The Application of Markov Random Fields in Image De-noising

Yige Sun
Department of Electrical Engineering and Computer Science
Case Western Reserve University
Cleveland, USA
yxs871@case.edu

***Abstract*—In recent years, The researching in Image Processing becomes popular due to the high developing speed of the internet applications and the correspond wide spread of the image information. In the specific aspect as Image De-noising, several well developed methods have been purposed, such as Gaussian filtering [1], Independent Component Analysis(ICA) [2], Fast Fourier Transform [3] and specified neural networks [4] and so on. These methods have been confirmed to perform pretty good jobs for eliminating noises in different types. Besides all the popular methods mentioned above, Markov Random Fields(MRF) [5] have also been applied in image de-noising by considering each image pixel's relationships with its neighbors and assumed values. Fairly good results could also be get from this method. In this project, We build a traditional Markov Random Fields model based on the structures of sample images [13] and de-noised different type noise images as binary images and colored image by minimizing the energy function defined for each case in MRF. Also, two different methods for minimizing the energy function have been applied, first one is the transitional Gibbs Sampling [12] and second one is modified Iteration Conditional Modes(ICM) algorithm defined in this project. The result shows that the MRF based on Gibbs Sampling performed better in de-noising binary images and the MRF based on ICM performed better in de-noising colored image. In the colored image case, We also de-noise the image using other prepared packages, like Wavelet De-noising, chambolle Projections algorithm of Total Variation De-noising and Gaussian Fliter and so on. By comparing the final results in visual aspect and PSNR, the result images from the MRF model performed best.**

***Index Terms*—Image De-noising, Markov Random Fields, Gibbs Sampling, Iteration Conditional Modes, PSNR**

I. Introduction

To establish the images with decent clarity and clear details, image de-noising is essential for removing the unrelated images pixels and replace them with updated pixels which are fitted for the original image. So, to realize this, many efforts had been made to look for advanced methods based on previous version models or algorithms: S. G. Mallat et al purposed a wavelet de-noising approach which requires tracking or correlation of the wavelet maxima and minima across the different scales [6]. Then, in 1995, D. L. Donoho found the updated version called wavelet based threshold approach with specified threshold defined or attached [7], and this method no longer needs the requirements above. After that, in the aspects about calculating the threshold and its parameters, Data adaptive thresholds was introduced [8]. In recent years, Probabilistic models were found to be more attractive and more advanced than the previous threshold approachs. Hidden Markov Models [9] and Gaussian Scale Mixtures [10] were applied in image de-noising for exploring better visual results in probabilistic view. Besides this, neural networks [11], like deep learning, also play an important part in this field.

Here, The MRF based on Gibbs Sampling [13] and modified ICM were used to de-noise images in probabilistic view. For MRF, the Markov Random Fields is a type of undirected graph and contains nodes for representing each variables [12]. Each node is connected with all the other nodes or just a portion of them, each connection means the dependency relationship between each pair. A probability distribution for these variables or nodes could be defined based on the MRF structure. Also, the marginal probability for each node could be defined as the messages or information from its neighbors. The most important point for applying MRF model in this project is to consider several relationships or difference among the node, its neighbors and assumed values for it. For Gibbs sampling, it is one basic MCMC technique [12] [13]. By using Gibbs sampling, we could generate the updated posterior samples for each variable from the correspond conditional probabilities, which are defined in the MRF model, given other variables fixed in their current state. Then, we iterate this step till the updated samples are maintained in stable state. In this project, we used Gibbs Sampling to update the most possible values for each pixel based on the correspond conditional probabilities, which is defined in the MRF model, given its current version neighbors. For the modified ICM, it is almost same with Gibbs Sampling, we could also generate the updated posterior samples for each variable from the correspond conditional probabilities, which are defined in the MRF model, given other variables fixed. Then, we iterate this step again till these updated samples are maintained in stable state. The only difference between modified ICM and Gibbs Sampling is the definition of the conditional probabilities for each variable. In this project, by using Modified ICM, we also updated the most possible values for each pixel based on the correspond conditional probabilities, which is defined in the MRF model, given its original version neighbors.

By constructing the MRF model for the noise images and

implementing sampling and modified ICM, respectively, we could get the result de-noised images from both cases and compare these result images with the original true images visually. Besides this, the PSNR values from both cases could also be calculated and compared by using them as the measurements for the de-noising performances of each cases.

II. Methods

A. The Basic Ideas for De-noising Images based on the MRF Model

In general, the MRF model applied for the noise images could define a conditional probability as $p(y|x)$ where y denotes the potential real values of pixels and x represents the observed values of the pixels. Based on this definition, we intend to find the most possible y from all candidates given x. The result y is the correspond de-noised image.

We could also find another method, $p(y[i,j]|N(y[i,j]))$ is defined for each pixel $y[i,j]$ given its neighbors $N(y[i,j])$. From $p(y[i,j]|N(y[i,j]))$ for $y[i,j]$, we could get the most possible $y[i,j]$ from its candidates under $N(y[i,j])$ being fixed. The result set $\{y[i,j]\}$ is the de-noised image.

Considering the computing load of each method: Setting binary image as an example, for the first method choice, if we need to de-noise a sample binary image with $M \times N$ size, y could have 2^{MN} candidates for the relative algorithms to try one by one. Too much time would be consumed during this de-noising process, which can't allow us to implement this method in our program. But, for the second choice, each pixels $y[i,j]$ have 2 choices, and we just need to calculate $2MN$ times in total under relative algorithms. The time consumed in this case would be much smaller than the first choice mentioned above. So, in this project, we found the most possible values for pixels based on the second method in MRF model.

B. The Methods for Forming Noise Images

For binary images case: Under knowing the sample image have size $M \times N$ with only one layer, we randomly choose about 20% pixels among all pixels in this image and change their values randomly into new values ranged from 0 to 1.

For colored image case: Under knowing the sample image have size $M \times N \times 3$ with 3 layers, we randomly choose about 7% pixels in each layer and change their values randomly into new integers ranged from 0 to 255. The reason to choose 7% pixels in each image layer is that, under knowing the locations of noise pixels are almost totally different with each other due to the random characteristic of their distribution in each layer, the fractions of the noise pixels in the sample image that are directly established to us would be about $3 \times 7\% = 21\% \approx 20\%$ which is quite cohered with the that of binary case.

C. The Markov Random Fields Structures for the Noise Images

The MRF models built for each noise image layer contains two main layers. We set each observed pixel $x[i,j]$ from the same image layer as the node $X[i,j]$ and form the upper layer called observed layer X. We also built another MRF layer Y contains the same number of nodes with the observed layer, and each node $Y[i,j]$ is set just below the correspond $X[i,j]$ in the observed layer. $Y[i,j]$ represents the potential true value $y[i,j]$ of the observed pixel $x[i,j]$, we call this layer Y as hidden layer. Then, for the connections between nodes in same layer and different layers, we initially connect each node $Y[i,j]$ with its neighbors in the same hidden layer and then connect it with the correspond node $X[i,j]$ above. In this case, we connect $Y[i,j]$ with all its neighbors in this structures for considering the dependency relationships between $Y[i,j]$ and its neighbors in this structure. The Figure.1 shows a MRF model built by assigning 5 neighbors for each $Y[i,j]$. In this

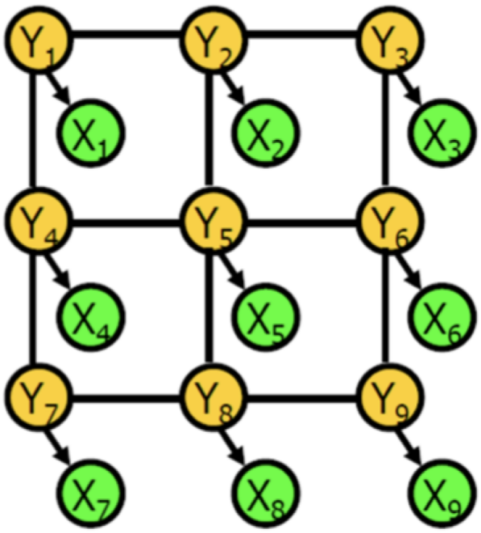

Fig. 1. The MRF model built with 5 neighbors for each $Y[i,j]$ [13]

project, the MRF model structure is designed with 9 neighbors for each $Y[i,j]$, which means each node $Y[i,j]$ in hidden layer is connect to its 8 neighbors in the same layer and $X[i,j]$ just above it. In this case, we consider the dependency relationships between $Y[i,j]$ and all its 9 neighbors mentioned above.

For binary images with only one layer, we build only one this two-layer structure for each image. For the colored image with 3 layers, we build the this two-layer structure for each image layer. After finding all the most possible $y[i,j]$ for each $Y[i,j]$ in the hidden layers, the output Y forms the final de-noised image.

D. Energy function

After the structure of the MRF model is defined, we could define the conditional probability for each $Y[i,j]$ given its 9 neighbors $N(Y[i,j])$ as $p(Y[i,j]|N(Y[i,j]))$ for considering the dependency relationships between $Y[i,j]$ and $N(Y[i,j])$ defined in the MRF model. Based on the given $N(Y[i,j])$, we could get the most possible candidates $y[i,j]$ for each $Y[i,j]$ from these conditional probabilities.

For the specific definition of $p(Y[i,j]|N(Y[i,j]))$, we could write the conditional probability as,

$$p(Y[i,j]|N(Y[i,j])) = \frac{1}{Z}e^{-E(Y[i,j]|N(Y[i,j]))} \tag{1}$$

where Z is the normalize parameter and could be viewed as a constant here. Based on equation (1), we could find the most possible candidates $y[i,j]$ for each $Y[i,j]$ by minimizing the energy function $E(Y[i,j]|N(Y[i,j]))$. Considering the various dependency relationships between $Y[i,j]$ and its neighbors $N(Y[i,j])$, we define the $E(Y[i,j]|N(Y[i,j]))$ in terms of D, G and F,

$$\begin{aligned} &E(Y[i,j]|N(Y[i,j])) = D + G + F \\ &D = D(Y[i,j]|N(Y[i,j]) \\ &G = G(Y[i,j]|N(Y[i,j]) \\ &F = F(Y[i,j]|N(Y[i,j]) \end{aligned} \tag{2}$$

For each term:

- $D = D(Y[i,j]|N(Y[i,j])$:
  This term is defined as,

$$\begin{aligned} &D(Y[i,j] = y[i,j]|N(Y[i,j]) = w_1 D_s + w_2 D_b \\ &D_s = \sum_{n \in N(Y[i,j]/x_{i,j})} |y[i,j] - n|^{t_D} \\ &D_b = |y[i,j] - x[i,j]|^{t_D} \end{aligned} \tag{3}$$

  D is intend to measure the difference between the $Y[i,j]$ and its neighbors, D_s denotes the difference between $Y[i,j]$ and its 8 neighbors in same hidden layer and D_b denotes the difference between $Y[i,j]$ and the correspond observed value $X[i,j] = x[i,j]$. w_1 and w_2 are the weight parameters for D_s, D_b, respectively. t_D is the parameter to make the definition of D be more like the that of "distance". Based on equation (3), If potential value $y[i,j]$ for $Y[i,j]$ have small difference with its neighbors, $D = D(Y[i,j]|N(Y[i,j])$ would have small value, which also make $E(Y[i,j]|N(Y[i,j])$ tend to be relatively small. So, the purpose for setting D is to realize the characteristic: The closer the $Y[i,j]$ being with its neighbors, the higher possibility that it would be the candidate that we intend to find.
- $G = G(Y[i,j]|N(Y[i,j])$:
  This term is defined as:

$$\begin{aligned} &G(Y[i,j] = y[i,j]|N(Y[i,j]) = \lambda_G|\bar{N} - y[i,j]|^{t_G} \\ &\bar{N} = \frac{1}{8}\sum_{n \in N(Y[i,j]/x_{i,j})} n \end{aligned} \tag{4}$$

  G is intend to consider the gradient change among the 3 region with $Y[i,j]$ centered in the hidden layer. In the equation (4), $\bar{N}$ denotes the average of the 8 neighbors in same layer with $Y[i,j]$, λ_G denotes the weight parameter designed for G in equation (2) and t_G is also the parameter for making the definition of G be more like the that of "distance". Based on equation (4), we thought the most possible $y[i,j]$ for $Y[i,j]$ should also not make a dramatic changing of gradient in this 3×3 region.
  To make the images be more "smooth" without protruding pixel established in them, the gradient in this 3×3 region with $Y[i,j]$ centered should also be maintained into a stable value or constant which is specific for this region. In this case, given the 8 neighbors of $Y[i,j]$, the pixels' values in this region should be changed smoothly with stable gradient. To realize this, what we do is to make the $Y[i,j]$ be close enough to the average of 8 neighbors.
  The closer $Y[i,j]$ to this average $\bar{N}$, the lower G would be. Also, the result $E(Y[i,j]|N(Y[i,j])$ would tend to be relatively small. So, the purpose for setting G is to realize the characteristic: The closer the $Y[i,j]$ being with the $\bar{N}$, the higher possibility that it could be the candidate that we intend to find.
- $F = F(Y[i,j]|N(Y[i,j])$:
  This term is defined as:

$$\begin{aligned} &F(Y[i,j] = y[i,j]|N(Y[i,j]) = \lambda_F|S_F - y[i,j]|^{t_F} \\ &S_F = \frac{1}{2}(Y_{c1} + Y_{c2}) \end{aligned} \tag{5}$$

  F is intend to consider the feature characteristic in this 3 region with $Y[i,j]$ centered in the hidden layer. In the equation (5), λ_F denotes the weight designed for F in equation (2) and t_F is the parameter for making the definition of F be more like the that of "distance". Y_{c1} and Y_{c2} are the two neighbors on the opposite sides of $Y[i,j]$ with the minimum difference compare to the other 3 pairs: In this part, we consider the features established in this region should be simple lines due to tiny size of this region. If one of these line get through $Y[i,j]$ and other two neighbors on its opposites sides, this feature line would be the most obvious feature in this region and should be highly considered. Because we don't know whether the feature line get through the $Y[i,j]$ or not, here, we directly make an assumption that the feature line getting through $Y[i,j]$ would always exist in any 3 region picked from the hidden layer. Based on this assumption, there are four possible angles of this feature line in this 3×3 region with the horizontal direction: $0°$, $45°$, $90°$, $135°$. For each angle case, we calculate the difference between the two neighbors that the feature line get through, as $\Delta Y = |Y_{c1} - Y_{c2}|$,

$$\begin{aligned} &\Delta Y_0 = |Y_{i-1,j} - Y_{i+1,j}| \\ &\Delta Y_{45} = |Y_{i-1,j-1} - Y_{i+1,j+1}| \\ &\Delta Y_{90} = |Y_{i,j-1} - Y_{i,j+1}| \\ &\Delta Y_{135} = |Y_{i-1,j+1} - Y_{i+1,j-1}| \end{aligned} \tag{6}$$

  where $Y[i,j]$ is the center of the 3×3 region. By comparing the set $\{\Delta Y_0, \Delta Y_{45}, \Delta Y_{90}, \Delta Y_{135}\}$, we could find the minimum of them which means that the difference of the correspond two neighbors on the opposite sides of $Y[i,j]$ is lowest compare to other pairs, then the feature line getting through $Y[i,j]$ should also form in the correspond angle because these two neighbors are the most close pair

which is required by the feature line. Here, we define the these two neighbors with lowest difference mentioned above as Y_{c1} and Y_{c2}, respectively. Based on the Y_{c1}, Y_{c2} we find, I intend to find the $y[i,j]$ of $Y[i,j]$ being close to the average of Y_{c1} and Y_{c2} because the forming of the feature line requires these three nodes contain stable value. So, this is the reason why the equation (6) is formed in this part: The closer $Y[i,j]$ being with $\frac{1}{2}(Y_{c1}+Y_{c2})$, the lower F would be, which also make the result $E(Y[i,j]|N(Y[i,j])$ tend to be relatively small. So, the purpose for setting F is to realize the characteristic: The closer the $Y[i,j]$ being with the $\frac{1}{2}(Y_{c1}+Y_{c2})$, the higher possibility that it could be the candidate that we intend to find to establish the feature characteristic in its local region.

E. Gibbs Sampling

Based on the definition of Gibbs Sampling mentioned in **Introduction**, in this project, we found the most possible $y[i,j]$ for each $Y[i,j]$ given its neighbors from conditional probability $p(Y[i,j]|N(Y[i,j]))$, we update this $Y[i,j] = y[i,j]$ in the hidden layer. After several iterations for updating the whole hidden layer, the value distribution in this layer are converged into stable state and forms the output. The detail algorithm is shown below.

- Step.1 Set the initial value of the hidden layer: In this step, we directly set the $Y[i,j] = x[i,j]$ which means that we set the observed pixels' values as the initial values of the correspond hidden layer nodes.
- Step.2 Find the most possible $y[i,j]$ for first $Y[i,j]$ in the hidden layer: In this step, based on the conditional probability $p(Y[i,j]|N(Y[i,j]))$ defined in equation (2), we could find the most possible $y[i,j]$ from all candidates for certain $Y[i,j]$. Then, we update $Y[i,j] = y[i,j]$ in the hidden layer.
- Step.3 Update the followed $Y[i,j]$ in the hidden layer: In the step, as what we do in the previous step, we update the follows $Y[i,j]$ in the hidden layer.
- Step.4 Iterate the Step.2 and Step.3 for certain number of times: In this step, we iterate the Step.2-3 for several times which could be determined, till the output hidden layer from each iteration are converged into a stable state.

F. Modified Iteration Component Modes

The work mechanism of Original ICM(Iteration Component Modes) is similar with Gibbs sampling mentioned above. Here, in this section, we modify the original type ICM to emphasize the consideration of the local information in the noise images. The algorithm below show the Modified ICM.

- Step.1 Set the initial value of the hidden layer: In this step, we also directly set the $Y[i,j] = x[i,j]$ that the observed pixels' values is set as the initial values of the correspond hidden layer nodes.
- Step.2 Find the most possible $y[i,j]$ for first $Y[i,j]$ in the hidden layer: In this step, also, based on the conditional probability $p(Y[i,j]|N(Y[i,j]))$ defined in equation (2), we could find the most possible $y[i,j]$ from all candidates for certain $Y[i,j]$. Then, we update $Y[i,j] = y[i,j]$ in the hidden layer.
- Step.3 Update the followed $Y[i,j]$ in the hidden layer: This step have huge difference with that of Gibbs Sampling. We update the follows $Y[i,j]$ in the hidden layer given its neighbors $N(Y[i,j])$ in their initial state from the correspond original noise image layer. This difference could enable us to emphasize the consideration of local information from noise images into de-noise processes.
- Step.4 Iterate the Step.2 and Step.3 for certain number of times: In this step, we iterate the Step.2-3 for several times which could be determined, till the output hidden layer from each iteration are converged into a stable state.

G. Peak Signal-to-Noise Ratio

In this project, we use Peak Signal-to-Noise Ratio(PSNR) to measure the difference between original true image I_{true} and the correspond de-noised image I_{de} for analyzing the performance of the de-noising model or algorithm.

Under assuming the size of I_{true} and I_{de} is $M \times N$, the PSNR between them P,

$$\begin{aligned} P &= 20log_{10}(\sqrt{\frac{R^2}{MSE}}) \\ MSE &= \frac{1}{MN}\sum_{[i,j]\in[M,N]} |I_{true}[i,j] - I_{de}[i,j]|^2 \end{aligned} \tag{7}$$

where R is the maximum fluctuation of pixels in both images, for binary image, $R = 1$ and for colored image, $R = 255$. MSE is the mean squared error between I_{true} and I_{de}. The lower MSE is, the better performance of correspond de-noising algorithm would be. So, from equation (7), the higher P is, the better performance of correspond de-noising algorithm would be.

III. The MRF Model for Binary Image cases

It should be mentioned that, to maintain all $Y[i,j]$ in the hidden layer in correspond MRF models have same number neighbors, for binary images cases with $M \times N$ size, we expand one extra outer layer with 0 for this image before applying it in MRF model, the result image has $M+2 \times N+2$ size. For colored image case with $M \times N$ size, we also expand one extra outer layer for each image layer with 255 before applying the whole image into MRF model, the result image has $M + 2 \times N + 2 \times 3$ size. In this case, all the nodes $Y[i,j]$ would have 8 "neighbors" in the hidden layer with $M \times N$ size.

Also, based on the algorithm characteristics of Gibbs sampling and Modified ICM built in this project, we universally determine the number of iteration as 1 under considering the output de-noised images in visual clarity and computing loads aspects.

For binary images case, we firstly set the parameters $w_1 = w_2 = 1$, and $t_D = 2$ for D, $\lambda_G = 1$ and $t_G = 1$ for G, $\lambda_F = 1$ and $t_F = 1$ for F. Under these parameters setting, we de-noise the noise images produced from original true image with Gibbs

Sampling and Modified ICM, respectively. The result de-noise images from both sampling methods are quite visually good which are shown in Figure.2 From these four plots, the two de-noised images from Gibbs Sampling and Modified ICM almost be similar with the original true images. Compare the PSNR value for both methods, the PSNR of de-noised image from Gibbs Sampling as $18.7134dB$ is lower than the PSNR of de-noised image from Modified ICM as $19.1633dB$, which means the de-noising performance of MRF model based on Modified ICM is better than that based on Gibbs Sampling. However, if we compare both de-noised images virtually, we could found that these are still some noise points randomly distribute in the de-noised image from Modified ICM but the de-noised image from Gibbs Sampling is formed in quite clear and clean appearance or look.

In this case, we also de-noise another type binary image under same parameter setting and the results are shown in Figure.3. From the Figure.3, we could find that the de-noising result for new type image is quite same with the previous version: After de-noise the new type noise image with the MRF model based on Gibbs Sampling and Modified ICM, respectively. The PSNR of the de-noised image from Gibbs Sampling as $17.0163dB$ is also lower than the PSNR of the de-noised image from Modified ICM as $17.8989dB$. What's more, if we compare this two de-noised image visually, we could find that the de-noised image from Gibbs Sampling with clean appearance is better then another one which is distributed with several noise points. Based on the de-noising results from these two type binary images shown above, though the MRF model based on Modified ICM performs better in image de-noising compare to MRF model based on Gibbs Sampling in PSNR aspect, the visual results from MRF model based on Gibbs Sampling is better compare to those from MRF model based on Modified ICM in visual aspect which is more essential in Image De-noising. So, we could know that the MRF model based on Gibbs Sampling is more suitable than MRF model based on Modified ICM in de-noising binary images.

Based on the conclusion drawn above, we also try different parameter setting to test the performances of the MRF model based on Gibbs Sampling in de-noising images under various parameters. The result plots are shown in Figure.4

By changing $w_2 = 0$ from initial parameter setting, we get the second plot in Figure.4. The main letters and logo in the first one which is under initial parameter setting are disappeared in the second figure, and the PSNR value is obvious decreased to $8.2828dB$ which means the performance of the MRF model based on Gibbs Sampling is decreased under ignoring the effects from the observed value $x[i, j]$ on $Y[i, j]$ in equation (2). So it seems that the difference between potential $Y[i, j]$ and correspond $x[i, j]$ is an essential factor in de-noising binary images under the MRF model based on Gibbs Sampling. For the last two plot, the third one is formed under changing $\lambda_G = 20$ and $n_2 = 3$ from initial state for improve the effect from G on $Y[i, j]$ under equation (2), and fourth one is formed under changing $\lambda_F = 20$ and $n_3 = 4$ from initial state for improve the effect from F on $Y[i, j]$ under equation (2). Though the PSNR values of both de-noised images are decreased from the one under initial parameter setting, the smoothness of both images are improved compare to the first plot under emphasized considering the effects of gradient stability and clarity of feature lines in the correspond images, respectively: If we see these two plots in detail, the letter "N" and lower corner of letter "N" for both plots are smoother than first plot, what's more, the track in letter "H" are partly filled in both plots but in first plot.

However, by redrawing the last two plots under reproduced noise images, this smoothness effects from G and F seem unstable, which means sometimes the smoothness effects would be shown in these plots, but sometimes not. But, in general, because the smoothness effects from G and F are established in this case, it means that G and F are still effective for $Y[i, j]$.

In general, four plots in Figure.4 are almost same, which means the parameter set changing in the MRF model based on Gibbs Sampling have few effects on the result de-noising in binary cases.

IV. The MRF Model for Colored Image case

For colored image case, we choose a small size image with 100×100 pixels under considering the time consuming and computing loads. After finishing add noise in the original true image and form the correspond image, we de-noise the noise image by MRF models based Gibbs Sampling, Modified ICM, respectively. Here, we finally decide the initial parameter set as $w_1 = 1$, $w_2 = 0$, and $t_D = 1$ for D, $\lambda_G = 0.5$ and $t_G = 1$ for G, $\lambda_F = 1$ and $t_F = 1$ for F after several time experiments and comparisons. w_2 means that we ignore the the effects from the observed value $x[i, j]$ on $Y[i, j]$ in equation (2) for this case, and $\lambda_G = \lambda_F = 0.5$ means that we decrease the weights of G and F in equation (2) and so as the effects from them on $Y[i, j]$. The de-noising results are shown in Figure.5. From the Figure.5, we could find that de-noised image from the MRF model based on Modified ICM is much better than that from the MRF model based on Gibbs Sampling, no matter in PSNR aspect or visual aspect: From the PSBR aspect, the PSBR value of the de-noised image under Gibbs Sampling as $27.8259dB$ is smaller than $29.6230dB$ of the de-noised image under Modified ICM, which means that the performance of the MRF model based on Modified ICM in de-noising colored images is much better than the MRF model based on Gibbs Sampling. From the visual aspect, the de-noised image under Gibbs Sampling is seriously distorted compare to original true image, but the de-noised image under Modified ICM could still maintain decent appearance and quite same shapes with original true image though the detail results are not awesome enough compare to original true image. Based on these two aspects, we are sure that the MRF model based on Modified ICM is more suitable to de-noise colored image than that based on Gibbs Sampling. Under knowing the RF model based on Modified ICM is fed with the neighbors for $Y[i, j]$ with original values in noise images, the local information

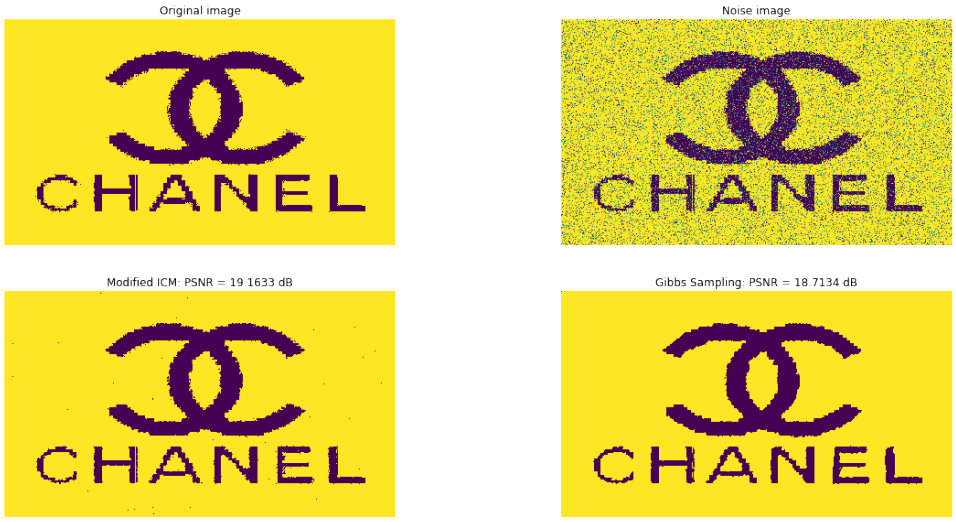

Fig. 2. The four plots in different conditions: The upper left is the original true image, the upper right one is the noised image. The lower right one is the de-noise image based on Gibbs Sampling, and left one is based on Modified ICM.

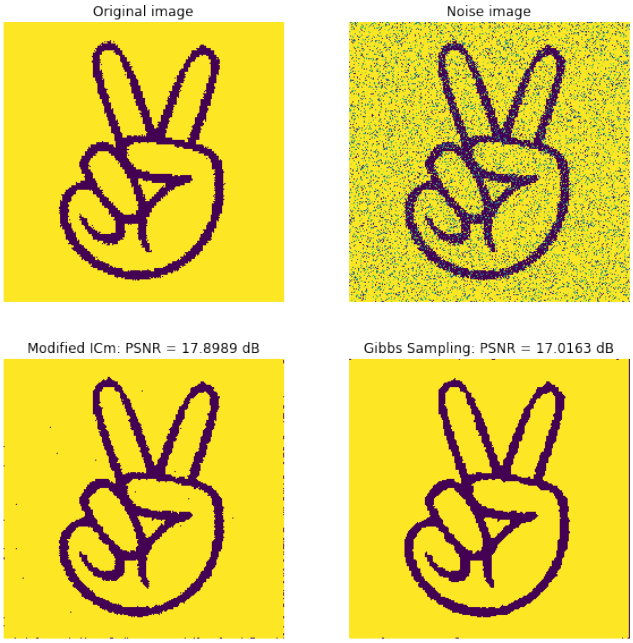

Fig. 3. The new type four plots in different conditions: The upper left is the new original true image, the upper right one is the correspond noised image. The lower right one is the de-noise image based on Gibbs Sampling, and left one is based on Modified ICM.

which contains detail of original true image in the noise image improves the performance of this model greatly.

What's more, by setting $w_2 = 0$ in this case, we could also know that the effect from the correspond $x[i,j]$ on $Y[i,j]$ could be ignored and still maintain a good performance for the result de-noised image, which means the difference between $Y[i,j]$ and its $x[i,j]$ for the MRF model based on Modified ICM in colored case is not as essential as that based Gibbs Sampling in binary case.

Continue the discussion, under knowing the difference between $Y[i,j]$ and its $x[i,j]$ is important for the MRF model based on Gibbs Sampling to perform decent in de-noising binary image, it is possible that this difference is also important for it in colored case. By changing $w_2 = 1$ from initial parameter state, we get the correspond de-noised image under Gibbs Sampling and show in Figure.6. Unfortunately, from Figure.6, the result de-noised image is still not promising, what's worse, the PSNR value as 27.7564 is decreased compare to the previous version. So, we could know that $w_2 = 0$ is not the potential reason for bad performance of MRF model based on Gibbs Sampling.

Based on the conclusion that the MRF model under Modified ICM is more suitable in this case, we pick its de-noised result and compare it with other de-noised images from other algorithms. These figures are shown in Figure.7. From Figure.7, it is obvious that the result from the MRF model under Modified ICM is best among all the de-noised images. Though the PSNR values of the de-noised images under Median Filter and Fast NI Means De-noising are higher than that of the de-noised image we get, de-noised image from the MRF model under Modified ICM still have best visual appearance and similar shapes with original true image compare to those from Median Filter and Fast NI Means De-noising in visual aspect which is essential in Image De-noising: The color distribution in the Median Filter case have huge difference with the original true image though the shapes contained in this image is similar with the original one. For another case, these are still many noise points contained in the de-noised image.

We also try different parameter settings on the MRF model under Modified ICM to see the performances of it based on different parameter settings. The result de-noised images are

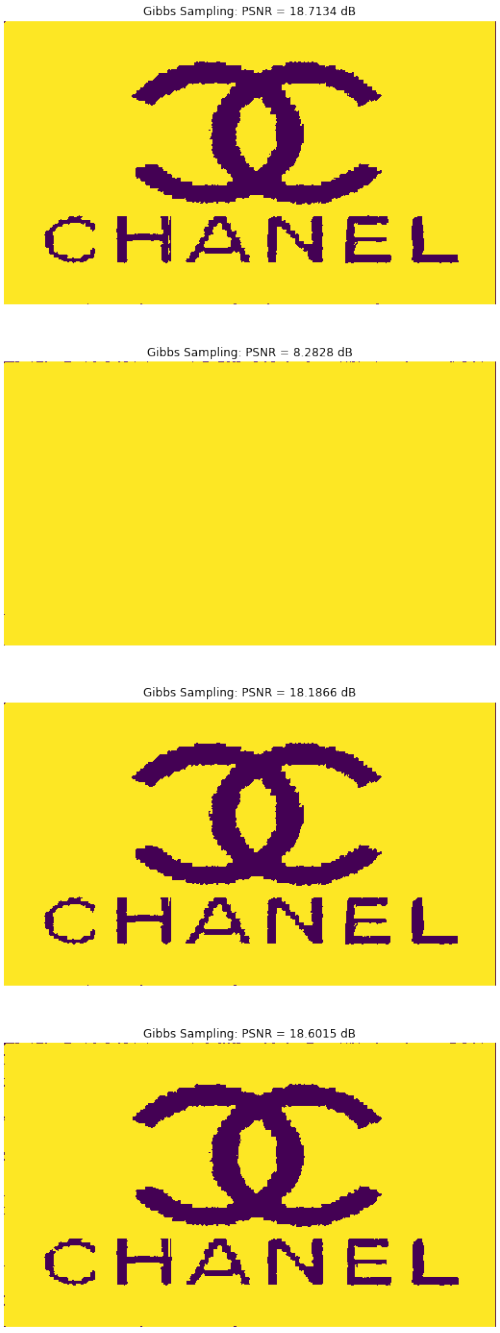

Fig. 4. The four plots in different parameter setting: The first one is under initial parameter setting. For the second one, we change w_2 to ignore the effect from $x[i,j]$ on $Y[i,j]$. For the third one, we change $\lambda_G = 20$ and $n_2 = 3$ to improve the effect from G on $Y[i,j]$. For the last one, we change $\lambda_F = 20$ and $n_3 = 4$ to improve the effect from F on $Y[i,j]$.

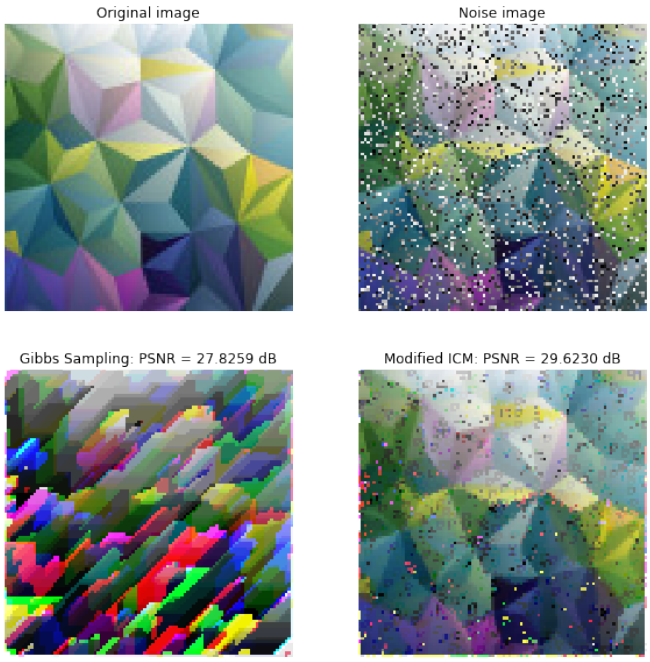

Fig. 5. The four colored plots in different conditions: The upper left is the original true image, the upper right one is the noised image. The lower left one is the de-noise image based on Gibbs Sampling, and right one is based on Modified ICM.

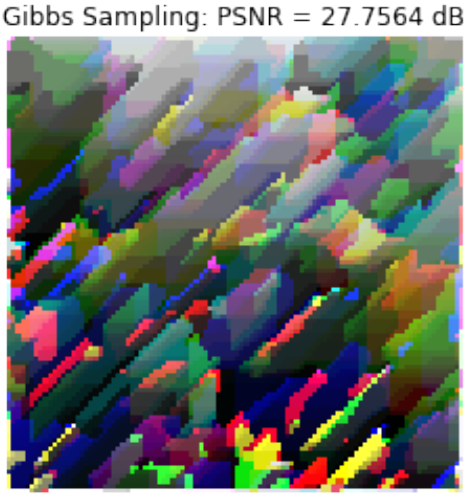

Fig. 6. The de-noised image under Gibbs Sampling with $w_2 = 1$

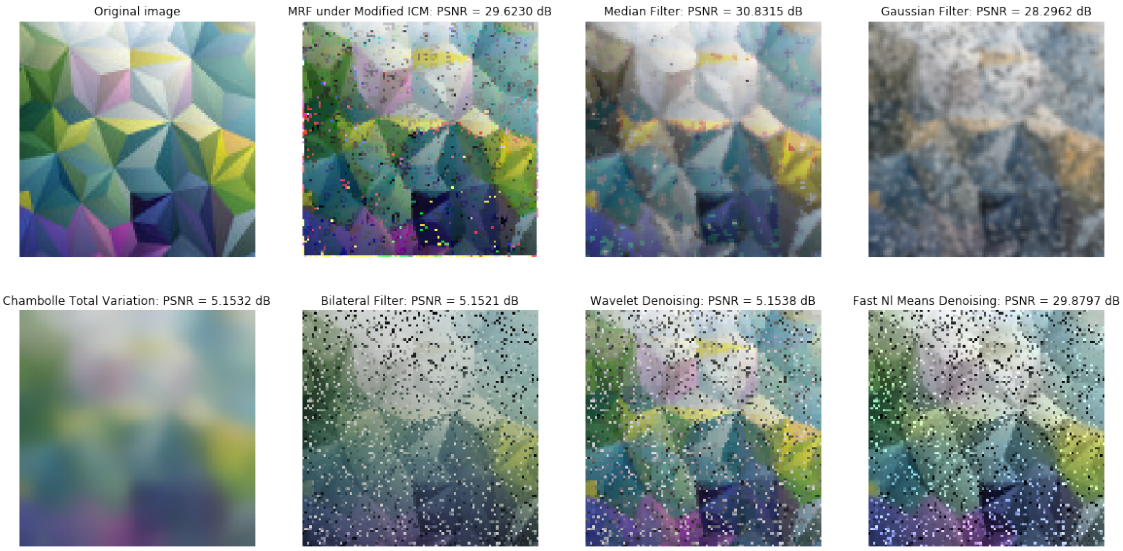

Fig. 7. The de-noised images under different algorithms

shown in Figure.8. From Figure.8, by changing $w_2 = 1$ from

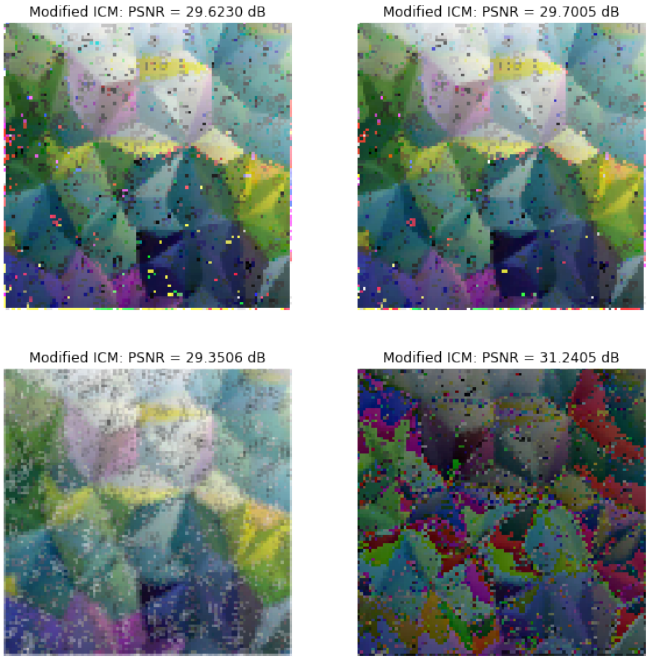

Fig. 8. The de-noised images from MRF model based on Modified ICM by feeding different parameter settings: The upper left is under initial parameter setting. For the upper right case, we set $w_2 = 1$ to reconsider the effect from $x[i, j]$ on $Y[i, j]$. For the lower left case, we change $\lambda_G = 2$ and $t_G = 2$ to improve the effect from G on $Y[i, j]$. For the lower right case, we change $\lambda_G = 5$ and $t_F = 2$ to improve the effect from F on $Y[i, j]$.

initial parameter setting, we get the result de-noised image shown in upper right, it almost be totally similar with the upper left one which is the de-noised image under initial parameters, besides the PSBR value is improved to $29,7005dB$. So, based on this found, it seems the differences between $Y[i, j]$ and the correspond $x[i, j]$ have small effects on the result de-noised image in this case, which confirms the conclusion we drawn about the MRF model based on Modified ICM above. For the lower left case, we get it by changing $\lambda_G = 2$ and $t_G = 2$ to improve the effects from G on $Y[i, j]$ under equation (2). In this case, the smoothness of whole image is improved too high to be clear due to the high weight of gradient characteristic in equation (2): The boundaries of shapes contained in noise image are obscured in the result de-noised image, the result PSNR value of the de-noised image is also decreased to $29.3506dB$. Then, for the upper right case, we get it by changing $\lambda_F = 5$ and $t_F = 2$ to improve the effects from F on $Y[i, j]$. The result image in this case are total distorted though its PSNR value is very high as $31.2405dB$ shown in Figure.8.

From the discussion above, In general, the MRF models based on Modified ICM under initial parameter we defined above or just change $w_2 = 1$ perform best compare to those under other parameter settings we tried above by combining the PSNR and visual aspect.

V. Drawbacks and Limitations

A. Limitation on Number of Iterations

In this project, we set the number of iterations for both de-noise models to 1. For binary images cases, when we apply the MRF model based on Gibbs Sampling on noise images, the pixels near the right boundaries of result de-noised images start shifting to reversal color state which is opposite to the color state they should be, as the number of iteration increases. Actually, if you observe the result de-noised images in this case from Gibbs Sampling, this phenomenon have already start when number of iteration be equal to 1: The figure on lower left in Figure.3 and four figures in Figure.4, each of them have a very tiny purple line almost attached with their right

boundaries. Interesting thing is, when we increase the number of iterations, these line tend to expand to left horizontally. Under knowing the de-noised image from Gibbs sampling could be formed in quite decent appearance and high similarity with original true image just after 1 iteration, we directly set the number of iterations to 1.

We intended to find the basic reason for the phenomenon mentioned above. Initially, we thought maybe the added outer layer with 0 for binary images cause this. However, after finding the result de-noised images from Modified ICM which also be added the same outer layer didn't have this issue, this possible reason became invalid. For now, we still not sure the actual reasons for the "expanded purple line" issue.

However, because these strange lines is quite thin compare to the entire de-noised images and could be ignored, they don't affect the general appearance and the PSNR value of de-noised images from Gibbs sampling too much. So this issue actually not affect the analyses and comparison between Gibbs Sampling and Modified ICM.

For the colored image case, the reason we set number of iterations to 1 is that the time consumed during one iteration for both de-noising models is too long. To improve the efficiency of programs running, we decide to set the number of iterations to 1 for saving time. Being similar the previous cases, we could still get the representative de-noised images from each de-noising models which could clearly describe the main characteristics for the correspond model. So, the comparison between Gibbs Sampling and Modified ICM in colored image case and the relative analyzes for each part are also not affected.

B. Boundary Issue

Being different with the boundary problem in Gibbs Sampling case mentioned in previous section, the first case of this issue is specific for binary de-noised images under Modified ICM. You could observe that the distribution of noise points in de-noised binary images under Modified ICM sometimes randomly spread among the de-noised image as shown at lower left in Figure.2, sometimes focus on the boundaries of the de-noised image as shown at lower left in Figure.3. Under knowing the outer layer added for all binary images in this project is the common part for both type binary images, it may not be the basic reason why almost all the noised points focus on the boundaries of the second type noised images. For now, the real reason still be unknown.

Another case is specific for colored case that observed from the result de-noised images. Being different with the outer layer added in binary case, which this layer is fit the binary images boundaries with same pixel values as 0, the outer layers added in colored image contain same pixel value 255 which establish white color if we plot it. So, in colored image case, the pixel values contained in these outer layer are different with those at the correspond image layers' boundaries. Then, the value of $Y[i, j]$ located at the boundaries of the result de-noised image would be affected by these "white" layers during de-noising process and be set to the values with high derivations: If you observe the de-noised images in Figure.5-8 from both models, you could find that the boundaries of each de-noised image are uneven. Because some boundary pixels are affected by the outer layers and turned into white in result de-noised images and others in the same boundary are not, this lead to the uneven boundaries for all de-noised images in colored case.

In general, the boundary pixels could be ignored compare to the entire de-noised images, also, this issue not affect the comparison between Gibbs Sampling and Modified ICM in colored image case and the relative analyzes for each part.

C. Time Consuming

Due to the characteristics and structures of the algorithms built for both de-noising models in this project, the time consumed in each iteration is very long for each model which have been mentioned in "Limitation on Number of Iterations" section. Especially for the colored case, the time period for each iteration under any model is about 10 20 minutes, this limitation restricted the efficiency of the correspond de-noised images producing.

D. The Simple Structure of F and G

To capture the detail gradient and features characteristics in the region with node $Y[i, j]$ centered, we need to consider more its neighbors in the same hidden layer and build more complex G, F energy function for describing the correspond features in this region with enough detail. After realizing these, could we balanced G, F with D in equation (2). Though changing F or G built in this paper could highly effect the result de-noised images under Modified ICM in colored case which have been shown in Figure.8, their effects on binary images cases under Gibbs Sampling would be decreased severely due to the simple structure and parameters setting for it, under knowing that sometimes their smoothness effects can't be established.

VI. Conclusion

From this Project, we built a MRF model for each image case with two-layer structure for each image layer under different sampling methods as Gibbs Sampling and Modified ICM which is defined in this project. Based on de-noising different types of noise image by using the MRF models under two sampling methods, we found that the MRF model based on Gibbs Sampling works well in de-noising binary images and the differences between $Y[i, j]$ in the hidden layer and correspond $x[i, j]$ play an important part in this model in de-noising binary images. The MRF model based on Modified ICm, which focus on all the local information in the noise image, works well in de-noising colored images. What's more, the de-noising result from this model is the best compared to other methods or algorithms chosen for this project.

References

[1] Shin, Dong-Hyuk, Rae-Hong Park, Seungjoon Yang, and Jae-Han Jung, "Block-based noise estimation using adaptive Gaussian filtering." IEEE Transactions on Consumer Electronics, vol.51, pp. 218-226, February 2005 .

[2] Haritopoulos, Michel, Hujun Yin, and Nigel M. Allinson, "Image denoising using self-organizing map-based nonlinear independent component analysis." Neural Networks, vol.15, 2002 ,pp. 1085-1098.
[3] Gopinathan, S., Radhakrishnan Kokila, and P. Thangavel, "Wavelet and FFT Based Image Denoising Using Non-Linear Filters." International Journal of Electrical & Computer Engineering, vol.5, 2015, pp.2088-8708.
[4] Xie, Junyuan, Linli Xu, and Enhong Chen. "Image denoising and inpainting with deep neural networks." Advances in neural information processing systems, 2012, pp. 341-349.
[5] Malfait, Maurits, and Dirk Roose, "Wavelet-based image denoising using a Markov random field a priori model." IEEE Transactions on image processing, vol.6, 1997, pp.: 549-565.
[6] S. G. Mallat and W. L. Hwang, Singularity detection and processing with wavelets, IEEE Trans. Inform. Theory, vol. 38, pp. 617643, March 1992.
[7] D. L. Donoho, De-noising by soft-thresholding, IEEE Trans. Information Theory, vol.41, pp.613-627, May 1995.
[8] R. Coifman and D. Donoho, "Translation invariant de-noising," in Lecture Notes in Statistics: Wavelets and Statistics, vol. New York: Springer-Verlag, pp. 125-150, 1995.
[9] Shahdoosti, Hamid Reza, and Seyede Mahya Hazavei."Image denoising in dual contourlet domain using hidden Markov tree models." Digital Signal Processing, vol.67, pp. 17-29, August 2017.
[10] Dong, Weisheng, Guangming Shi, Yi Ma, and Xin Li. "Image restoration via simultaneous sparse coding: Where structured sparsity meets gaussian scale mixture." International Journal of Computer, vol.114, pp: 217-232, September 2015.
[11] Zhang, Kai, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising." IEEE Transactions on Image Processing, vol.26, pp. 3142-3155, July 2017.
[12] Murphy, Kevin P, "Machine learning: a probabilistic perspective." MIT press, 2012.
[13] Chang Yue, "Markov Random Fields and Gibbs Sampling for Image Denoising", 2018, Unpublished. http://stanford.edu/class/ee367/Winter2018/yue_ee367_win18_report.pdf.