Financial Document Image Coding with Regions of Interest Using JPEG2000

[1, 2] Xu-Cheng Yin, [1] Chang-Ping Liu, and [1, 2] Zhi Han
[1] Character Recognition Engineering Center, Institute of Automation, Chinese Academy of Sciences, Beijing 100080, China
[2] Graduate School of Chinese Academy of Sciences, China
{xucheng.yin; changping.liu; zhi.han}@mail.ia.ac.cn

Abstract

Document image coding is a very important issue in document analysis and recognition systems provided with vast samples. An image compression algorithm with Regions Of Interest (ROIs) using JPEG2000 is proposed for financial document images which have various categories, complex layouts, and irregular noises. Three types of ROIs: filled information ROIs, seal ROIs, and handwriting ROIs, are detected and extracted through document knowledge analysis and handwriting identification. The first ROIs are detected by document classification, the second are extracted by connected component analysis based on color and shape information, and the third are located by handwriting identification using an incremental Fisher linear discriminant classifier. A ROI mask with a random shape is constructed by thresholding and merging these ROIs. Finally, a financial document image is encoded using JPEG2000 Part 1 with this ROI mask. Compared to JPEG and DjVu, the method improves visual quality while decreasing storing space.

1. Introduction

Many document image compression methods are proposed mainly for binary images, and partly for gray images: CCITT G4 (fax), JBIG1, JBIG2, and other methods [4, 6, 7]. As technology has improved, color becomes the norm for many document archives formats. Consequently, several compression algorithms are proposed for coding color documents. Most good methods for document image coding, such as JBIG2, DjVu [1], and other methods introduced in [4, 7, 11], adopt hybrid-coding schemes, which decompose a document into several regions or layers, and each of these parts is coded by different compression methods.

Forms are structured documents that can be filled in, distributed, and handled in other ways. Financial documents are a typical class of forms. In document analysis and recognition systems provided with vast samples, one important issue is how to code and store document images efficiently. Researchers have proposed several efficient methods for financial/form document image coding. Most of these methods are based on document knowledge [5, 6, 8]. However, the resulting document data are stored in specific formats, which are used only in specific systems and are not convenient for sequent print, search, retrieval, and transmission.

JPEG2000 is the latest image compression standard, which represents advances in image compression technologies. And the JPEG2000 codec allows different regions of an image to be coded with differing fidelity. This feature is known as Region-Of-Interest (ROI) coding. Particularly, the option of ROI encoding has the ability to sacrifice image quality at the uninterested region, background, and enhance image quality at the interested region, foreground, in an embedded bit stream [12]. However, no specific method is stated for ROI map in JPEG2000 coding standard. In general, the ROI is determined by object segmentation and recognition or by user's hand operation.

In Chinese, there are huge financial documents in banks, which have various categories, complex layouts, irregular noises, and color seals. Furthermore, most financial documents are needed to be stored in color images for getting a better quality. Therefore, document image compression is a very important issue. When we make use of financial documents, such as print, check, and retrieve, some information on the document is very important, and the other is less useful. The essential (important) parts include three types: (1) filled information regions, regions of filled-in data attached in pre-designed layouts, which can be machine printed or/and handwritten characters; (2) seal regions, regions of seal entities of banks, groups, companies, and persons; and (3) handwriting regions, regions of other handwritten strings located randomly which indicate corrections, additions, and other important supplemental information. In fact, these parts are ROIs of financial documents. Consequently, in document image coding, these ROIs should be coded with greater fidelity than the common regions.

In this paper, we proposed an efficient algorithm for color financial document compression with ROIs using JPEG2000, especially for Chinese financial documents. And the ROIs have a good quality in each color financial image. Unlike the former document image coding algorithms [5, 6, 8], which encode documents in their owner specific formats, our proposed method codes financial document images using a common image format standard – JPEG2000. Though the method in [13] is very similar with ours and both methods code documents with ROIs using JPEG2000, there are many differences between them. The key difference is the approach of the ROI extraction. Our proposed method includes more difficult processing steps, such as document classification, seal detection, and handwriting identification. Another difference is that the ROIs in our method can have arbitrary shapes and be disjoint, while the ROI shape in [13] is just rectangular. Finally, we utilize more document knowledge for ROI extraction based on a document model library. As a consequence, our scheme can gain a more desired compression performance.

The rest of this paper is organized as follows. In section 2, extracting ROIs of three types in financial documents is discussed. In section 3, document image coding using JPEG2000 with ROIs is presented. And In section 4, some experiments of financial document image coding are reported. Finally, several conclusions and future issues are summarized in section 5.

2. Regions of Interest Extraction

As the above explanation, the ROIs appeared on a financial document can be classified into three categories: filled information regions, seal regions, and handwritten regions. In our method, filled information regions are found in the pre-designed locations through form classification. Seal ROIs are detected based on color and shape information. The key issue of extracting handwriting regions is how to identify handwriting vs. printed text and noises. We use a Fisher classifier for handwriting identification.

Before compression processing, we construct a document model library for multiple documents. Each category has one model, which describes the line structure layout, titles, color, and filled information regions with different attributes. And a standard size of a machine-printed character for each pressed category document is defined.

2.1. Model-Based Extraction

2.1.1. Financial Document Classification

Document classification is based on matching the input document with a library of pre-defined document models to find the best match. In this article, financial documents are classified by a hierarchical method [15]. First, form classification is based on elastic matching of document structure shape [2]. Then, OCR of document titles is performed. Third, document color is re-confirmed. As a result, the sequent range of document categories becomes more and more tighter. And last, the final decision of document categories is performed by linear combination of the first two classifiers.

Let C_{graph} represent the elastic matching classifier, C_{title} be the title OCR classifier, C_{color} be the color decision classifier, and x be an input sample. Firstly, x is classified against all the N categories by C_{graph}, and there are N matching scores $S^i_{graph}(x)$, and $0 \le S^i_{graph}(x) \le 100$. Then the L categories are selected by

$$S^i_{graph}(x) > S^{threshold}_{graph} \qquad i \in \{0,1,\ldots,N-1\}$$

where $S^{threshold}_{graph}$ is a pre-decided threshold, e.g., $S^{threshold}_{graph} = 60$. Secondly, x is again classified on the current L categories by C_{title}, and there are L matching scores $S^l_{title}(x)$ too, and $0 \le S^l_{title}(x) \le 100$. Then the M categories are selected by

$$S^l_{title}(x) > S^{threshold}_{title} \qquad l \in \{0,1,\ldots,L-1\}$$

where $S^{threshold}_{titile}$ is another pre-decided threshold, e.g., $S^{threshold}_{titile} = 80$. Thirdly, x is classified on these M categories by C_{color}, and there are M matching scores $S^m_{color}(x)$, where $m = 0,1,\ldots,M-1$, and $S^m_{color}(x) = 0$ or $S^m_{color}(x) = 100$. And the K categories are selected by

$$S^m_{color}(x) = 100$$

Let x's matching scores against the current K categories be $S'^k_{graph}(x)$ and $S'^k_{title}(x)$ respectively, where $k \in \{0,1,\ldots,K-1\}$. Then these two scores are combined linearly by

$$S'_k(x) = \alpha \times S'^k_{graph}(x) + \beta \times S'^k_{title}(x)$$

where $0 < \alpha < 1$, $0 < \beta < 1$, and $\alpha + \beta = 1$. Generally, there are $\alpha = \beta = 0.5$. And last, x is classified into the k'th category by

$$S'_{k'}(x) = \max_{k=0,\ldots,K-1} S'_k(x)$$

Note that x will be rejected or belong to a new category if $S'_{k'}(x) < S'^{threshold}_{\max}$, where $S'^{threshold}_{\max}$ is a pre-threshold, e.g., $S'^{threshold}_{\max} = 70$.

2.1.2. ROI Extraction

After document classification, the exact locations of filled information ROIs in the input document x are known based on the document model of this category. Obviously, if there are K_F filled information ROIs in the model document, we will easily extract K_F ROIs from x, $\{ROI^i_F(x)\}$, where F represents this type of these ROIs is the filled information ROI, and $i = 0, ..., K_F - 1$.

2.2. Identification-Based Extraction

2.2.1. Seal Detection

Financial documents have many varied sizes and shape seals, and the seals are located randomly. Followed by some ways of text detection and time-stamp location, we propose a simple but efficient algorithm for seal detection and location.

We observe that most seals on financial documents are in red color. First, the parts in red color are reserved through color filtering. There may be many noises such as red text or red shadings. Then, the current document image is transferred into a bi-level image by Otsu thresholding method. Connected components are extracted. Based on spatial proximity and size, connected components are then merged into blocks. In a document image, there are N connected components. Let C_w, C_h, C_s are the width, height, and size of a connected component respectively. Then M connected components are selected by,

$$C_w > W_{th} \;\;\&\;\; C_h > H_{th} \;\;\&\;\; C_s > S_{th}$$

where W_{th}, H_{th}, S_{th} are the thresholds of width, height, and size respectively. And the next step is shape confirming. Most Chinese seals are rectangle, triangle, circle and ellipse in shape. And if a connected component is not in the above shapes, then it is removed. Finally, the remained connected components are again merged into much bigger blocks, which are the detected regions of seal.

2.2.2. Knowledge Based Handwriting Identification

Some work has been done on handwriting/machine printed text identification. The classification is typically performed at a text line level, a word level, or a character level [16]. In this paper, we perform a hybrid level for handwriting identification. The pattern unit of identification is a segmented block based on connected component analysis. In our method, a block can be constructed from a text line, or several characters, or only one character, or even several lines. As we know, incremental learning can gain new information from update samples and decrease the computing complexity [14]. We use an incremental Fisher Linear Discriminant (FLD) classifier to identify handwriting which incrementally updates new samples. The processed steps of our method are described in Algorithm 1.

Algorithm 1 The Handwriting Identification Algorithm

(1) Filled information ROIs are removed from the original color image.
(2) The parts of red color, such as seal regions, and red shadings, are removed, because most handwriting texts are in black or blue color.
(3) The current color document image is then transferred into a bi-level image by the Otsu binarization method.
(4) Page segmentation and zone classification are performed.
(5) Connected components are extracted, and blocks are constructed by merging connected components based on geometric relationships.
(6) Character segmentation is performed on each block. The size of each character is evaluated by the standard size of the known category knowledge. If most of the characters in a block are size-similar to the standard character, then this block is decided as the pressed block and removed.
(7) Handwriting is identified for each block by an incremental FLD classifier.

Our incremental classifier is similar to the incremental FLD classifier used in [10], though our classification is a two-class problem. The projection matrix (vector) of a classical FLD algorithm is,

$$W = S_w^{-1}(m_1 - m_2)$$

where $S_w = C_1 + C_2$ is the within-class scatter matrix, and m_i is the mean vector of each class [3]. We uses the sequential Karhunen-Loeve algorithm (SKL) introduced in [9] to incrementally update C_i. The SKL algorithm approximates C_i using the K largest eigenvalues D_i and the corresponding eigenvectors U_i of C_i,

$$C_i \approx U_i D_i U_i^T$$

where D_i is a $K \times K$ diagonal matrix and U_i is a matrix with K columns. Because the dimension of features is not much large, the SKL algorithm here directly uses Singular Value Decomposition (SVD), not R-SVD in [9], to update U_i and D_i with new samples added in.

In our incremental classifier, m_i is updated by a simple adaptive filter, $m_i^{new} = \alpha m_i + (1-\alpha)x_i$, where α is an averaging constant, typically set to a small value such as 0.05, and x_i is a new sample of the ith class.

For simple computing, features used in our incremental Fisher classifier are only "Structural", "Run-length histogram", and "Crossing count histogram" features from [16], not of each character, but of each block, without feature selection.

2.2.3. ROI Extraction

After seal detection and handwriting identification, ROI extraction of the input document x is a rather easy task. K_S seal ROIs are extracted from the original document image based on seal detection, $\{ROI_S^i(x)\}$, where S represents this type of these ROIs is the seal ROI, and $i = 0,...,K_S - 1$. In a similar way, K_H handwriting ROIs are identified and extracted from the image, $\{ROI_H^i(x)\}$, where H represents this type of these ROIs is the handwriting ROI, and $i = 0,...,K_H - 1$.

3. Image Coding Using JPEG2000 Part 1

In section 2, K_F filled information ROIs, K_S seal ROIs, and K_H handwriting ROIs are detected and extracted from the document image, which are all rectangular in shape. As a result, there are K=K_F+K_S+K_H ROIs, $\{ROI_k(x)\}$, where $k = 0,...,K - 1$. To achieve a more tight and efficient compression, ROI should have an arbitrary shape and be disjoint because the foreground (texts or seals) generally occupies a small part of a rectangular region. To do this an ROI mask is calculated,

$$M(x,y) = \begin{cases} 1 & P(x,y) \in \{ROI_k\} \ \& \ B_k(x,y) = 1 \\ 0 & otherwise \end{cases}$$

where *(x,y)* is a point in the mask, and $B_k(x,y)$ is a bi-level image pixel of the *k*th ROI color part. And the map has the same width and height as the original color document image. Every ROI part is regarded as a single image, and is binarized by Otsu thresholding. That is to say, the values of points in the map corresponding to pixels of foreground (texts or seals) in the document image are 1, and the values of points in the map to background pixels are 0.

JPEG2000 Part 1 provides two mechanisms for assigning higher priority to ROI. One mechanism is Maxshift method. The second method is to adjust the cost function which drives the PCRD-opt rate allocation algorithm [12]. For the Maxshift method, the chief disadvantage is that the scaling factor must be very large. Another problem is that the block decoder implementation must be capable of decoding a large number of magnitude bit-planes. These problems of the Maxshift method may both be avoid by using the second approach, in which the cost function used to allocate code-block contributions to quality layers is modified in accordance with the regions of interest. In our method, the second approach is adopted. And our implement of JPEG2000 with ROIs is based on the source codes of the Kakadu software, which is attached in a book (the reference [12]).

4. Experiment Results

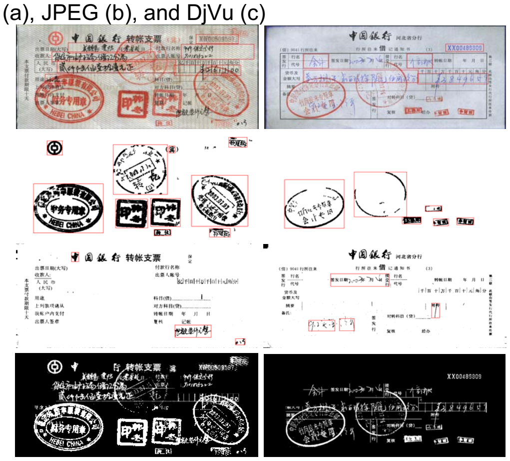
Our proposed method with ROIs using JPEG2000 is compared with DjVu, and the general JPEG. Two images used as examples in our experiments are randomly chosen from huge financial document images. Compressed images are depicted in Figure 2, where the blue rectangle is the zoomed-in part of its close bottom, and actually is a ROI region. We use the **b**it **p**er **p**ixel (bpp) of the coded document images represents the compression performance of the above three methods.

(a) Left: 0.0541 bpp; Right: 0.0189 bpp

(b) Left: 0.2412 bpp; Right: 0.1132 bpp

(c) Left: 0.0536 bpp; Right: 0.0184 bpp

Figure 2 Compressed images in JPEG2000 with ROIs (a), JPEG (b), and DjVu (c)



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

Figure 3 Extracted ROIs (indicated by red rectangles) and the ROI masks, (a) filled information ROIs, (b) seal ROIs, (c) handwriting ROIs, and (d) the ROI masks

At the filled information ROI extraction step, after document classification, several ROI areas are detected, which are shown in Figure 3(a). At the seal ROI extraction step, several separate seal ROI areas are detected and located (Figure 3(b)). And for handwriting ROI extraction, some handwritten text areas are identified, and the results are shown in Figure 3(c). Then, all the ROI areas are thresholded and merged, and the last ROI mask is given in Figure 3(d).

In the left picture of Figure 3(c), the machine-printed character “[illegible]” is identified as a handwritten character because it has a handwritten font. However, it is convenient to update our Fisher classifier with this sample for its incremental style.

From figure 2, the essential image parts (ROIs) have fairly good quality in JPEG and JPEG2000 images, but the JPEG2000 image gets a low rate which is less than a quarter of that of the JPEG image. Observed from the JPEG2000 image and the DjVu image, JPEG2000 gives better objective performance than DjVu. These compared results are more obvious in the blue rectangles (figure 2). Though DjVu has a comparatively efficient compression with JPEG2000, financial document images in DjVu are distorted seriously for some important texts, especially for seal regions. A probable reason is that financial documents have varied color seals, complex layouts, and poor qualities, and the image layer-segmentation method in DjVu is not very suitable. It is clear that JPEG2000 with ROIs codes financial document image at good image quality while maintaining a fair amount of compression.

5. Conclusions

In this article, an algorithm with ROIs using JPEG2000 is proposed for financial document image coding. ROIs of financial document are divided into three types: filled information ROIs, seal ROIs, and handwriting ROIs. Filled information ROIs are extracted by document classification, seal ROIs are detected by connected component analysis based on color and shape information, and handwriting ROIs is identified by an incremental Fisher linear discriminant classifier. A ROI mask with a random shape is constructed by binarizing and merging all these extracted ROIs. Finally, a financial document image is compressed by JPEG2000 Part 1 with this ROI mask. Compared to JPEG and DjVu, JPEG2000 with ROIs can encode financial document images with a good quality while maintaining a fair amount of compression.

A sequent work is that more experiments should be performed and compared with many different kinds of form data by measuring form classification accuracy, ROI accuracy, and also compression ratio.

In the future, another major problem is how to reduce the ROI extraction complexity, and improve the JPEG2000 encoding algorithm. Moreover, in JPEG2000 Part 6, a document image can be divided into several layers, and each layer can be coded using different algorithms, such as JPEG2000 Part 1, JPEG, and JBIG2. Document coding with JPEG2000 Part 6 is another interesting issue of future works.

References

[1] L. Bottou, et al., “High Quality Document Image Compression with DjVu”, *Journal of Electronic Imaging*, pp. 410-425, 1998.

[2] D.J. Burr, “Elastic Matching of Line Drawings”, *IEEE Trans. PAMI*, 3(6), pp. 708-713, 1981.

[3] R.O. Duda, P.E. Hart, and D.G. Stork, *Pattern Classification (2nd)*, John Wiley & Sons, Inc., 2001.

[4] O.N. Gerek, et al., “Subband Domain Coding of Binary Textual Images for Document Archiving”, *IEEE Trans. Image Processing*, 8(10), pp. 1438-1446, 1999.

[5] S. Gopisetty, et al., “Automated Forms-Processing Software and Services”, *IBM Journal of Research and Development*, 40(2), pp. 211-230, 1996.

[6] H. Joung, et al., “Document Image Compression Using Straight Line Extraction and Block Context Model”, *Proceedings of ICIP*, pp. 530-534, 1998.

[7] O.E. Kia, and D.S. Doermann, “Residual Coding in Document Image Compression”, *IEEE Trans. Image Processing*, 9(6), pp. 961-969, 2000.

[8] A.L. Koerich, and L.L. Lee, “Compression of Bank Cheque Images Based on Layout Knowledge”, *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, pp. 3185-3188, 1997.

[9] A. Levy, and M. Lindenbaum “Sequential Karhunen-Loeve Basis Extraction and its Application to Images”, *IEEE Trans. Image Processing*, 9(8), pp. 1371-1374, 2000.

[10] R.S. Lin, M.H. Yang, and S.E. Levinson, “Object Tracking Using Incremental Fisher Disrciminant Analysis”, *Proceedings of ICPR*, pp. 757-760, 2004.

[11] E. Saykol, et al., “Content-Based Retrieval of Historical Ottoman Documents Stored as Textual Images”, *IEEE Trans. Image Processing*, 13(3), pp. 314-325, 2004.

[12] D.S. Taubman, and M.W. Marcellin, *JPEG2000 Image Compression: Fundamentals, Standards and Practice*, Kluwer Academic Publishers, 2001.

[13] X.W. Yin, et al., “A Region-of-Interest Method for Texturally-Rich Document Image Coding”, *IEEE Signal Processing Letters*, 11(11), pp. 912-917, 2004.

[14] X.-C. Yin, Z. Han, and C.-P. Liu, “Selective Bagging Based Incremental Learning”, *Proceedings of International Conference on Machine Learning and Cybernetics*, pp. 2412-2417, 2004.

[15] X.-C. Yin, S.-S. Jiang, Z. Han, and C.-P. Liu, “A Hierarchical Method for Form Classification of Financial Document Images” (In Chinese), *Journal of Chinese Information Processing*, to be published, 2005.

[16] Y.F. Zheng, H.P. Li, and D. Doermann, “Machine Printed Text and Handwriting Identification in Noisy Document Images”, *IEEE Trans. PAMI*, 26(3), pp. 337-353, 2004.