IMPROVED JPEG DECOMPRESSION OF DOCUMENT IMAGES BASED ON IMAGE SEGMENTATION

Tak-Shing Wong, Charles A. Bouman, and Ilya Pollak

School of Electrical and Computer Engineering
Purdue University

ABSTRACT

We propose a JPEG decompression algorithm for document images based on image segmentation. Our segmentation algorithm classifies each JPEG block of the image into one of three classes: text, background, or picture. We develop a different decoding strategy for each class and apply this strategy to every block in this class. Our experiments demonstrate that this approach can improve the quality of JPEG decompression.

1. INTRODUCTION

The JPEG algorithm is still one of the most prevalent image compression algorithms today. Further, a large number of images have been compressed with the JPEG algorithm and archived in various document and image databases. These considerations motivate research on improving the quality of JPEG decompression. We show that, for the class of document images, the quality of the decompressed image can be improved significantly by applying different decompression strategies over regions with different characteristics. We propose partitioning a document image into text, background, and picture blocks, and using different decompression algorithms for each class. Section 2 is a review of the baseline JPEG standard. Section 3 is an overview of our algorithm. Section 4 describes the segmentation algorithm we propose for classifying each image block. Section 5 introduces the three decompression strategies used for the three different classes.

2. REVIEW OF THE BASELINE JPEG STANDARD

With JPEG compression [1], an achromatic image is partitioned into 8×8 blocks. Each block then undergoes the forward discrete cosine transform (FDCT), quantization, and entropy encoding (Fig. 1). Let X_s be a column vector of the 64 pixels of the 8×8 block s. The FDCT is an orthogonal transform given by $Y_s = BX_s$, where B is a 64×64 matrix whose columns are pairwise orthogonal. The first element $Y_{s,0}$ of Y_s is called the DC coefficient of the block; the other elements are the AC coefficients. The next stage uses a quantization matrix Q, to perform element-wise uniform quantization $\tilde{Y}^Q_{s,i} = \left[\frac{Y_{s,i}}{Q_i}\right]$, where $[\cdot]$ is the integer-round operator. Finally the quantized DCT coefficients are packed into the JPEG bit-stream by entropy encoding. For a color image, this operation sequence is applied to each color channel independently. Usually the image is first converted to the YC_bC_r color space before compression.

Baseline JPEG decompression reverses the operations of compression. The JPEG compressed image is first entropy decoded. Since entropy encoding is a lossless operation, the quantized DCT coefficients $\tilde{Y}^Q_s$ are retrieved perfectly. De-quantization then estimates the original true DCT coefficients ($Y_{s,i}$) by $\tilde{Y}_{s,i} = Q_i \tilde{Y}^Q_{s,i}$. Finally, inverse discrete cosine transform (IDCT) produces the decoded block as $\tilde{X}_s = B^t \tilde{Y}_s$.

This work was supported in part by a grant from the Xerox Foundation.

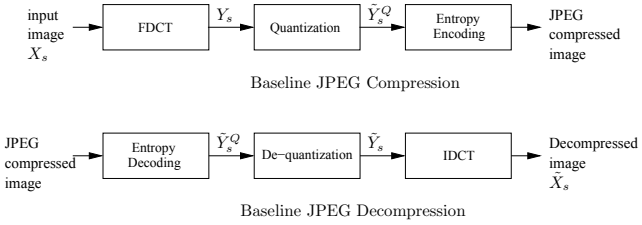


Fig. 1. Baseline JPEG compression and decompression

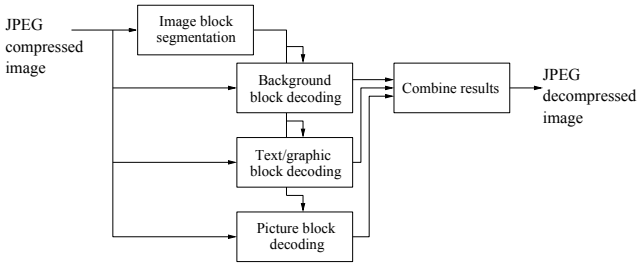


Fig. 2. Overview of the scheme

3. OVERVIEW OF THE PROPOSED ALGORITHM

Fig. 2 shows a block diagram of our scheme. The input image is first segmented using the 8×8 blocks of the image into three classes: (i) text blocks, (ii) background blocks, and (iii) picture blocks. These three classes represent image regions with very different characteristics. Accordingly, different techniques are employed for decoding blocks of each class. For a color image, the decoding stage is applied to each color channel independently.

Text blocks cover the regions of text and graphics which usually contain many sharp edges. The pixels in text blocks are typically either the background or the foreground. These blocks suffer most severely from the ringing artifacts of JPEG because they have a high level of high frequency content.

Background blocks include the background of the document and smooth regions of natural images. They have almost no high frequencies. The slow variation of the image intensity in these regions makes them very susceptible to the blocking artifacts of JPEG.

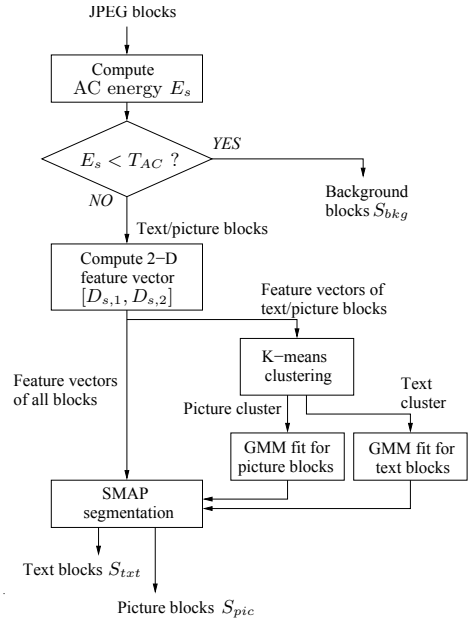


Fig. 3. Block-based Segmentation

The remaining picture blocks consist of non-smooth regions of natural images. They suffer from both the ringing and blocking artifacts. However, as noted in [2], significant high-frequency content in highly textured blocks makes the JPEG artifacts less noticeable.

4. BLOCK-BASED SEGMENTATION

To exploit the unique characteristics of each class of blocks, a block-based unsupervised segmentation algorithm (Fig. 3) is first applied to the input image to partition the set of image blocks S into the three classes: S_{bkg} for background blocks; S_{txt} for text blocks; S_{pic} for picture blocks. First, background blocks are differentiated from the other blocks by computing the AC energy in each block and thresholding. Next, a 2-D feature vector is calculated for each block using the luminance channel. The feature vectors of text and picture blocks are clustered into two groups by k-means clustering [3]. To make use of spatial regularity, each group of feature vectors is fit with a Gaussian mixture model (GMM) whose order is determined by the minimum description length criterion [4]. Finally, the two models for text and picture blocks are used to segment the feature vector image by the SMAP segmentation algorithm [5]. The result of the SMAP algorithm is combined with the background blocks detected in the AC energy thresholding stage to produce the final segmentation map.

4.1. Background Block Detection

For an achromatic image, the AC energy of the block s is defined as $E_s = \sum_{i=1}^{63} \tilde{Y}_{s,i}^2$ where $\tilde{Y}_{s,i}$ is the estimate of the i-th DCT coefficient of the block s, produced by JPEG decompression. E_s is compared with a small positive threshold T_{AC}. If $E_s < T_{AC}$, the block is classified as a background block. For a color image, the AC energy is calculated for each of the three color channels of the block s. The maximum is defined as E_s and compared with T_{AC}.

4.2. Feature Vectors

The classification of the non-background blocks into text and picture blocks is based on a 2-D feature vector computed from the luminance channel. It was reported that the code lengths of text blocks after entropy encoding tend to be longer than non-text blocks due to the higher level of high frequency content in these blocks [6],[7]. Thus the first feature is based on the encoding length and computed as [7]

$$D_{s,1} = \frac{1}{64}\left(f(\tilde{Y}_{s,0} - \tilde{Y}_{s-1,0}) + \sum_{i=1}^{63} f(\tilde{Y}_{s,i}) \right) \qquad (1)$$

$$\text{where } f(x) = \begin{cases} \log_2(|x|) + 4 & \text{if } |x| > 1 \\ 0 & \text{otherwise} \end{cases}$$

The second feature measures how close a block is to being a two-color block. For each block s, we perform a two-color projection as follows with the luminance channel decoded by conventional JPEG. The luminance values from a 16×16 window centered at the block are first clustered into two groups by k-means clustering, with means denoted by $\tilde{\theta}_{s,1}$ and $\tilde{\theta}_{s,2}$. The two-color projection is formed by clipping each luminance of each pixel to the mean of the cluster to which the luminance value belongs. This two-color projection is the best representation of the block with two colors in the MMSE sense. The ℓ^2 distance between the luminance of the block and its two-color projection is then a measure of how closely the block resembles a two-color block. We normalize this projection error by the square of the difference of the two estimated means, $|\tilde{\theta}_{s,1} - \tilde{\theta}_{s,2}|^2$, so that a high contrast block has a higher chance to be classified as a text block. The second feature is then calculated as

$$D_{s,2} = \frac{1}{|\tilde{\theta}_{s,1} - \tilde{\theta}_{s,2}|^2} \sum_{i=0}^{63} |\tilde{X}_{s,i} - \tilde{X}'_{s,i}|^2 \qquad (2)$$

where $\tilde{X}_{s,i}$ is the estimate of the i-th pixel of the block s, produced by JPEG decompression, and $\tilde{X}'_{s,i}$ is the value of the i-th pixel of the two-color projection. If $\tilde{\theta}_{s,1} = \tilde{\theta}_{s,2}$, we define $D_{s,2} = 0$.

4.3. Text and Picture Block Classification

In the last stage, the feature vectors of non-background blocks are clustered into two groups by k-means clustering. For the norm used in k-means clustering, the contributions of the two features are differently weighted. Specifically, for a vector $D^t = [D_1, D_2]$, the norm is calculated by $\|D\| = \sqrt{D_1^2 + \gamma D_2^2}$. In our experiments, we set $\gamma = 15$. The resulting cluster with a higher mean of $D_{s,1}$ and a lower mean of $D_{s,2}$ represents the class of text blocks, and the other cluster represents the picture blocks.

Each cluster is fit with a Gaussian mixture model. The two models are then employed within the SMAP segmentation algorithm [5] in order to classify each non-background image block as either a text block or a picture block.

5. IMAGE BLOCK DECODING

JPEG decoding can be posed as an inverse problem. Because JPEG quantization is a many-to-one transform, many possible image blocks X_s can produce the same de-quantized DCT coefficients $\tilde{Y}_s$ at the decoder. The non-uniqueness of the solution of reconstructing the original block X_s based on the de-quantized DCT coefficients $\tilde{Y}_s$ makes this problem ill-posed. We regularize the problem by assuming a prior model for the original image block. We then decode each block by computing the maximum *a posteriori* (MAP) estimate.

Let $T_Q(\cdot)$ be the quantization transform so that $T_Q(Y_s) = \tilde{Y}_s$. The inverse image $B^{-1}T_Q^{-1}(\tilde{Y}_s) = \{x \in \mathbb{R}^{64} : T_Q(Bx) = \tilde{Y}_s\}$ is the set of image blocks which encode to $\tilde{Y}_s$. The forward model is:

$$p_{\tilde{Y}_s|X_s}(\tilde{Y}_s|x) = \begin{cases} 1 & \text{if } x \in B^{-1}T_Q^{-1}(\tilde{Y}_s) \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

If the original image block X_s follows the prior distribution $p_{X_s}(x)$, the MAP estimate of X_s is

$$\begin{aligned} \hat{X}_s &= \arg\max_{x\in\mathbb{R}^{64}}\{p_{X_s|\tilde{Y}_s}(x|\tilde{Y}_s)\} \\ &= \arg\min_{x\in B^{-1}T_Q^{-1}(\tilde{Y}_s)}\{-\log p_{X_s}(x)\} \end{aligned} \quad (4)$$

The constraint $x \in B^{-1}T_Q^{-1}(\tilde{Y}_s)$, however, is difficult to enforce. As the DCT is an orthogonal transform, instead of $\hat{X}_s$, we will be solving for the MAP estimate of the DCT of $\hat{X}_s$ as follows:

$$\hat{Y}_s = \arg\min_{y\in T_Q^{-1}(\tilde{Y}_s)}\{-\log p_{X_s}(B^{-1}y)\} \quad (5)$$

It should be noted that the constraint now becomes the simpler pointwise constraints $|y_i - \tilde{Y}_{s,i}| \le Q_i/2$ for $i = 0 \dots 63$.

The significance of segmentation manifests itself here in that it allows us to apply a different prior model for each class of image blocks. The prior model chosen for each class captures the unique characteristics of the class and consequently produces a decoded block that is of higher quality.

5.1. Text Block Decoding

We observe that pixels in a text block typically concentrate around two values of the foreground and the background colors, and choose a prior model reflecting this. Let $\theta_{s,1}$ and $\theta_{s,2}$ be these two values of the text block s, $\theta_{s,1} < \theta_{s,2}$. The bimodal distribution of the pixels is modeled as

$$p_{\theta_s}(x_s) = \frac{1}{z}\exp\{-\frac{1}{2\sigma^2}\sum_{i=0}^{63}\min(|x_{s,i}-\theta_{s,1}|^2, |x_{s,i}-\theta_{s,2}|^2)\} \quad (6)$$

where $\theta_s = [\theta_{s,1}, \theta_{s,2}]$ are the parameters of the distribution, z is a normalization constant, and σ^2 is related to the variance of each cluster. The MAP estimates of the pixels X_s in each block s then depend on the unknown parameter vector θ_s. We use the maximum likelihood estimate of θ_s instead of its true value, to yield the following MAP estimate of X_s:

$$\hat{X}_s = \arg\min_{x_s\in B^{-1}T_Q^{-1}(\tilde{Y}_s)} \min_{\theta_s}\Big\{\frac{1}{2\sigma^2}\sum_{i=0}^{63}\min(|x_{s,i}-\theta_{s,1}|^2, |x_{s,i}-\theta_{s,2}|^2)\Big\} \quad (7)$$

Intuitively, we seek a set of values for an image block $x \in B^{-1}T_Q^{-1}(\tilde{Y}_s)$ and two colors $\theta_{s,1}$ and $\theta_{s,2}$ so that each pixel value is as close as possible to one of the two colors. While this decoding scheme produces reasonably good results, we have found that the estimation of the two colors is not accurate for some text blocks. This happens when, for example, the majority of the pixels are in the foreground, leaving very few background pixels for a good estimate of the background color. To improve the accuracy, we observe that two neighboring text blocks generally have similar foreground and background colors. Also, a background block next to a text block is usually a continuation of the background of the text.

Based on these observations, we apply spatial regularization to the estimation of the two colors of each text block. Specifically, we let $\rho(\cdot)$ be a monotone function, and penalize

$$\rho(|\theta_{r,1} - \theta_{s,1}|) + \rho(|\theta_{r,2} - \theta_{s,2}|) \quad (8)$$

for every pair $\{r, s\}$ of neighbor text blocks. (We use neighborhoods of size eight.) We use the convention that $\theta_{s,1} < \theta_{s,2}$ for all blocks, so that our penalty enforces similarity between the lighter colors of two neighbor blocks, and between the darker colors of two neighbor blocks. The specific function ρ that we use in our experiments is given by $\rho(x) = \min(x^2, T^2)$ where the threshold is T = 40. This threshold is imposed on the growth of the penalty function in order to account for the possibility that two neighboring text regions may come from different pieces of text and therefore may have very different background and/or foreground colors. In this situation, we would like to avoid imposing an excessive penalty. In addition, for every pair of neighboring blocks $\{r, s\}$ such that s is a text block and r is a background block, we assume that it is very likely that the background colors of the two blocks are similar. We estimate the color of the background block r from its estimated DC coefficient as $\mu_r = \tilde{Y}_{r,0}/8$, and penalize the difference between this color and the one of the two colors of the text block s that is closer to μ_r, by using the following penalty:

$$\min(\rho(\mu_r - \theta_{s,1}), \rho(\mu_r - \theta_{s,2})). \quad (9)$$

Combining the two new penalty terms of Eqs. (8,9) with the cost function of Eq. (7) and recasting the cost in the DCT transform domain, we obtain the following global cost function:

$$\begin{aligned} C \quad = \quad & \sum_s\sum_{i=0}^{63}\min(|(B^t y_s)_i - \theta_{s,1}|^2, |(B^t y_s)_i - \theta_{s,2}|^2) \\ & + w\sum_{r,s}\big[\rho(\theta_{r,1} - \theta_{s,1}) + \rho(\theta_{r,2} - \theta_{s,2})\big] \\ & + w\sum_{r,s}\min(\rho(\mu_r - \theta_{s,1}), \rho(\mu_r - \theta_{s,2})) \end{aligned} \quad (10)$$

where $w > 0$ is a weight factor, $(B^t y_s)_i$ is the i-th element of the intensity vector $x_s = B^t y_s$ for the block s, and

- the first summation is performed over all text blocks s;
- the second summation is performed over all the pairs of neighboring text blocks s and r;
- the third summation is performed over all the pairs of neighboring blocks s and r such that s is a text block and r is a background block.

This cost must be minimized with respect to all the DCT coefficients y_s of all the text blocks and all the parameters θ_s of all the text blocks. This minimization problem is non-convex and large, and for these reasons is difficult to solve. To compute an approximate solution, we use the iterative coordinate descent (ICD) algorithm. The ICD algorithm allows us to use a local cost function for which it is easier to construct the optimization update. Our experiments show that the resulting two-color decoding strategy decodes the text blocks at high quality.

The ICD algorithm successively optimizes the cost function with respect to one variable at a time. In our scheme, one full iteration of the ICD algorithm consists of two phases. In the first phase, called block update, the cost function is optimized with respect to the DCT coefficients of all the text blocks. In the second phase, called parameter update, the cost function is optimized with respect to the

parameters θ_s of all the text blocks. The DCT coefficients y_s of all the text blocks are initialized with the JPEG de-quantized coefficients $\tilde{Y}_s$. The parameter θ_s of a text block s is initialized with the means of the two clusters resulting from k-means clustering of $\tilde{X}_s$. The ICD iteration is repeated until the change in the cost function between two successive iterations is smaller than a predetermined threshold.

During the block update stage, the cost function is optimized with respect to one DCT coefficient at a time. The update of the k-th DCT coefficient of the text block s is guided by $\frac{\partial C}{\partial y_{s,k}}$. Depending on the sign of $\frac{\partial C}{\partial y_{s,k}}$, $y_{s,k}$ is adjusted until $\frac{\partial C}{\partial y_{s,k}} = 0$ or $y_{s,k}$ reaches either one of the end points of the constraint interval $[\tilde{Y}_{s,k} - Q_i/2, \tilde{Y}_{s,k} + Q_i/2]$.

During the parameter update stage, the cost function is optimized with respect to one parameter at a time. Similar to the block update stage, $\theta_{s,1}$ is adjusted according to the sign of $\frac{\partial C}{\partial \theta_{s,1}}$ until $\frac{\partial C}{\partial \theta_{s,1}} = 0$.

5.2. Background Block and Picture Block Decoding

Background blocks include both the background of the document and smooth regions in natural images. For these blocks, most of the information is captured by the DC coefficients. To smooth out the discontinuities across block boundaries, a continuous Gaussian Markov random field (GMRF) [8], [9] is chosen as the prior model for the DC coefficients of the background blocks.

A GMRF is specified completely by the neighborhood system and the prediction filter. In our application, we choose a 8-point neighborhood system, and a spatial invariant prediction filter h with coefficients given by $h_{s,r} = 1/6$ if r and s are vertical or horizontal neighbor blocks, and $h_{s,r} = 1/12$ if r and s are diagonal neighbor blocks. The MAP estimate of the background blocks is again computed using the ICD algorithm.

For picture blocks, we use the conventional JPEG decoding.

6. RESULTS AND CONCLUSION

Fig. 5(a) shows an image converted from PDF at 300dpi. Figs. 4(b) and 6(b) show enlarged versions of the two rectangular regions marked in Fig. 5(a). The result of applying the block-based segmentation algorithm of Section 4 to the image in Fig. 5(a) is shown in Fig. 5(b). The original image of Fig. 5(a) was compressed by baseline JPEG with default quantization matrix multiplied by 3, and decompressed by baseline JPEG. The same compressed image was also decompressed by our proposed scheme. Fig. 6 shows the results of both baseline JPEG and our scheme for a small text region. Ringing artifacts are effectively eliminated by our proposed algorithm. Note that our formulation allows the foreground and background colors to be locally adaptive, and there is no obvious color shift in the decoded image. Fig. 4 shows the results for a small region containing both background blocks and picture blocks. In the slowly varying blue sky, contouring and blocking artifacts are significantly reduced by our algorithm.

7. REFERENCES

[1] Gregory K. Wallace, "The JPEG still picture compression standard," *Commun. ACM*, vol. 34, no. 4, pp. 30–44, 1991.

[2] Thomas Meier, King N. Ngan, and Gregory Crebbin, "Reduction of blocking artifacts in image and video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 3, pp. 490–500, Apr 1999.

[3] J. McQueen, "Some methods for classification and analysis of multivariate observations," *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, pp. 281–297, 1967.

[4] C. A. Bouman, "Cluster: An unsupervised algorithm for modeling Gaussian mixtures," Available from http://www.ece.purdue.edu/~bouman, April 1997.

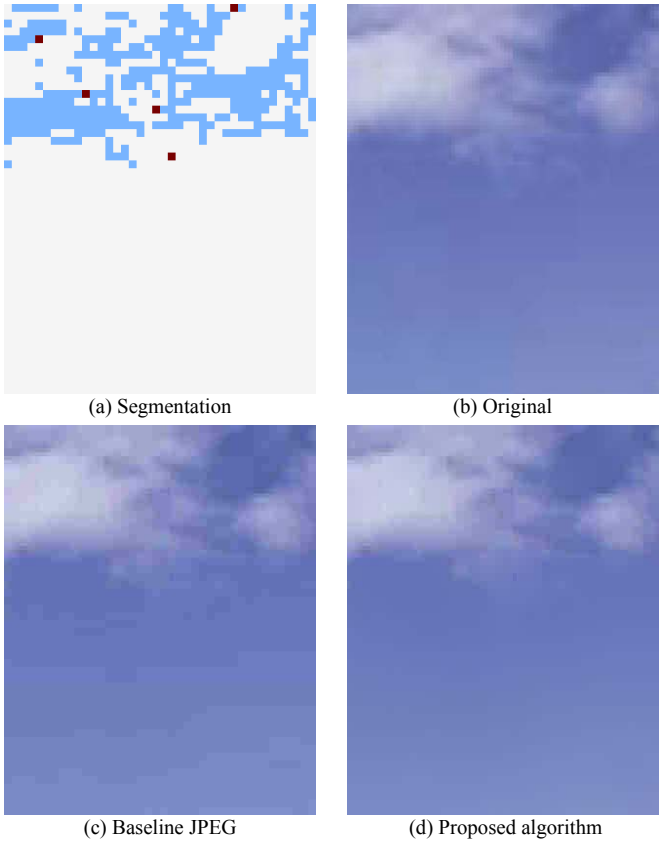
[5] Charles A. Bouman and Michael Shapiro, "A multiscale random field model for Bayesian image segmentation," *IEEE Transactions on Image Processing*, vol. 3, no. 2, pp. 162–177, Mar 1994.

[6] Ricardo L. de Queiroz, "Processing JPEG-compressed images and documents," *IEEE Transactions on Image Processing*, vol. 7, no. 12, pp. 1661–1672, Dec 1998.

[7] K. Konstantinides and D. Tretter, "A JPEG variable quantization method for compound documents," *IEEE Transactions on Image Processing*, vol. 9, no. 7, pp. 1282–1287, Jul 2000.

[8] Julian Besag, "On the statistical analysis of dirty pictures," *Journal of the Royal Statistical Society*, vol. 48, no. 3, pp. 259–302, 1986.

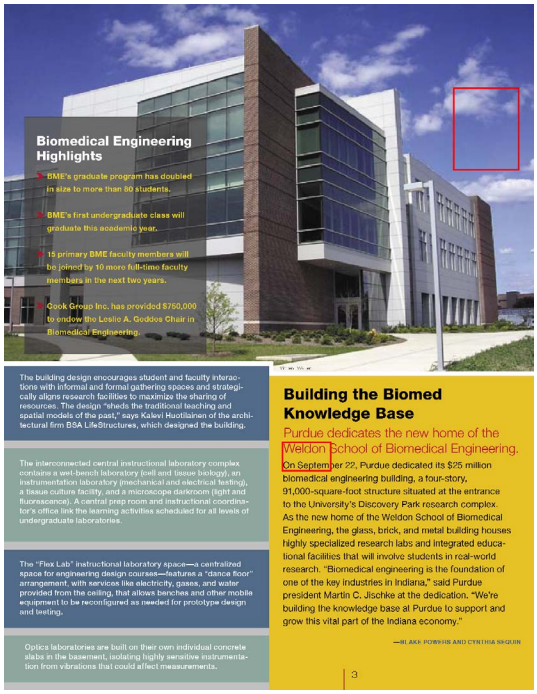
[9] Julian Besag, "Spatial interaction and the statistical analysis of lattice systems," *Journal of the Royal Statistical Society*, vol. 36, no. 2, pp. 192–236, 1974.



(a) Segmentation (b) Original

(c) Baseline JPEG (d) Proposed algorithm

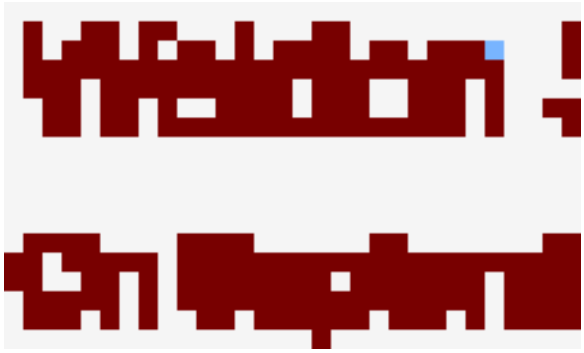
Fig. 4. Background blocks and picture blocks decoding.



(a) Original

(b) Block-based segmentation

Fig. 5. Block-based segmentation. (a) Original. (b) Segmentation. White: Background blocks; Red: Text blocks; Blue: Picture blocks.



(a) Segmentation

(b) Original

(c) Baseline JPEG

(d) Text block decoding

Fig. 6. Text block decoding result.