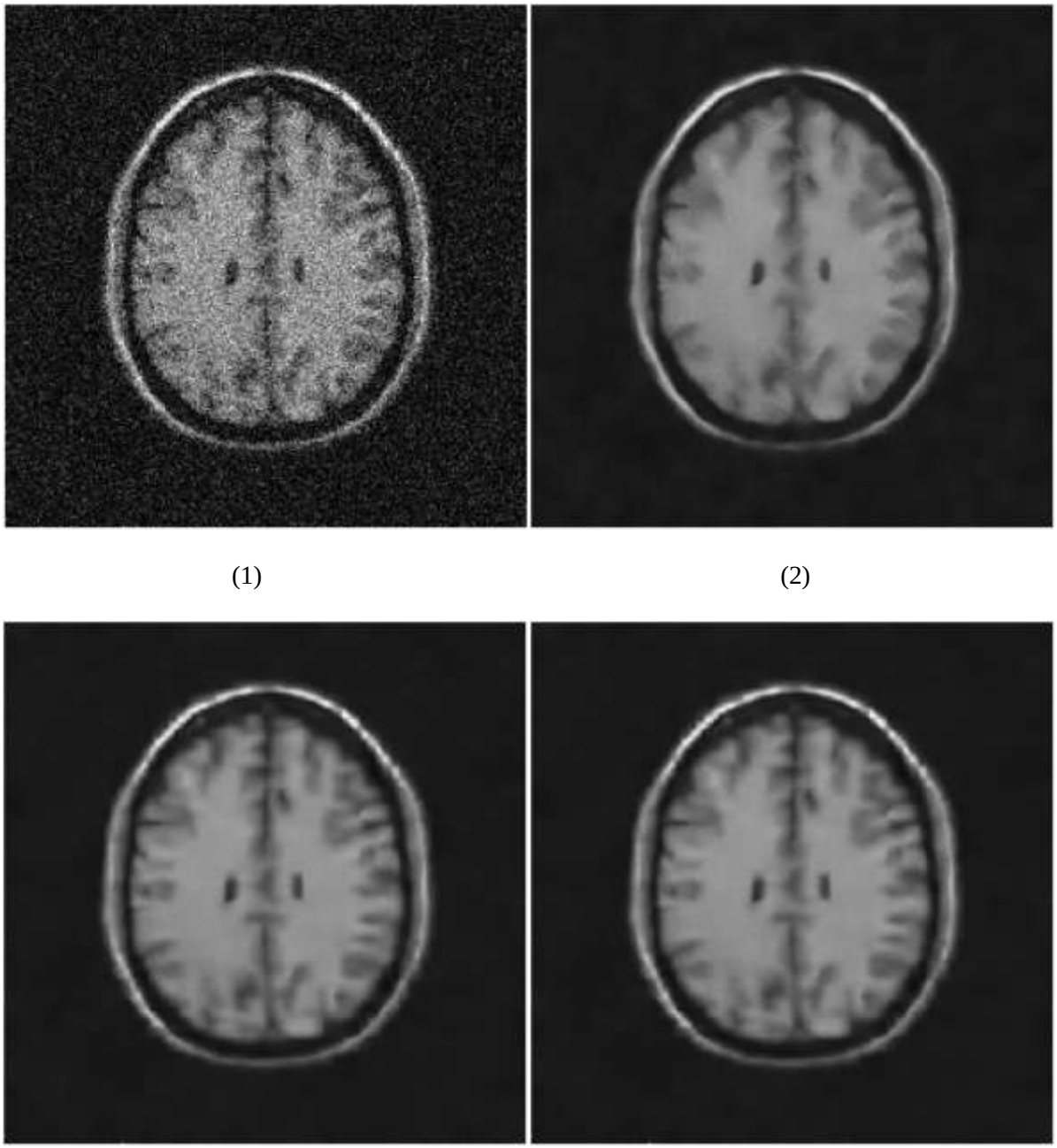


Figure 4.12. Improvement of the ensemble learning of the MR images with Rician noise comparing with the best result of FoE and NLM

Figure 4.11 shows the average denoising PSNR results of 20 MR images with the standard error. Unlike the PSNR results of 20 MR images with Gaussian noise, the PSNR results of 20 MR images with Rician noise did not show an outstanding improvement. The ensemble learning shows an improvement against the NLM in most cases except when the input sigma is 15. The results of ensemble learning have similar with FoE when the input sigma is 20, 30, 40, and 50, because of the selection of σ_{NL}. When the σ_{NL} increases like 250 or 500, the N_{pseudo} becomes similar to the N, which is the original noisy image. Therefore, the denoised results with this N_{pseudo} have almost same results like FoE. Figure 4.12 shows an improvement of the ensemble learning of the MR images with Rician noise comparing with the best results of FoE and NLM. Figure 4.13 shows the results of denoising images by using the NLM, FoE and ensemble learning. Because the assumption of FoE algorithm is based on Gaussian distribution, the ensemble learning with FoE did not perform well on Rician noise. However, FoE could be adapted to handle Rician noise, or alternate denoising methods could be selected for the ensemble learning.



(1) (2)

(3) (4)

Figure 4.13. Denoising results. (1) Image with Rician noise, σ = 30 (PSNR = 20.09dB), (2) Denoised image using the NLM (PSNR = 25.30dB), (3) Denoised image using the FoE (PSNR = 25.60dB), (4) Denoised image using the ensemble learning (PSNR = 25.60dB) with σ_{NL} = 500

5 Discussion and Conclusion

In this study, the ensemble learning based on the Bayesian model was built using the Non-local Means and Fields of Experts. The ensemble learning was used for image denoising of natural images and medical images with Gaussian and Rician noise. Provided algorithm codes of NLM and FoE were used, and the ensemble learning code was written with several lines of MATLAB codes. The Berkeley Segmentation database was used for natural images, and BrainWeb dataset was used for medical images. The training dataset of natural images consists of 40 images which were randomly selected from the Berkeley database. The σ_{NL} numbers, which were used to calculate σ_{pseudo}, were selected from the training dataset. Another set of 80 natural images from the Berkeley database were used for testing, along with 6 canonical images such as Barbara and Lena. The training dataset of MR images consists of 30 images which were chosen randomly from the BrainWeb database. Another set of 20 MR images were used for testing medical image denoising. Both Gaussian and Rician distribution were applied to build medical noisy images. PSNR was used to perform the quantitative comparisons with the original images and denoised images which were done by the NLM, FoE, and ensemble learning.

For natural images, the ensemble learning was able to acquire the denoised images from the noisy images. The results showed that the ensemble learning quantitatively had outperformed the NLM and FoE. The most PSNR results of the ensemble learning showed an advanced output, and its denoising performance augmented the quality of the denoised images when the input noise increased. Despite the PSNR of FoE

dropped rapidly when the input sigma went high such as from 75 to 100, the PSNR of ensemble learning slightly dropped similar to NLM. For example, when the input sigma dropped from 75 to 100, the PSNRs of FoE were 23.19dB and 19.47dB, respectively. However, the PSNRs of ensemble learning were 23.62dB and 22.75dB, respectively. When the PSNR of FoE was higher or similar to NLM, the PSNR of ensemble learning still outperformed both NLM and FoE. The ensemble learning could recover a disadvantage of NLM and FoE visually. The denoising results of FoE sometimes could lead to blur at some edges when the input sigma is high. Cyclic borders sometimes are shown on the denoising images of NLM. However, the ensemble learning is available to remove the cyclic borders from NLM and get better sharpened images at the edges. Therefore, the ensemble learning may have advantages over NLM and FoE.

For medical images, the ensemble learning outperformed the NLM and FoE with Gaussian noise and slightly exceeded the NLM and FoE with Rician noise. The ensemble learning for medical image with Gaussian noise denoising outperformed the NLM and FoE, like what we did on natural images. The PSNR results of the most cases showed improvements over NLM and FoE. For example, the average PSNR of NLM and FoE, when the input sigma was 50, were 27.61dB and 28.13dB, respectively. In this case, the average PSNR of the ensemble learning was 28.52dB.

The ensemble learning did not perform well on Rician noise because an assumption of FoE is based on Gaussian noise. However, the results still showed that the ensemble learning outperformed NLM and FoE, or similar with NLM and FoE. For example, the average PSNR of NLM and FoE, when the input sigma was 30, were 25.41dB and 25.66dB, respectively. The average PSNR of the ensemble learning was same as FoE, 25.66dB. In another case, the average PSNR of NLM and FoE, when the input sigma was 75,

were 18.67dB and 18.17dB, respectively. The average PSNR of the ensemble learning outperformed NLM and FoE, and the PSNR was 18.71dB.

There remain some features that could be improved to the ensemble learning for our future work. The calculation to get σ_{pseudo} and N_{pseudo} is based on the proportional value. In other words, we may need to consider the detailed calculation procedure to get the exact value of the σ_{pseudo} and N_{pseudo}. Our model for $p(D_{NL}|I)$ is simple, so this could be improved by learning it from natural images. Since the training dataset of the selection of proper σ_{pseudo} consists of 40 image patches from the Berkeley Segmentation Benchmark, we might need to consider more training datasets for accurate selection of σ_{pseudo}.

We also need to consider medical image denoising with Rician noise distribution. Because 20,000 image patches are selected randomly from the Berkeley Segmentation database, which is consisted of only natural images, to build unlikely 5x5 filters of FoE algorithm, it is required to build the filters by using a great diversity of medical images. FoE could be adapted to handle Rician noise with learned filters, or an alternative denoising method should be selected to build the ensemble learning. The idea of identifying features within the noisy images, such as edges, that predict whether NLM or FoE will be superior for that image region.

We can include additional non-probabilistic denoising methods (such as Gaussian Mixture [27] or Sparse coding [3]) into our algorithm. If we add more denoising method like Gaussian Mixture, the formula (3.6) will be written as follow:

$$P(\mathrm{I}|N, D_{Gau}, D_{NL}) = P(N|I, D_{Gau}, D_{NL}) \frac{P(I|D_{Gau}, D_{NL})}{P(N|D_{Gau}, D_{NL})} \quad (5.1)$$

The formula (5.1) can be written as follow:

$$P(\mathrm{I}|N, D_{Gau}, D_{NL}) \propto P(N|I)P(D_{Gau}, D_{NL}|I)\frac{P(I)}{P(D_{Gau}, D_{NL})} \tag{5.2}$$

$$\propto P(N|I)P(D_{Gau}|I)P(D_{NL}|I)P(I) \tag{5.3}$$

$P(N|I), P(D_{Gau}|I), (D_{NL}|I)$ are all Gaussian distribution and $P(I)$ is the Fields of Experts model. Therefore, we would be available to calculate σ_{pseudo} and N_{pseudo} again by using formula (3.8) – (3.12).

Future work will include further improvements by enhancing these features of the ensemble learning.

Bibliography

[1] L. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: Nonlinear Phenomena*, vol. 60, no. 1–4, pp. 259–268, 1992.

[2] S. P. Ghael, A. M. Sayeed, and R. G. Baraniuk, "Improved wavelet denoising via empirical Wiener filtering," in *Proceedings of SPIE*, 1997, vol. 3169, pp. 389–399.

[3] B. A. Olshausen and D. J. Field, "Sparse coding with an overcomplete basis set: a strategy employed by V1?," *Vision Research*, vol. 37, no. 23, pp. 3311–3325, 1997.

[4] A. Buades, B. Coll, and J. M. Morel, "A non-local algorithm for image denoising," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, 2005, vol. 2, pp. 60–65.

[5] S. Roth and M. J. Black, "Fields of experts," *International Journal of Computer Vision*, vol. 82, no. 2, pp. 205–229, 2009.

[6] U. Schmidt, Q. Gao, and S. Roth, "A generative perspective on MRFs in low-level vision," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, 2010, pp. 1751–1758.

[7] K. Dabov, A. Foi, and V. Katkovnik, "Image denoising with block-matching and 3 D filtering," *Proceedings of SPIE*, vol. 6064, pp. 1–12, 2006.

[8] 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," *Proceedings Eighth IEEE International Conference on Computer Vision ICCV 2001*, vol. 2, no. July, pp. 416–423, 2001.

[9] J. V. Manjón, J. Carbonell-Caballero, J. J. Lull, G. García-Martí, L. Martí-Bonmatí, and M. Robles, "MRI denoising using non-local means.," *Medical image analysis*, vol. 12, no. 4, pp. 514–23, Aug. 2008.

[10] H. Liu, C. Yang, N. Pan, E. Song, and R. Green, "Denoising 3D MR images by the enhanced non-local means filter for Rician noise.," *Magnetic resonance imaging*, vol. 28, no. 10, pp. 1485–96, Dec. 2010.

[11] P. Coupé, P. Yger, and C. Barillot, “Fast non local means denoising for 3D MR images.,” *Medical image computing and computer-assisted intervention : MICCAI ... International Conference on Medical Image Computing and Computer-Assisted Intervention*, vol. 9, no. Pt 2, pp. 33–40, Jan. 2006.

[12] T. Thaipanich and C. C. J. Kuo, “An adaptive nonlocal means scheme for medical image denoising,” in *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, 2010, vol. 7623, p. 21.

[13] R. Matsuura, K. Fuji, S. Goto, Y. Azuma, and K. Inamura, “The phantom for quality evaluations of the nonlinear noise reduction filter in multidetector row computed tomography,” in *IFMBE Proceedings*, 2009, vol. 25, no. 2, pp. 758–761.

[14] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity.,” *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, vol. 13, no. 4, pp. 600–12, Apr. 2004.

[15] “BrainWeb: Simulated Brain Database.” [Online]. Available: http://brainweb.bic.mni.mcgill.ca/brainweb/.

[16] M. Lindenbaum, M. Fischer, and A. Bruckstein, “On Gabor’s contribution to image enhancement,” *Pattern Recognition*, vol. 27, no. 1, pp. 1–8, 1994.

[17] H. Gudbjartsson and S. Patz, “The Rician distribution of noisy MRI data.,” *Magnetic Resonance in Medicine*, vol. 34, no. 6, pp. 910–914, 1995.

[18] J. Sijbers, A. J. Den Dekker, P. Scheunders, and D. Van Dyck, “Maximum-likelihood estimation of Rician distribution parameters.,” *IEEE Transactions on Medical Imaging*, vol. 17, no. 3, pp. 357–361, 1998.

[19] A. A. Efros and T. K. Leung, “Texture synthesis by non-parametric sampling,” *Proceedings of the Seventh IEEE International Conference on Computer Vision*, vol. 2, no. September, pp. 1033–1038 vol.2, 1999.

[20] P. Coupe, P. Yger, S. Prima, P. Hellier, C. Kervrann, and C. Barillot, *An Optimized Blockwise Nonlocal Means Denoising Filter for 3-D Magnetic Resonance Images*, vol. 27, no. 4. IEEE, 2008, pp. 425–41.

[21] S. Roth, “Fields of experts: A framework for learning image priors,” *2005. CVPR 2005. IEEE Computer Society*, vol. 2, pp. 860–867, 2005.

[22] R. E. Schapire, *The strength of weak learnability*, vol. 5, no. 2. Springer Netherlands, 1989, pp. 197–227.

[23] B. Efron and R. J. Tibshirani, *An Introduction to the Bootstrap*, vol. 57, no. 57. Chapman & Hall, 1993, p. 436.

[24] L. Breiman, "Stacked regressions," *Machine Learning*, vol. 24, no. 1, pp. 49–64, 1996.

[25] Y. Lou and P. Favaro, "Nonlocal similarity image filtering," *Image Analysis and Processing– ...*, pp. 1–10, 2009.

[26] A. Chambolle, "An Algorithm for Total Variation Minimization and Applications," *Journal of Mathematical Imaging and Vision*, vol. 20, no. 1/2, pp. 89–97, 2004.

[27] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, "Image denoising using scale mixtures of Gaussians in the wavelet domain.," *IEEE Transactions on Image Processing*, vol. 12, no. 11, pp. 1338–1351, 2003.

[28] "ITTC bioinformatics Cluster." [Online]. Available: http://bioinfo.ittc.ku.edu/resources.php.

[29] N. Wiest-Daesslé, S. Prima, P. Coupé, S. P. Morrissey, and C. Barillot, "Rician noise removal by non-Local Means filtering for low signal-to-noise ratio MRI: applications to DT-MRI.," *Medical Image Computing and Computer-Assisted Intervention*, vol. 11, no. Pt 2, pp. 171–179, 2008.