Bayesian ensemble learning for image denoising

Hyuntaek Oh
University of Kansas
Lawrence, Kansas 66045
hyuntaekoh@ku.edu

Brian Potetz
University of Kansas
Lawrence, Kansas 66045
potetz@ittc.ku.edu

Abstract

Natural images are often affected by random noise and the image denoising always has been issued in Computer Vision. Many algorithms have been introduced to remove the noise from the natural images, such as Gaussian, Wiener filtering and wavelet thresholding. However, many of these algorithms remove the fine edges and make them blur. Recently, many promising denoising algorithms have been introduced such as Non-local Means, Fields of Experts, and BM3D.

In this paper, we implement the Bayesian ensemble learning for image denoising. The Bayesian ensemble models are BM3D and Fields of Experts. BM3D, a block matching 3D, is in 3D transformation domain by integrating sliding-window convert processing with block-matching. A 3D array could be formed by piling the matched blocks which show high level of correlation. 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 natural images. The experimental results obtained are used to compare with the single algorithm and discuss the ensemble learning and their approaches.

1. Introduction

Most natural images contain some degree of natural and artificial noise. These noises usually affect the visual quality of the original images so the goal of image denoising is to reconstruct reasonable estimate of the original image from the noisy image. Ideally, the resulting denoising image will not contain any noise or added artifacts.

Major denoising algorithms include total variation minimization [1], Wiener filtering [2], Sparse Coding [3], etc. Most of these methods make assumptions about the image that can be lead to blurring. More algorithms have been developed such as Non-local means [4], Fields of Experts [5], and BM3D [6]. They have shown the promising denoising results rather than the old denoising algorithms.

In this paper, we study the ensemble learning based on Bayesian model with BM3D and Fields of Experts and use 6 different natural images for image denoising. BM3D is based on grouping. This is a process that finds similar 2D image blocks and piling them up in 3D arrays called grouping [6]. A 3D array shows high level of correlation because of the similarity between the grouped blocks. Fields of Experts, recently proposed by Roth and Black, is based on Markov random field. Fields of Experts develop a method for learning rich Markov random field image priors by exploiting ideas from Sparse image coding. In comparison with prior Markov random field approaches, all parameters in the Fields of Experts model are learned from a set of training data [5].

2. Background

2.1. Image Denoising

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

$$y(i) = x(i) + n(i)$$

where *y(i)* is the observed image, *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 noisy image. The noisy value, *n(i)*, is the Gaussian with known variance σ^2 and zero mean [4]. The ideal denoising algorithm is to remove the noisy, *n(i)*, and recover the original image, *x(i)*.

Previous methods such as Gaussian [7] 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 [6].

2.2. BM3D

BM3D was proposed by Kostadin Dabov and Karen Egiazarian. BM3D is based on the concept called block-matching and grouping. Block-matching is used to enhance the efficiency of program coding by using similarity between the blocks. After block-matching, we could utilize the blocks in 2D transform domain. Then blocks stack together in a 3D array called grouping [6].

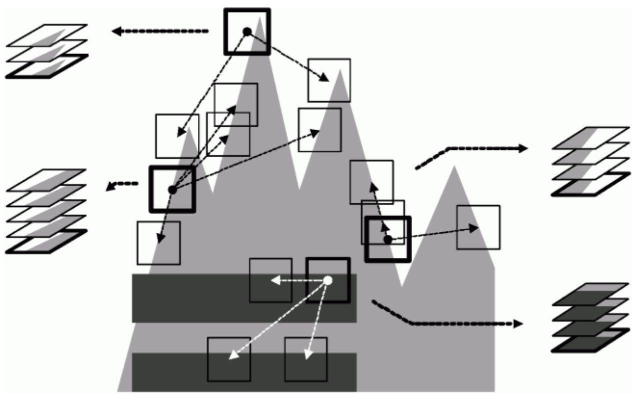

Figure 1. A simple example of the block-matching in an artificial image, where for each reference block (with thick borders) there exist perfectly similar ones.

The procedure of the BM3D denoising algorithm is the following.

1. *Block-matching*
   Find blocks that have high correlation to Z_{xR}, which is the currently processed block. Calculate the distance between two blocks to exhibit the high correlation. And then stack them together in a 3D array which we call group. The example of the grouping is explained in Figure 1.

2. *Denoising in 3D transform domain*
   Apply a unitary 3D transform to the groups and attenuate the noise by hard-thresholding the transform coefficients. Invert the 3D transform by the operator T_{3D}^{-1} to yield estimates of all grouped blocks. We can calculate the reconstructed 3D array, $\hat{Y}_{S_x}$ with the following formula:
   $$\hat{Y}_{S_x} = T_{3D}^{-1}\left(\gamma\left(T_{3D}\left(Z_{S_x}\right), \lambda_{thr3D}\sigma\sqrt{2\log(N_1^2)}\right)\right)$$
   where λ_{thr3D} is a fixed threshold parameter and γ is a hard-threshold operator. Return the estimates of the blocks to their original points.

3. *Aggregation*
   Compute the basic estimate of the output images by weighted averaging all of the obtained block-wise estimates that are overlapping.

This is the basic estimate and a detailed procedure of the BM3D denoising algorithm can be found in [6]. In this study, we use the BM3D algorithm code provided by the author and intentionally give only general parameters of the BM3D algorithm referring to the paper.

2.3. Fields of Experts

Fields of Experts was proposed by Stefan Roth and Michael J. Black. 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 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. To do this, Products of Experts were used. In comparison with prior Markov Random Field approaches, all parameters in the Fields of Experts model are learned from a set of training data [5]. Those models prior probability of images can be calculated with the following formula:

$$P(\vec{I}) \propto \prod_{C}\prod_{i=1}^{K}((1+\frac{1}{2}(\vec{I_c}\cdot\vec{J_c})^2)^{-\alpha_i}$$

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.

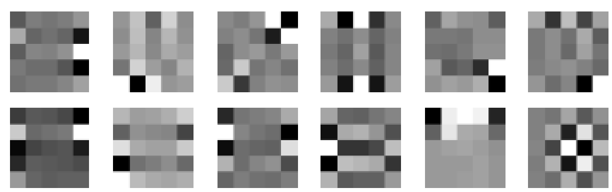

Figure 2. Selection of the 5x5 filters

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:

$$p(D|N) \propto p(N|D)p(D)$$

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

$$p(N|D) \propto \prod_{j} exp(-\frac{1}{2\sigma^2}(D_j - N_j)^2)$$

where σ is known standard deviation and D_j and N_j are the denoised and noisy image at pixel j, respectively. In this study, we use the Fields of Experts algorithm code provided by the author as same as what we did for BM3D and use the similar parameters to get the same results of the paper.

3. Application of the Ensemble learning

To recover the original image from the noisy image, Bayesian model with a prior value was used for the ensemble learning [8]. We can find the denoised image *D* that maximizes the prior probability with the following formula: $p(D|N, D_{BM})$ where D_{BM} is the denoised image after using BM3D algorithm. This formula can be calculated with Bayes Rule as follow:

$$p(D|N, D_{BM}) = p(N|D, D_{BM}) \frac{p(D|D_{BM})}{p(D_{BM})}$$
$$\propto p(N|D)p(D_{BM}|D)p(D)$$

where N is a noisy image with Gaussian, $p(D|N, D_{BM})$ is the prior probability of the denoised image by the ensemble learning. $p(D)$ is the denoised image by Fields of Experts. In this formula, $p(N|D)$ has been shown when the potentials are Gaussian and $p(D_{BM}|D)$ is the probability of the denoised image by using BM3D algorithm. These formulas can be written as follows:

$$p(N|D) \propto \exp(-\frac{\sum(N-D)^2}{2\sigma_N^2})$$

$$p(D_{BM}|D) \propto \exp(-\frac{\sum_{x,y}(D_{BM}(x,y) - D(x,y))^2}{2\sigma_{BM}^2})$$

where σ_N is the input Gaussian sigma value and σ_{BM} is another sigma value from the BM3D denoised image. We combined these two formulas to calculate the parameters σ_{pseudo} and N_{pseudo} as follows:

$$p(N|D)p(D_{BM}|D) \propto (\frac{(N-D)^2}{2\sigma_N^2} + \frac{(D_{BM}-D)^2}{2\sigma_{BM}^2})/(2\sigma_{BM}^2)$$
$$\propto \frac{(N_{pseudo} - D)^2}{2{\sigma_{pseudo}}^2}$$

We can get the parameters σ_{pseudo} and N_{pseudo} by summarizing the above formulas:

$$\frac{(N^2 - 2 \cdot \mathrm{N} \cdot \mathrm{D} + D^2)}{2\sigma_N^2} + \frac{(D_{BM}^2 - 2 \cdot D_{BM} \cdot D + D^2)}{2\sigma_{BM}^2}$$
$$\propto \frac{(N_{pseudo}^2 - 2 \cdot N_{pseudo} \cdot D + D^2)}{2{\sigma_{pseudo}}^2}$$

Therefore, the parameters σ_{pseudo} and N_{pseudo} are

${\sigma_{pseudo}}^2 = 1/\{(1/\sigma_N^2) + (1/\sigma_{BM}^2)\}$

$N_{pseudo} = \mathrm{N} \cdot \alpha + D_{BM} \cdot \beta$

where

$\alpha = (1/\sigma_N^2)/\{(1/\sigma_N^2) + (1/\sigma_{BM}^2)\}$

$\beta = (1/\sigma_{BM}^2)/\{(1/\sigma_N^2) + (1/\sigma_{BM}^2)\}$.

The MATLAB algorithm for the ensemble learning is as follows:

```
% perform BM3D denoising

files = filenames;
sigmas = [10 15 20 25 50];
for i = 1:length(files)
    orig_im =
double(imread('filename.png'));
    for j = 1:length(sigmas)
        BM3D algorithm
        save BM3D.mat
    end
end

% perform the ensemble learning with
different sigma_bm

sigma_bms = [1 2 3 .. 250 500];

for i = length(files)
    for j = 1:length(sigmas)
        load BM3D.mat
        for k = 1:length(sigma_bms)
            Fields of Experts algorithm
            save Ensemble_bm.mat
        end
    end
end
```

4. Experimentations and Results

The ensemble learning was evaluated on the six different natural images from the Berkeley Segmentation Database [9]. Different numbers of input Gaussian noise, σ, were added to the original image. We used the provided BM3D and Fields of Experts MATLAB code from the author's website and built the ensemble learning code with several lines of MATLAB codes [5, 6]. All the codes were run through the Bioinformatics Cluster at the Information and Telecommunication Technology Center at the University of Kansas. Single BM3D and Fields of Experts were evaluated to compare with the ensemble learning. Comparison between the ensemble learning, BM3D, and Fields of Experts were performed using the Peak to Signal to Noise Ratio (PSNR): $20log_{10}(255/\sigma)$, where σ is the standard deviation [10].

The noisy image was obtained from the original image with different numbers of input noise value, σ = 10, 15, 20, 25, 50. BM3D algorithm was used to get the D_{BM} and several numbers of sigma values from the BM3D denoised image were used, σ_{BM} = 1, 2, 3, 5, 10, 20, 30, 40, 50, 100, 250, 500. We used all these sigma values, BM3D denoised images and Fields of Experts algorithm to get the ensemble learning denoised images.

The 5x5 filter of Fields of Experts was used to obtain the denoised images. 5,000 iteration numbers were implemented for Fields of Experts [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 we calculate the average of their PSNR values with different input sigma values and other sigma values from the BM3D denoised images.

Table 1 and Figure 3 shows the average PSNR values of the BM3D, Fields of Experts and the ensemble learning with different number of input noise sigma and other sigma

Table 1. The average PSNR values from the six natural images

Sigma	*BM3D*	*00001*	*00002*	*00003*	*00005*	*00010*	*00020*	*00030*	*00040*	*00050*	*00100*	*00250*	*00500*	*FoE*
10	34.75	34.70	34.70	34.69	34.71	34.51	34.10	33.95	33.88	33.85	33.80	33.78	33.77	33.80
15	32.89	32.84	32.84	32.79	32.78	32.51	32.17	32.00	31.89	31.84	31.74	31.71	31.70	31.70
20	31.59	31.55	31.56	31.53	31.52	31.29	30.93	30.70	30.56	30.46	30.28	30.21	30.21	30.18
25	30.54	30.52	30.52	30.50	30.47	30.28	29.93	29.68	29.50	29.38	29.16	29.05	29.03	29.05
50	27.30	27.29	27.30	27.29	27.26	27.14	26.88	26.56	26.21	25.94	25.20	24.85	24.79	24.80

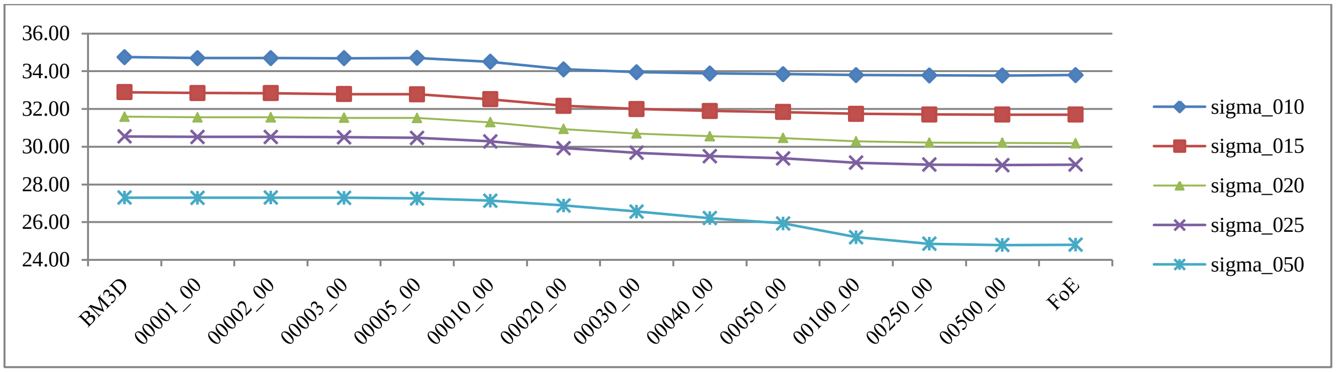

Figure 3. The average PSNR values

Figure 4. Denoising Results. (1) Original noiseless image. (2) Image with Gaussian noise, σ=25, (3) Denoised image using the BM3D, (4) Denoised image using the ensemble learning

values from the BM3D denoised images. The actual value from the each image is shown at the last page of this paper.

Figure 4 shows the results of the denoised images using the BM3D and ensemble learning. The image on the left is the original image without noise and the next image shows the noisy image with σ=25. The third image is the denoised image using the BM3D algorithm and the last is the denoised image using the ensemble learning. The PSNR between the original image and the BM3D denoised image was 30.637 and the PSNR between the original image and the ensemble learning was 30.242.

5. Summary and Conclusion

In this study, the ensemble learning based on the Bayesian model was built using the BM3D and Fields of Experts algorithm. We used the provided algorithm codes of BM3D and Fields of Experts and the ensemble learning code was written with several lines of MATLAB codes. Single BM3D and Fields of Experts were measure on the same natural images to compare with the ensemble learning. PSNR was used to perform the quantitative comparisons with the original images and the denoised images which were done by the ensemble learning. Because the ensemble learning was so time consuming, all the ensemble learning works were done by the cluster and it usually took around 5 hours per each image.

The results showed that the ensemble learning with BM3D and Fields of Experts had better result than the single Fields of Experts and similar value with the single BM3D. The ensemble learning result of the one image, Barbara, showed the better result than single BM3D and Fields of Experts. This result can be found at the end of this paper. However, most PSNR results of the ensemble learning from the different images did not show an advanced beyond the single BM3D. From the Table 1, the average PSNR value of BM3D with σ=25 was 30.54 when the ensemble learning showed 30.52, the maximum PSNR average value with σ_{BM}=2. Most results of the ensemble learning showed that the average PSNR values decreases when the σ_{BM} increases. The average PSNR values get close to the Fields of Experts results when the σ_{BM} increases and get close to the BM3D results when the σ_{BM} decreases.

Even though the average PSNR results of the ensemble learning showed an improvement comparing with the Fields of Experts, there still remains some features that could be improved the ensemble learning better than BM3D algorithm. The calculation to get the 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}. In addition, we should consider for the parameters which could be maximized the probability of the Fields of Experts algorithm. To do this, more trials with different parameters of the Fields of Experts algorithm would be required. Future work will include further improvements by enhancing these features of the ensemble learning.

6. Acknowledgements

The author would like to thank to Dr. Brian Potetz for his advice and helpful discussion during the process of this project.

7. References

[1] L.Rudin, S. Osher, and E. Fatemi. "*Nonlinear total variation based noise removal algorithms.*" Physica D, 60:259-268, 1992.

[2] S. Ghael, A. M. Sayeed, and R. G. Baraniuk, "*Improved wavelet denoising via empirical Wiener filtering,*" Proc. SPIE, Wavelet Applications in Signal and Image Processing V, col. 3169, pp. 389-399, Oct 1997.A. Alpher. Frobnication. Journal of Foo, 12(1):234–778, 2002.

[3] B. Olshausen and D. Field. "*Sparse coding with an over complete basis set: A strategy employed by V1?*", Vision Research, 37(23):3311-3325, 1997

[4] A. Buades, B. Coll, and J.M. Morel. "*A Non-local algorithm for image denoising*". In IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, 2005.

[5] S. Roth and M. J. Black. "*Fields of Experts: A framework for learning image priors*". In IEEE Conf. on Computer Vision and Pattern Recognition, 2005.

[6] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3D transform-domain collaborative filtering," IEEE Trans. Image Process., vol. 16, no. 8, pp. 2080-2095, August 2007.

[7] M. Lindenbaum, M. Fischer, and A. Bruckstein. On gabor contribution to image enhancement. Pattern Recognition, 27:1–8, 1994.

[8] Daphne Koller and Nir Friedman. Probabilistic graphical models. The MIT Press.

[9] http://www.cs.tut.fi/~foi/GCF-BM3D/

[10] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli. "Image quality assessment: From error visibility to structural similarity". IEEE Trans. Image Proc., 13(4):600-612, 2004.

Table 2. The PSNR values of the Barbara image ('barbara.png')

Sigma	*BM3D*	*00001*	*00002*	*00003*	*00005*	*00010*	*00020*	*00030*	*00040*	*00050*	*00100*	*00250*	*00500*	*FoE*
10	34.93	34.95	35.15	35.30	35.62	35.30	34.01	33.48	33.24	33.14	32.97	32.93	32.92	32.93
15	33.05	33.00	32.99	32.93	32.82	32.20	31.27	30.83	30.61	30.48	30.28	30.22	30.21	30.26
20	31.79	31.75	31.75	31.70	31.61	31.06	30.05	29.43	29.08	28.87	28.53	28.42	28.40	28.42
25	30.64	30.61	30.60	30.56	30.45	29.95	28.99	28.38	27.98	27.72	27.28	27.08	27.03	27.02
50	27.31	27.30	27.28	27.25	27.15	26.81	26.11	25.21	24.47	24.03	23.30	23.18	23.17	23.12

Table 3. The PSNR values of the Boat image ('boat.png')

Sigma	*BM3D*	*00001*	*00002*	*00003*	*00005*	*00010*	*00020*	*00030*	*00040*	*00050*	*00100*	*00250*	*00500*	*FoE*
10	33.88	33.84	33.76	33.71	33.70	33.55	33.35	33.30	33.27	33.26	33.25	33.24	33.22	33.28
15	32.11	32.08	32.03	32.00	31.99	31.80	31.61	31.54	31.50	31.48	31.43	31.41	31.41	31.41
20	30.83	30.80	30.77	30.75	30.75	30.57	30.37	30.28	30.23	30.16	30.09	30.05	30.07	30.07
25	29.84	29.82	29.81	29.79	29.77	29.63	29.45	29.32	29.24	29.17	29.06	28.99	28.98	28.99
50	26.67	26.66	26.65	26.64	26.61	26.51	26.34	26.18	25.96	25.75	25.19	24.91	24.86	24.85

Table 4. The PSNR values of the Fingerprint image ('fingerprint.png')

Sigma	*BM3D*	*00001*	*00002*	*00003*	*00005*	*00010*	*00020*	*00030*	*00040*	*00050*	*00100*	*00250*	*00500*	*FoE*
10	32.47	32.48	32.51	32.52	32.51	32.41	32.26	32.19	32.16	32.15	32.12	32.11	32.11	32.11
15	30.29	30.29	30.30	30.31	30.29	30.19	30.02	29.90	29.82	29.77	29.68	29.65	29.64	29.63
20	28.80	28.80	28.81	28.81	28.80	28.69	28.52	28.38	28.29	28.22	28.10	28.05	28.04	28.05
25	27.72	27.72	27.72	27.72	27.70	27.59	27.42	27.28	27.18	27.10	26.94	26.87	26.86	26.90
50	24.53	24.53	24.53	24.52	24.51	24.43	24.26	24.04	23.69	23.36	22.26	21.62	21.51	21.48

Table 5. The PSNR values of the House image ('house.png')

Sigma	*BM3D*	*00001*	*00002*	*00003*	*00005*	*00010*	*00020*	*00030*	*00040*	*00050*	*00100*	*00250*	*00500*	*FoE*
10	36.67	36.51	36.41	36.30	36.12	35.78	35.42	35.30	35.25	35.22	35.18	35.16	35.16	35.21
15	34.96	34.85	34.83	34.71	34.69	34.38	34.03	33.87	33.73	33.69	33.59	33.55	33.55	33.54
20	33.80	33.72	33.74	33.69	33.71	33.51	33.14	32.86	32.70	32.55	32.31	32.22	32.20	32.13
25	32.85	32.79	32.81	32.78	32.75	32.60	32.23	31.94	31.71	31.58	31.29	31.18	31.14	31.17
50	29.75	29.73	29.74	29.74	29.73	29.64	29.41	29.15	28.84	28.62	27.88	27.49	27.42	27.49

Table 6. The PSNR values of the Lena image ('lena.png')

Sigma	*BM3D*	*00001*	*00002*	*00003*	*00005*	*00010*	*00020*	*00030*	*00040*	*00050*	*00100*	*00250*	*00500*	*FoE*
10	35.87	35.77	35.72	35.69	35.66	35.51	35.27	35.21	35.17	35.14	35.12	35.12	35.10	35.16
15	34.27	34.19	34.20	34.15	34.18	33.96	33.70	33.59	33.51	33.47	33.41	33.37	33.36	33.36
20	33.04	32.99	33.00	32.97	33.00	32.78	32.54	32.39	32.28	32.23	32.08	32.05	32.04	32.00
25	32.06	32.02	32.04	32.03	32.01	31.86	31.60	31.40	31.25	31.17	31.02	30.93	30.92	30.91
50	29.02	29.01	29.02	29.02	29.02	28.97	28.79	28.56	28.28	28.04	27.36	27.00	26.93	26.86

Table 7. The PSNR values of the Peppers image ('peppers.png')

Sigma	*BM3D*	*00001*	*00002*	*00003*	*00005*	*00010*	*00020*	*00030*	*00040*	*00050*	*00100*	*00250*	*00500*	*FoE*
10	34.69	34.64	34.64	34.63	34.62	34.50	34.31	34.24	34.19	34.17	34.16	34.15	34.15	34.10
15	32.67	32.64	32.66	32.64	32.67	32.55	32.37	32.26	32.19	32.14	32.06	32.03	32.03	31.99
20	31.27	31.25	31.26	31.25	31.25	31.13	30.98	30.87	30.76	30.69	30.55	30.50	30.49	30.44
25	30.16	30.14	30.15	30.14	30.13	30.03	29.88	29.74	29.63	29.54	29.34	29.25	29.24	29.34
50	26.54	26.54	26.54	26.54	26.54	26.50	26.38	26.23	26.02	25.82	25.23	24.91	24.83	25.01