Model-based Iterative Restoration for Binary Document Image Compression with Dictionary Learning

Yandong Guo[1] * Cheng Lu[2] Jan P. Allebach[3] Charles A. Bouman[3]
[1]Microsoft Research [2]Sony Electronics Inc. [3]Purdue University at West Lafayette
yandong.guo@microsoft.com, cheng.lu@am.sony.com, {allebach, bouman}@purdue.edu

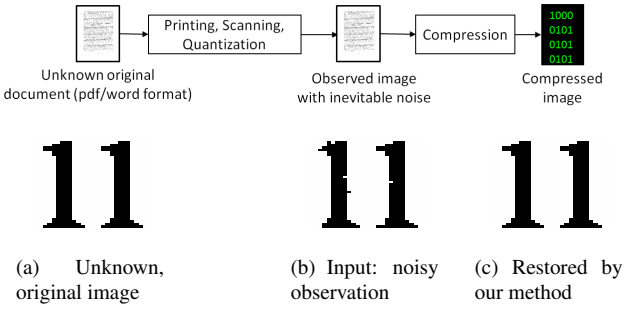
Abstract

The inherent noise in the observed (e.g., scanned) binary document image degrades the image quality and harms the compression ratio through breaking the pattern repentance and adding entropy to the document images. In this paper, we design a cost function in Bayesian framework with dictionary learning. Minimizing our cost function produces a restored image which has better quality than that of the observed noisy image, and a dictionary for representing and encoding the image. After the restoration, we use this dictionary (from the same cost function) to ***encode the restored image*** *following the symbol-dictionary framework by JBIG2 standard* ***with the lossless mode****. Experimental results with a variety of document images demonstrate that our method improves the image quality compared with the observed image, and simultaneously improves the compression ratio. For the test images with synthetic noise, our method reduces the number of flipped pixels by* 48.2% *and improves the compression ratio by* 36.36% *as compared with the best encoding methods. For the test images with real noise, our method visually improves the image quality, and outperforms the cutting-edge method by* 28.27% *in terms of the compression ratio.*

1. Introduction

To have binary document images with better quality and smaller sizes are the two goals that have been pursued for decades. The high compression ratio of document images mainly relies on the information redundancy embedded in the repeated patterns of the document image, as well as an intelligent way to leverage this pattern repentance.

Unfortunately, when the document image is obtained through scanning or other imaging devices, noise is inevitably introduced. This inherent noise breaks the pattern repentance, increases the entropy, and therefore lowers the compression ratio. As shown in Figure 1 (b), the same letter "l" look different from each other in the observed image, though they used to have the same typeface and font size in the original image in Figure 1 (a). The stroke lost its smooth boundary in the observed image. In most of the scenarios, only the observed noisy images are available. More examples are shown in Figure 4 and 7.



(a) Unknown, original image (b) Input: noisy observation (c) Restored by our method

Figure 1. The imaging and compression pipeline for document images. In the bottom area of the figure, we zoom in the document image to visualize the details of the two characters "l". As shown in subfigure (b), the input of our system contains noise inevitably introduced by using the imaging device (scanners, cameras, etc.). Our method restores the input noisy image and compresses the restored image. As shown in subfigure (c), our method successfully removes the noise and maintains/recovers the very fine details (one-pixel width stroke). Moreover, we present in Sec. 5 that encoding our restored images, compared with encoding the observed images, improves the compression ratio by 36.36% in the synthetic noise setup and 28.27% in the real noise setup.

Conventionally, there are two options to compress these observed images. In the first option, people encode the observed image as it is (lossless mode). In this case, the quality of the compressed image is equal to the quality of the observed image, while a significant portion of the bits are used to store nothing but noise. Option two is the lossy mode, which tends to have high compression ratio, but would typically make the image quality worse than the quality of the observed image input, or even introduces semantic errors to

*This research work was done when Yandong Guo and Cheng Lu were Ph.D. students at Purdue University.

the document image.

In this paper, we solve the above problem from a different perspective. We propose a restoration method to improve the quality of the observed noisy image, and simultaneously favors the compression ratio (compared with directly encoding the observed image). The intuition is that the pattern repentance of the observed noisy document image is naturally recovered during our image restoration procedure, and this pattern repentance benefits the compression ratio. Our method is summarized in two steps. First, we restore the image by minimizing a cost function (Eq. 1) in Bayesian framework. Second, after the restoration, we use the same dictionary for restoration to encode the restored document image.

Our cost function is the summation of a likelihood term and a prior term,

$$\{\hat{\mathbf{x}}, \hat{\mathbf{D}}\} = \underset{\mathbf{x},\mathbf{D}}{\operatorname{argmin}} \{-\log p(\mathbf{y}|\mathbf{x}) - \log p(\mathbf{x}|\mathbf{D}) - \log p(\mathbf{D})\}\,. \tag{1}$$

The likelihood term $-\log p(\mathbf{y}|\mathbf{x})$ is used to simulate a typical imaging pipeline (from the unknown, noise-free image $\mathbf{x}$ to the observed noisy image $\mathbf{y}$), while the prior term (the rest of the cost) is designed to encourage the image $\mathbf{x}$ to be sparsely represented by a dictionary $\mathbf{D}$ of limited size. We learn this dictionary globally from the observed noise image, and leverage the non-local information embedded in the dictionary to improve the image quality and recover the repeated patterns of the document image.

More specifically, we learn our dictionary in the conditional entropy estimation (CEE) space in [16], and leverage CEE to calculate the sparse representation cost $-\log p(\mathbf{x}|\mathbf{D})$ in the prior term. The previous art [16] demonstrates that the distribution of binary signals is better modeled in the CEE space (compared with that in the Euclidean space), and the CEE space has significant advantages in evaluating the amount of the information contained in image patches given the associated dictionary entries.

After the restoration, we first encode the dictionary $\hat{\mathbf{D}}$ estimated in the cost function in Eq. 1, and then encode the restored image $\hat{\mathbf{x}}$ using this dictionary as a reference. Our encoding follows the JBIG2 lossless encoding standard [4].

Since our sparse representation cost $-\log p(\mathbf{x}|\mathbf{D})$ is calculated by estimating the information entropy in the image given the dictionary, and we use the same dictionary for restoration and compression, our prior term in Eq. 1 has the capability of approximating the number of bits required to encode the image. Therefore, minimizing the cost function in the preprocessing step does not only improve the image quality, but also numerically reduces the approximated file size required to encode the image, with the constraint $-\log p(\mathbf{y}|\mathbf{x})$. To the best of our knowledge, this is the first time that the same dictionary is shared by restoration and compression.

We conduct experiments with test images with synthetic noise and real noise. Experimental results demonstrate that our restored image has higher quality than that of the observed image, and encoding the restored image generates higher compression ratio compared with directly encoding the input observed image.

The contribution of our paper is summarized as follows.

- We design a cost function in Eq. 1. This cost function is used to model image restoration, and also approximate the number of bits required to encode the image. Minimizing this cost function simultaneously improves the quality of the observed (e.g., scanned) document image, and improves the compression ratio.
- We learn our dictionary in the conditional entropy space, where the binary signal distribution is better modeled [16].
- To the best of our knowledge, it is the first time that the same dictionary is used for restoration and compression.
- Our bistream is compliant with the JBIG2 standard.

The paper is organized as follows. In Sec. 2, we review some of the most related work. In Sec. 3, we describe our mathematical model for both imaging and prior learning. In Sec. 4, the method to optimize our model is presented. Experimental results for the test images with synthetic and real noise are shown in Sec. 5.

2. Related works

Since we have not yet seen much effort published in optimizing restoration quality and compression ratio together, we review compression and restoration methods separately.

2.1. JBIG2 encoding

After we finish preprocessing the image with Eq. 1, we encode the **restored image** with the symbol-dictionary framework defined in the JBIG2 compression standard with the **lossless mode**, developed by the Joint Bi-level Image Experts Group [4]. The JBIG2 compression standard produces higher compression ratios than the previous standards, such as T.4, T.6, and T.82 [1, 2, 3, 22, 5], through the symbol-dictionary framework. A typical JBIG2 encoder works by first separating the document images into repeated connected components, called symbols. Then, the encoder encodes the learned dictionary entries as part of the bitstream, then encode the image using the dictionary entries as reference [15, 21, 8, 20, 34].

With the lossless mode, all the difference between the image patch and the associated dictionary entry is entropy encoded. The conventional JBIG2 lossless encoders compress the observed noisy image. In this case, the inherent

noise tends to increase entropy in the image and consumes extra bits when the document image is encoded. On the contrary, our method compress the restored image to produce better quality and higher compression ratio.

While all the conventional JBIG2 encoders compress the observed image, some encoders achieve higher compression by better dictionary learning. The dictionary learning typically consists of two critical tasks; one is to construct the dictionary, the other one is to select the best dictionary entry for a given image patch (symbol). These two tasks could be done alternatively or simultaneously.

Typically, the dictionary entry selection for a given symbol is accomplished by minimizing a measure of dissimilarity between the symbol and the dictionary entry. Dissimilarity measures widely used in JBIG2 include the Hamming distance, known as XOR [19], and weighted Hamming distance, known as WXOR [29, 14]. The weighted Hamming distance is calculated as the weighted summation of the difference between a symbol bitmap and a dictionary entry bitmap. Zhang, Danskin, and Yong have also proposed a dissimilarly measure based on cross-entropy which is implemented as WXOR with specific weights [39, 40]. The XOR has the lowest computational cost, while WXOR and cross-entropy methods are more widely used because they are more sensitive to clustered errors and can achieve lower substitution error [14, 13]. These days, to evaluate the dissimilarity between the symbol and the dictionary entry using conditional probability estimation shows great potential in [31, 16, 17]. The OCR-based method needs extensive training, and is sensitive to font and/or language type, so is beyond the discussion in this paper.

For dictionary construction, various methods have been proposed. These methods typically cluster the symbols into groups, according to a dissimilarity measure, using K-means clustering or a minimum spanning tree [34, 36, 35]. Within each group, one dictionary entry is used to represent all the symbols of that group.

Note that the JBIG2 standard also provides a lossy option. Different from the typical definition of "lossy" in JPEG or typical video coding, the lossy-JBIG2 refers to replacing the image symbols with their associated dictionary entries. The lossy option is very risky to use due to the following two types of potential quality degradation. The first one is called substitution error, which happens when the symbol is replaced by a dictionary entry with different semantic meaning. For example, the letter "c" could easily be replaced by the letter "o", especially in the low resolution scanning condition. Though many methods, including [29, 14], have been proposed to control the substitution error, we have yet to see any of them claims zero error rate. The second type of quality degradation happens when the symbol is substituted by a dictionary entry with the same semantic meaning, but lower quality. However, there has not been much effort in this field to ensure the dictionary entry has better quality than the symbols to be replaced. Due to these reasons, we do not consider the lossy mode of JBIG2 encoder in this paper.

2.2. Image restoration

The paper [26] provides a very comprehensive review from the perspective of filtering. Among all these methods, model-based reconstruction/restoration methods with a Markov random field (MRF) prior [18, 12, 6], offers very robust results. Moreover, recent methods utilizing non-local information obtain the cutting-edge performance in restoring gray/color images, *e.g.*, [42, 37, 7, 24, 9, 10, 11, 25], and promising results in various reconstruction applications, *e.g.*, [32, 41, 33, 23].

Extra work is needed to transfer these methods designed for gray image restoration to our problem. One major reason is that the distortion in binary document images has different patterns which can not be well approximated by Gaussian distribution (the implicit assumption in most of the restoration works above). The non-local information of the binary document image need to be used in a better way. Moreover, none of these above restoration methods are designed to improve the compression ratio. We solve these problems in this paper by optimizing one cost function, which simultaneously takes care of image quality and compression ratio.

3. Statistical model

Let $\mathbf{x} \in \{0,1\}^K$ denote the unknown noise-free image, the vector $\mathbf{y} \in \{0,1\}^K$ denote the observed image, we obtain the restored image to be encoded by minimizing the cost function in Eq. 1. Details of each term in Eq. 1 are presented in the following subsections.

3.1. Forward model for the likelihood term

Given the distortion-free unknown image $\mathbf{x} \in \{0,1\}^K$, the observed image $\mathbf{y} \in \{0,1\}^K$ has the following likelihood distribution,

$$p(\mathbf{y}|\mathbf{x}) = \prod_k p(y_k|\mathbf{x})\,, \tag{2}$$

where,

$$p(y_k|\mathbf{x}) = 1 - |y_k - \mu_k| \tag{3}$$

$$\boldsymbol{\mu} = A\mathbf{x}\,. \tag{4}$$

The term $|y_k - \mu_k|$ is the absolute value of $y_k - \mu_k$. In the above equations, Eq. (4) is based on the low pass assumption of printing and scanning due to the limited resolution of these procedures. We formulate this low pass filter using the matrix $A \in \Re^{K\times K}$, each row of which performs a low

pass filter to the image $\mathbf{x}$, and denote the intermediate image to be $\boldsymbol{\mu} \in [0,1]^K$. We constrain the matrix A to be sparse to achieve low computational cost, and also constrain A to be circulant to achieve homogeneous filtering to the image $\mathbf{x}$. Moreover, we propose the following constraint on each row of A to ensure there is no energy change introduced by filtering.

$$\sum_l A_{k,l} = 1\,. \tag{5}$$

Equation (3) describes the conditional probability distribution of the k^{th} pixel y_k. Since the pixel y_k has the value of either 1 or 0, we can express Eq. (3) as follows,

$$p(y_k = 1|\mu_k) = \mu_k \tag{6}$$
$$p(y_k = 0|\mu_k) = 1 - \mu_k\,. \tag{7}$$

The above Eq. (6) and (7) show that Eq. (3) is a valid probability distribution. Moreover, Eq. (6) and (7) demonstrate our intuitions to design the likelihood function: if the pixel μ_k in the intermediate image has a large value closer to 1, we have larger chance to obtain $y_k = 1$; while if the pixel μ_k has a small value closer to 0, we have larger chance to obtain $y_k = 0$.

With the two models for low pass filtering in Eq. (4) and following quantization described in Eq. (6) and (7), we establish the likelihood function in Eq. (2) based on the assumption that each of the pixels in the observed image $\mathbf{y}$ are conditionally independent distributed, given the latent image $\mathbf{x}$.

$$p(\mathbf{y}|\mathbf{x}) = \prod_k \left(1 - |y_k - \sum_l A_{k,l} x_l|\right) \tag{8}$$

Here, for both simplicity reason and the model generality, we assume that the probability distribution of the pixel y_k is only determined by the pixel value of μ_k. For a specific quantization algorithm, such as error diffusion, we can update the likelihood function accordingly.

3.2. Prior model with dictionary learning

We design the prior term in Eq. (1) as follows,

$$-\log p(\mathbf{x}|\mathbf{D}) - p(\mathbf{D}) \propto -\sum_i \log p(B_i\mathbf{x}|\mathbf{d}_{f(i)};\boldsymbol{\phi}) - \sum_j \log p(\mathbf{d}_j)\,. \tag{9}$$

In the first summation term, the term $p(B_i\mathbf{x}|\mathbf{d}_{f(i)};\boldsymbol{\phi})$ is the conditional probability of the i^{th} symbol given the $f(i)^{th}$ dictionary entry $\mathbf{d}_{f(i)} \in \mathbf{D}$, parameterized by $\boldsymbol{\phi}$. The matrix B_i is the operator used to extract the i^{th} patch (called the i^{th} *symbol*) in the image, and $j = f(i)$ denote the function that maps each individual symbol, $B_i\mathbf{x}$, to its corresponding dictionary entry, $\mathbf{d}_j \in \mathbf{D}$. For notation simplicity, we define

$$\mathbf{s}_i = B_i\mathbf{x}\,. \tag{10}$$

The second summation term is the penalizer of the dictionary size.

Our prior design has two meanings. One is for restoration: to encourage the image to be represented by a dictionary with limited size. The other one is to approximate the number of the bits required to encode the image.

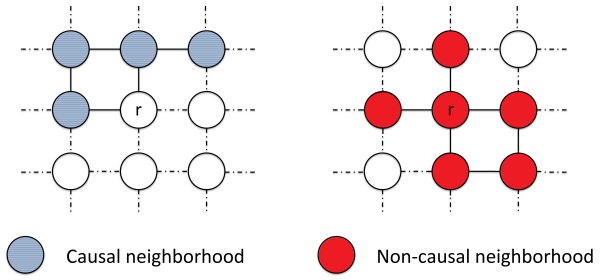
More specifically, the variable $\boldsymbol{\phi}$ is introduced to parameterize the conditional probability $p(\mathbf{s}_i|\mathbf{d}_j;\boldsymbol{\phi})$. We do not calculate Euclidean distance between the image batch and the associated dictionary entry as the log of the conditional probability because the distortion in document binary images typically does not follow the independently identically Gaussian distributed assumption well (which is the prerequisite of using Euclidean distance). Intuitively speaking, the benefit of using $\boldsymbol{\phi}$ to parameterize the conditional probability is that we can have larger weight for the rare distortion patterns, while have smaller weight for the common distortion patterns, through a rigid optimization procedure over $\boldsymbol{\phi}$. Different weights for different distortion patterns introduce a good approximation to the amount of information needed to be encoded for the symbol given the associated dictionary entry [16, 17]. This good approximation benefits the dictionary entry selection and construction, which eventually benefits the restoration and the compression. More detailed experimental results in Sec. 5 further demonstrate advantages in estimating $\boldsymbol{\phi}$ in aspects of both compression and restoration.

We briefly review how we model the conditional probability $p(\mathbf{s}_i|\mathbf{d}_j;\boldsymbol{\phi})$. The conditional probability $p(\mathbf{s}_i|\mathbf{d}_j;\boldsymbol{\phi})$ can have a very complicated form, since both $\mathbf{s}_i$ and $\mathbf{d}_j$ are high dimensional random variables. This makes the parameter vector $\boldsymbol{\phi}$ contain too many elements to be estimated. To solve this problem, we model $p(\mathbf{s}_i|\mathbf{d}_j;\boldsymbol{\phi})$ as the product of a sequence of simple probability density functions,

$$p(\mathbf{s}_i|\mathbf{d}_j;\boldsymbol{\phi}) = \prod_s p\left(s_i(r)|\mathbf{c}(\mathbf{s}_i,\mathbf{d}_j,r);\boldsymbol{\phi}\right)\,, \tag{11}$$

where the term $p\left(s_i(r)|\mathbf{c}(\mathbf{s}_i,\mathbf{d}_j,r);\boldsymbol{\phi}\right)$ is the conditional probability for the r^{th} symbol pixel $s_i(r)$ conditioned on its reference context $\mathbf{c}(\mathbf{s}_i,\mathbf{d}_j,r)$, of which the definition is shown in Fig. 2.

Figure 2 graphically illustrates one example of the structure of the reference context. As shown, the reference context $\mathbf{c}(\mathbf{s}_i,\mathbf{d}_j,r)$ is a 10-dimensional binary vector, consisting of 4 causal neighborhood pixels of $s_i(r)$ in $\mathbf{s}_i$, and 6 non-causal neighborhood pixels of $d_j(r)$ in $\mathbf{d}_j$. The decomposition in (11) is based on the assumption that, the symbol pixel $s_i(r)$, given its reference context $\mathbf{c}(\mathbf{s}_i,\mathbf{d}_j,r)$, is conditionally independent of its previous (in raster order) symbol



(a) Neighbors in symbol (b) Neighbors in dictionary entry

Figure 2. The 4 causal neighborhood pixels of $s_i(r)$ in $\mathbf{s}_i$, and the 6 non-causal neighborhood pixels of $d_j(r)$ in $\mathbf{d}_j$. Note that this is not the only neighborhood system we can use. We choose the neighborhood system which is also used in the JBIG2 standard [4], but estimate the conditional probability in a different way, as described in Sec. 4.

pixels except its 4 casual neighbors. This conditional independency design makes our decomposition different from the existing decomposition/factorization methods in inference complicated distributions [28, 27, 30].

With the decomposition in Eq. (11), we further heuristically assume that for a given document image, the natural parameter $\boldsymbol{\phi}$ in $p\left(s_i(r)|\mathbf{c}(\mathbf{s}_i,\mathbf{d}_j,r);\boldsymbol{\phi}\right)$ is completely determined by the reference context $\mathbf{c}(\mathbf{s}_i,\mathbf{d}_j,r)$. Since the symbol pixels are binary, we model their conditional distribution given a particular reference context as a Bernoulli distribution, shown as follows,

$$p(s_i(r)|\mathbf{c}(\mathbf{s}_i,\mathbf{d}_j,r);\boldsymbol{\phi}) = \phi_c^{1-s_i(r)}(1-\phi_c)^{s_i(r)}\,, \quad (12)$$

where the variable ϕ_c denotes the natural parameter of the Bernoulli distribution and fully determined by the value of the reference context vector $c = \mathbf{c}(\mathbf{s}_i,\mathbf{d}_j,r)$. In total, this reference context $\mathbf{c}(\mathbf{s}_i,\mathbf{d}_j,r)$ could possibly have 2^{10} different values with our 10 bit neighborhood system in Fig. 2, so there are 2^{10} parameters to be estimated.

$$\boldsymbol{\phi} = [\phi_1,\phi_2,\ldots,\phi_{1024}]^T \quad (13)$$

4. Optimization

With the likelihood distribution in Eq. (2),(3), and (4), and the prior distribution in Eq. (9), we obtain the cost function to be optimized as,

$$\{\hat{\mathbf{x}},\hat{\mathbf{D}},\hat{f},\hat{\boldsymbol{\phi}}\} = \underset{\mathbf{x},\mathbf{D},f,\boldsymbol{\phi}}{\operatorname{argmin}} -\sum_k \log(1-|y_k-\sum_l A_{k,l}x_l|) - \sum_i \log p(B_i\mathbf{x}|\mathbf{d}_{f(i)};\boldsymbol{\phi}) - \sum_j \log p(\mathbf{d}_j) \quad (14)$$

We propose to use an alternating optimization strategy. First, we initialize the unknown image $\mathbf{x}$ by,

$$\mathbf{x} \leftarrow \mathbf{y}\,. \quad (15)$$

MBIR_DL_Encoding($\mathbf{y}$) {

/ * Initialization * /

$\hat{\mathbf{x}} \leftarrow \mathbf{y}$

$\{\hat{\mathbf{D}}^{(0)}, \hat{f}^{(0)}\} \leftarrow$ XOR-OP $(\hat{\mathbf{x}})$

repeat

Update $\hat{\boldsymbol{\phi}}$ using (19)

Update $\hat{\mathbf{D}}, \hat{f}$ using (20)

Update $\hat{\mathbf{x}}$ using (25)

until Converge **or** Maximum number of iterations reached

Encode $\hat{\mathbf{x}}$ using JBIG2 with lossless option

return JBIG2 bitstream

}

Figure 3. Pseudocode of our method called model based iterative restoration for compression with dictionary learning (MBIR-DL-Encoding). First, as the initial step, we initialize the unknown image $\mathbf{x}$ with the observed image $\mathbf{y}$. Then, we repeat the parameter estimation, dictionary construction, and image restoration for multiple times until converge. After convergence, we encode the restored image $\hat{\mathbf{x}}$ using the JBIG2 lossless option.

Then, we update the dictionary $\mathbf{D}$, the mapping f, parameter $\boldsymbol{\phi}$, and the unknown image $\mathbf{x}$ alternatively. Overall structure of our method is listed in Fig. 3, while details are provided in the following subsections.

4.1. Dictionary learning

At the initial stage, we learn a temporary dictionary $\hat{\mathbf{D}}$ and mapping $\hat{f}$ from the current image estimation $\hat{\mathbf{x}}^{(0)}$. During the dictionary learning, we first estimate the parameter $\boldsymbol{\phi}$,

$$\hat{\boldsymbol{\phi}} = \underset{\boldsymbol{\phi}}{\operatorname{argmin}} -\sum_i \log p(B_i\hat{\mathbf{x}}|\hat{\mathbf{d}}_{\hat{f}(i)};\boldsymbol{\phi}) - \log p_{\phi}(\boldsymbol{\phi})\,, \quad (16)$$

where the term $p_{\phi}(\boldsymbol{\phi})$ is proposed to to stabilize the estimation of $\boldsymbol{\phi}$. In this distribution, we assume that all the elements in $\boldsymbol{\phi}$ are independent and identically distributed, following Beta distribution,

$$p_{\phi}(\boldsymbol{\phi}) = \prod_c \text{Beta}(\phi_c|a,b)\,, \quad (17)$$

$$\text{Beta}(\phi_c|a,b) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}\phi_c^{a-1}(1-\phi_c)^{b-1}\,. \quad (18)$$

We set $a = b = 2$.

With Eq. (11) and (12), and the prior (17) and (18), we update Eq. (16) as the following Eq. (19), which leads to

an efficient calculation of $\hat{\boldsymbol{\phi}}$.

$$\hat{\boldsymbol{\phi}} = \underset{\boldsymbol{\phi}}{\operatorname{argmax}} \left\{ \sum_{i=1}^{N} \sum_{r} [1 - \hat{s}_i(r)] \log \phi_{c(\hat{\mathbf{s}}_i, \hat{\mathbf{d}}_{\hat{f}(i)}, r)} + \sum_{i=1}^{N} \sum_{r} \hat{s}_i(r) \log \left(1 - \phi_{c(\hat{\mathbf{s}}_i, \hat{\mathbf{d}}_{\hat{f}(i)}, r)}\right) + \sum_{c} \log \phi_c (1 - \phi_c) \right\} \tag{19}$$

With the estimation of the conditional probability parameter $\hat{\boldsymbol{\phi}}$ fixed, we construct the dictionary $\hat{\mathbf{D}}$ and the mapping $\hat{f}$ using,

$$\{\hat{\mathbf{D}}, \hat{f}\} \leftarrow \underset{\mathbf{D}, f}{\operatorname{argmin}} - \sum_{i} \log p(B_i \hat{\mathbf{x}} | \mathbf{d}_{f(i)}; \hat{\boldsymbol{\phi}}) - \sum_{j} \log p(\mathbf{d}_j) \tag{20}$$

We treat this optimization as a clustering problem in entropy space, and use unsupervised greedy agglomerative clustering method to build up the dictionary and mapping.

4.2. Image restoration

In section, we present our method to restore the image with the dictionary $\hat{\mathbf{D}}$ and the mapping $\hat{f}$ fixed,

$$\hat{\mathbf{x}} \leftarrow \underset{\mathbf{x}}{\operatorname{argmin}} - \sum_{k} \log(1 - |y_k - \sum_{l} A_{k,l} x_l|) - \sum_{i} \log p(B_i \mathbf{x} | \hat{\mathbf{d}}_{\hat{f}(i)}; \hat{\boldsymbol{\phi}}) \tag{21}$$

Due to the complexity of Eq. (21), we design an iterative restoration method. At each step, we update only one pixel of the unknown image $\mathbf{x}$, and keep the rest pixels the same. We use $\tilde{\mathbf{x}}^u$ to denote the new image with the u^{th} pixel to be updated. The value change of the likelihood term (21) is simplified as,

$$\Delta_1 = -\log \frac{\prod_{\{k | A_{k,u} \neq 0\}} (1 - \|y_k - \sum_l A_{k,l} \tilde{x}_l^u\|)}{\prod_{\{k | A_{k,u} \neq 0\}} (1 - \|y_k - \sum_l A_{k,l} x_l\|)} \tag{22}$$

Note that only the rows in A of which the u^{th} element is nonzero need to be evaluated.

With the image update, the value change of the prior term is

$$\Delta_2 = -\sum_{i} \log p(B_i \tilde{\mathbf{x}}^u | \hat{\mathbf{d}}_{\hat{f}(i)}; \hat{\boldsymbol{\phi}}) + \sum_{i} \log p(B_i \mathbf{x} | \hat{\mathbf{d}}_{\hat{f}(i)}; \hat{\boldsymbol{\phi}}) , \tag{23}$$

which is efficiently calculated because only the symbol which contains the updated pixel $\tilde{x}_u$ needs to be considered. Suppose $s_{i(u)}(r)$ is the $i(u)^{th}$ symbol which contains the updated u^{th} pixel, and the changed pixel has a index r, we can rely on the decomposition in Eq. (11) to simplify Eq. (23) as,

$$\Delta_2 = \log p \left(\tilde{s}_{i(u)}(r) | \mathbf{c}(\tilde{\mathbf{s}}_{i(u)}, \hat{\mathbf{d}}_{\hat{f}(i)}, r); \hat{\boldsymbol{\phi}}\right) - \log p \left(s_{i(u)}(r) | \mathbf{c}(\mathbf{s}_{i(u)}, \hat{\mathbf{d}}_{\hat{f}(i)}, r); \hat{\boldsymbol{\phi}}\right) , \tag{24}$$

With the discussion above, we can update the u^{th} pixel as,

$$\hat{x}_u = \underset{x_u \in \{0,1\}}{\operatorname{argmin}} \Delta_1 + \Delta_2 . \tag{25}$$

As shown in Fig. 3, we repeat the parameter estimation, dictionary construction, and image restoration for multiple times until convergence, or a predefined maximum number of iterations is reached due to computing time reason. After convergence, we encode the restored image $\hat{\mathbf{x}}$ using the JBIG2 lossless option. The value of Eq. (1) is guaranteed to keep decreasing during the optimization procedure. We can not guarantee the global optimum due to a lack of convexity, but experimental results show that the local optimum we obtained is promising.

5. Experimental result

In this section, we present all the methods for comparison, and list all the parameter values we have used. We conducted experiments with both synthetic noise and real noise to evaluate the performance of our method in terms of both image quality and compression ratio.

5.1. Methods for comparison

We investigated four cutting-edge methods in our paper. All these methods follows symbol-dictionary framework in JBIG2 with lossless mode.

The first two methods encode the observed image (input) **without restoration**. The major difference between these two methods is the way they construct dictionary for encoding: one method learns the dictionary based on the weighted-XOR dissimilarity measurement (WXOR-Lossless) [29, 14], while the other method, called CEE-Lossless, learns a dictionary based on the conditional entropy estimation [16].

The other two methods encode the **restored image** estimated from the observed image. One is the method we proposed in this paper, called model-based iterative restoration with dictionary learning (MBIR-DL). In our MBIR-DL method, we fixed the matrix A in Eq. (4) as a Gaussian filter with $\sigma_r^2 = 0.2$ throughout all the experiments, and applied the JBIG2 lossless mode after the restoration.

In order to emphasize the benefits from the dictionary used in MBIR-DL, we replace the dictionary prior in our MBIR-DL with a standard Markov Random field (MRF)

Method	Restoration	Encoding Dict.
WXOR-Lossless	No	WXOR [29, 14]
CEE-Lossless	No	CEE [16]
MBIR-MRF	Yes, MRF prior	CEE [16]
MBIR-DL	Yes, dictionary prior	CEE [16]

Table 1. The methods for comparison. The first two methods (WXOR-Lossless and CEE-Lossless) encode the input observed image as it is. The other two methods encode the restored image estimated from the observed image. Our method MBIR-DL restores the observed image with a dictionary prior, while MBIR-MRF uses Markov Random field as prior. In regards of encoding, all these methods follow the symbol-dictionary framework in JBIG2 with lossless mode. The WXOR-Lossless method encodes image with a dictionary learned based on Weighted-XOR (WXOR) dissimilarity measurement. The rest three methods use the same method (conditional entropy estimation (CEE) described in [16]) to construct the dictionary for encoding.

for binary signals using the 8-pixel neighborhood system, defined in Eq. (26),

$$p(x_k) \propto \exp\left\{ - \sum_{\{l,k\} \in C} |x_k - x_l| \right\} . \qquad (26)$$

We call this method MBIR-MRF. After its restoration, MBIR-MRF encodes the restored image using the same way as MBIR-DL. These methods are summarized in Tab. 1.

5.2. Synthetic noise

We generate test images with synthetic noise so that we can evaluate the quality of the restored image with a perfectly aligned, noise-free reference image. Let $\mathbf{x}$ denote the reference image (noise free), and $\hat{\mathbf{x}}$ denote the restored image estimated from the observed noisy image, we count the total number of different pixels between $\mathbf{x}$ and $\hat{\mathbf{x}}$ as our quality metric, defined as

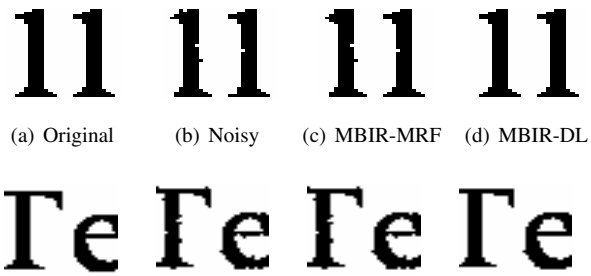
$$e = \sum_k |\hat{x}_k - x_k| \ , \qquad (27)$$

where k is the pixel index. Note that for a scanned image with inherent real noise, it is very difficult to obtain a perfectly aligned, noise free reference image (even the original document pdf is available).

5.2.1 Data generation

We obtain the noise free reference image $\mathbf{x}$ from the web. First, we downloaded pdf files of curriculum vitae of well-known professors. [1] Then, we rastered them into binary document images with the the resolution 3240×2550. Together, there are 114 binary document images containing mainly text.

In order to synthesize the noise introduced during the imaging procedure, we applied a Gaussian low-pass filter to each of the test images, which corresponds to A in Eq. (4). Note that a similar Gaussian filter is implemented in the firmware in many commercial products, such as Multi-functional printers (MFP). We followed the same noise model in Eq. (3) to generate the scanned image $\mathbf{y}$. Since different value of σ lead to different blurry levels and introduce different levels of distortions, in our experiment, we applied a 3×3 size Gaussian filter with $\sigma^2 = 0.1$, 0.12, 0.14, and 0.16 to simulate different levels of noise introduced during the imaging process. Then we obtained 4 groups of noisy images with different noisy levels.



(a) Original (b) Noisy (c) MBIR-MRF (d) MBIR-DL

(e) Original (f) Noisy (g) MBIR-MRF (h) MBIR-DL

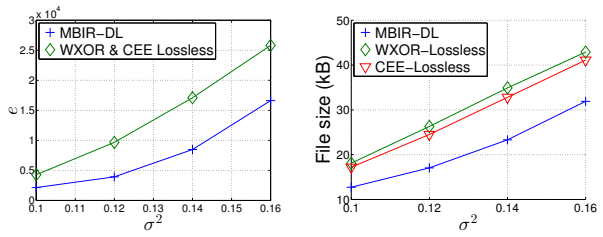
Figure 4. Visualization of the restoration results obtained by using MBIR-DL and MBIR-MRF. We re-list the example of letter "l" in Fig. 1 (a) here for the convenience of the comparison between MBIR-MRF and MBIR-DL.

5.2.2 Compare with compression without restoration

We compare our method with WXOR-Lossless in [29, 14] and CEE-Lossless in [16]. Both WXOR-Lossless and CEE-Lossless encode the observed image directly with the JBIG2 lossless mode. The quality of their compressed image is exactly the same as that of the observed image. On the contrary, our MBIR-DL method (parameter fixed) consistently improves the image quality for the test images with different noise levels, as shown in Fig. 5.

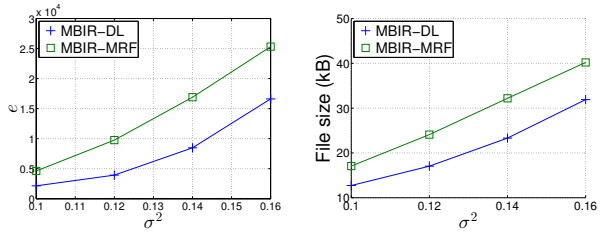
Moreover, our MBIR-DL method also consistently outperforms CEE-Lossless and WXOR-Lossless in terms of image compression ratio. This is because MBIR-DL restores the observed images and recovers the pattern repentance. Note that the CEE-Lossless method produces smaller file size compared with the file size with the WXOR-Lossless, because the dictionary learned in the conditional entropy space better represents the binary image.

[1] Due to space limit, we publish the test data and more detailed experimental results in supplementary materials.



(a) Number of error pixels e (b) File size

Figure 5. Comparison between our MBIR-DL and WXOR-Lossless, CEE-Lossless. Neither WXOR-Lossless nor CEE-Lossless change the pixel value of the input image and they have the same quality. Our MBIR-DL improves image quality and reduces the file size of the bitstream. Note that more noise (larger σ^2) generally increases file size.



(a) Error count e (b) File size

Figure 6. The MBIR-DL method outperforms MBIR-MRF in terms of both image quality and compression ratio.

5.2.3 MBIR-DL v.s. MBIR-MRF

In order to demonstrate the benefit from the dictionary used in the restoration of MBIR-DL, we compare MBIR-DL with MBIR-MRF. As described in 5.1, the only difference between the two methods is that MBIR-MRF uses Markov random field (MRF) as prior, while MBIR-DL uses the dictionary as prior.

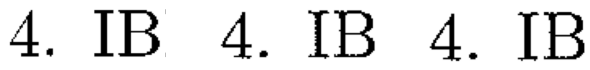
As shown in Fig. 6, our MBIR-DL methods outperforms MBIR-MRF in terms of both restoration quality and compression ratio. In Fig. 4, we visualize the restoration results comparison by zooming in the test images. Note that the subfigure (d) is a very typical case that our MBIR-DL can recover a very sharp left-corner of the left letter "l" through the usage of the non-local information. However, without non-local information usage, MBIR-MRF does not have the ability to recover this type of fine details with only one pixel wide. Also, the subfigures in the last row demonstrate that our MBIR-DL can recover images from severe distortion, though still not perfect.

5.3. Real noise

In order to evaluate the performance of our MBIR-DL method in real application scenarios, we scanned 41 binary document images. The noise is from the imaging device and more complicated than the synthetic noise. All of our test images in this subsection were scanned at 300 dpi, and have size 3275×2525 pixels. These test images contain mainly text, but some of them also contain line art, tables, and generic graphical elements, but no halftones. The text in these test images has various typefaces and font sizes.

Method	File size (KB)	Compression ratio
Lossless-TIFF	53.7 KB	19.37
XOR-Lossless	35.4 KB	29.36
CEE-Lossless	27.8 KB	37.40
MBIR-MRF	27.3 KB	38.08
MBIR-DL	21.5 KB	48.01

Table 2. Bitstream file size obtained by using different methods to the scanned test images with real noise



(a) Original image (b) Scanned image (c) MBIR-DL

Figure 7. Visualization of the restored image obtained by using MBIR-DL

As shown in Tab. 2, MBIR-DL achieves the highest compression ratio among all the competitors. Since there is no reference image, we evaluate the image quality with non-reference metrics. Using the non-reference metric specifically define for binary document images in [38], we demonstrate that the visual quality of our restored image has been improved by 5.1%. We zoomed in to sample areas in the test image for better visualization, as shown in Fig. 7. Moreover, we verified the compressed images using both tesseract-OCR and human visual check for each of the symbols in the image. No substitution error was found in the MBIR-DL compressed image.

6. Conclusion

We propose a model-based iterative restoration with dictionary learning method to solve a joint optimization regards of image quality and compression ratio. By reducing the inevitable noise introduced during the imaging process, including printing, scanning and quantization, our method simultaneously improves the image quality and compression ratio substantially, compared directly encoding the observed image input). For the test images with synthetic distortion, our method reduced the number of flipped pixels by 48.2%, improves the compression ratio by 36.36% as compared to the cutting-edge methods. For the test images with real distortion, our method outperforms the cutting-edge compression method by 28.27% in terms of the compression ratio.

References

[1] Standardization of Group 3 Facsimile Apparatus for Document Transmission. *CCITT Recommend. T.4*, 1980. 2

[2] Facsimile Coding Schemes and Coding Control Functions for Group 4 Facsimile Apparatus. *CCITT Recommend. T.6*, 1984. 2

[3] Progressive Bi-level Image Compression. *CCITT Recommend. T.82*, 1993. 2

[4] JBIG2 final draft international standard. *ISO/IEC JTC1/SC29/WG1N1545*, Dec. 1999. 2, 5

[5] R. B. Arps and T. K. Truong. Comparison of international standards for lossless still image compression. *Proc. of the IEEE*, 82:889–899, 1994. 2

[6] C. A. Bouman and K. D. Sauer. A unified approach to statistical tomography using coordinate descent optimization. *IEEE Trans. on Image Processing*, 5(3):480–492, 1996. 3

[7] A. Buades, B. Coll, and J.-M. Morel. A non-local algorithm for image denoising. In *Proc. of IEEE Computer Soc. Conf. on Computer Vision and Pattern Recognition*, pages 60–65, 2005. 3

[8] C. Constantinescu and R. Arps. Fast residue coding for lossless textual image compression. In *IEEE Data Compression Conf.(DCC)*, pages 397–406, 1997. 2

[9] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image restoration by sparse 3D transform-domain collaborative filtering. In *SPIE Electronic Imaging*, 2008. 3

[10] M. Elad and M. Aharon. Image denoising via learned dictionaries and sparse representation. In *Proc. of IEEE Computer Soc. Conf. on Computer Vision and Pattern Recognition*, pages 17–22, 2006. 3

[11] M. Elad and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Trans. on Image Processing*, 15(12):3736–3745, 2006. 3

[12] M. A. Figueiredo, J. M. Bioucas-Dias, and R. D. Nowak. Majorization-Minimization algorithms for wavelet-based image restoration. *IEEE Trans. on Image Processing*, 16(12):2980–2991, Dec. 2007. 3

[13] M. Figuera. *Memory-efficient algorithms for raster document image compression*. PhD thesis, Purdue University, West Lafayette, IN, USA, 2008. 3

[14] M. Figuera, J. Yi, and C. A. Bouman. A new approach to JBIG2 binary image compression. In *Proc. SPIE 6493, Color Imaging XII: Processing, Hardcopy, and Applications*, page 649305, 2007. 3, 6, 7

[15] O. Fumitaka, R. William, A. Ronald, and C. Corneliu. JBIG2 - the ultimate bi-level image coding standard. In *Proc. of IEEE Int'l Conf. on Image Proc.*, pages 140–143, 2000. 2

[16] Y. Guo, D. Depalov, P. Bauer, B. Bradburn, J. P. Allebach, and C. A. Bouman. Binary image compression using conditional entropy-based dictionary design and indexing. In *Proc. SPIE 8652, Color Imaging XIII: Displaying, Processing, Hardcopy, and Applications*, volume 8652, page 865208, 2013. 2, 3, 4, 6, 7

[17] Y. Guo, D. Depalov, P. Bauer, B. Bradburn, J. P. Allebach, and C. A. Bouman. Dynamic hierarchical dictionary design for multi-page binary document image compression. In *Proc. of IEEE Int'l Conf. on Image Proc.*, 2013. 3, 4

[18] E. Haneda and C. A. Bouman. Implicit priors for model-based inversion. In *Proc. of IEEE Int'l Conf. on Acoust., Speech and Sig. Proc.*, pages 3917–3920. IEEE, 2012. 3

[19] M. J. J. Holt. A fast binary template matching algorithm for document image data cmpression. In *Proc. of IEEE Int'l Conf. on Pattern Recognition*, pages 230–239, 1988. 3

[20] P. G. Howard;. Lossless and lossy compression of text images by soft pattern matching. In *1996 IEEE Data Compression Conf.(DCC)*, pages 210–219, 1996. 2

[21] P. G. Howard, F. Kossentini, B. Martins, S. Forchhammer, and W. J. Rucklidge. The emerging JBIG2 standard. *IEEE Trans. on Circuits and Systems for Video Technology*, 8:838–848, 1998. 2

[22] R. Hunter and H. Robinson. International digital facsimile coding standards. *Proc. of the IEEE*, 68:854–867, July 1980. 2

[23] P. Jin, E. Haneda, and C. Bouman. Implicit Gibbs prior models for tomographic reconstruction. In *Signals, Systems and Computers (ASILOMAR), 2012 Conference Record of the Forty Sixth Asilomar Conference on*, pages 613–616, 2012. 3

[24] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Non-local sparse models for image restoration. In *Proc. of Int'l Conf. on Computer Vision*, pages 2272–2279. IEEE, 2009. 3

[25] J. Mairal, M. Elad, and G. Sapiro. Sparse representation for color image restoration. *IEEE Trans. on Image Processing*, 17(1):53–69, 2008. 3

[26] P. Milanfar. A tour of modern image filtering: New insights and methods, both practical and theoretical. *IEEE Signal Process. Mag.*, 30(1):106–128, 2013. 3

[27] T. P. Minka. *A family of algorithms for approximate bayesian inference*. PhD thesis, 2001. AAI0803033. 5

[28] G. Parisi. *Statistical Field Theory*. Addison-Wesley, 1988. 5

[29] W. Pratt, P. Capitant, W.-H. Chen, E. Hamilton, and R. Wallis. Combined symbol matching facsimile data compression system. *Proc. of the IEEE*, 68:786–796, 1980. 3, 6, 7

[30] Y. Qi and Y. Guo. Message passing with l_1 penalized kl minimization. In *Proc. of Int'l Conf. on Machine Learning*, volume 28, Atlanta, Georgia, USA, 2013. 5

[31] Q. Qiu, V. M. Patel, and R. Chellappa. Information-theoretic dictionary learning for image classification. *CoRR*, abs/1208.3687, 2012. 3

[32] G. Wang and J. Qi. Penalized likelihood PET image reconstruction using patch-based edge-preserving regularization. *IEEE Trans. on Medical Imaging*, 31(12):2194–2204, 2012. 3

[33] Q. Xu, H. Yu, X. Mou, L. Zhang, J. Hsieh, and G. Wang. Low-dose X-ray CT reconstruction via dictionary learning. *IEEE Trans. on Medical Imaging*, 31(9):1682–1697, 2012. 3

[34] Y. Ye and P. C. Cosman. Dictionary design for text image compression with JBIG2. *IEEE Trans. on Image Processing*, 10:818–828, 2001. 2, 3

[35] Y. Ye and P. C. Cosman. Fast and memory efficient text image compression with JBIG2. *IEEE Trans. on Image Processing*, 12:944–956, 2003. 3

[36] Y. Ye, D. Schilling, P. C. Cosman, and H. H. Ko. Symbol dictionary design for the JBIG2 standard. In *IEEE Data Compression Conf.(DCC)*, pages 33–42, 2000. 3

[37] G. Yu, G. Sapiro, and S. Mallat. Solving inverse problems with piecewise linear estimators: from Gaussian mixture models to structured sparsity. *IEEE Trans. on Image Processing*, 21(5):2481–2499, 2012. 3

[38] L. Zhang, A. Veis, R. Ulichney, and J. Allebach. Binary text image file preprocessing to account for printer dot gain. In *Proc. of IEEE Int'l Conf. on Image Proc.*, 2014. 8

[39] Q. Zhang and J. M. Danskin. Entropy-based pattern matching for document image compression. In *Proc. of IEEE Int'l Conf. on Image Proc.*, pages 221–224, 1996. 3

[40] Q. Zhang, J. M. Danskin, and N. E. Young. A codebook generation algorithm for document image compression. In *IEEE Data Compression Conf.(DCC)*, pages 300–309, 1997. 3

[41] R. Zhang, C. Bouman, J.-B. Thibault, and K. Sauer. Gaussian mixture Markov random field for image denoising and reconstruction. In *Global Conference on Signal and Information Processing (GlobalSIP), 2013 IEEE*, pages 1089–1092, Dec 2013. 3

[42] D. Zoran and Y. Weiss. From learning models of natural image patches to whole image restoration. In *Proc. of Int'l Conf. on Computer Vision*, pages 479–486, 2011. 3