Mixed Raster Content for Compound Image Compression

Final Project Presentation – EE-5359
Spring 2009

Submitted to: Dr. K.R. Rao
Submitted by: Pritesh Shah (1000555858)
MOTIVATION

- In today’s world it is impossible to imagine a day without information or information exchange in digital form over internet.

- With new advances in data processing systems and scanning devices, documents are present in a wide variety of printing systems.

- Documents in digital form are easy to store and edit, and can also be transmitted within seconds. These documents may contain text, graphics and pictures.

- So their storage requires huge memory and also transmission requires high compression and bit rates to avoid expenses and delay.
INTRODUCTION

- Compound images are documents containing both, binary text and continuous tone images.

- JPEG can be used for documents containing only pictures and graphics. But when compressing a compound document, MRC is found to have an upper hand.

- MRC uses a multi-layer, multi-resolution representation of a compound document.

- Instead of using a single algorithm, it uses multiple compression algorithms, including the ones specifically developed for text and images. So it can combine the best of new or existing algorithms and can provide various quality-compression ratio tradeoffs.
The 3-layer MRC model contains 2 colored image layers (foreground (FG) and background (BG)) and one binary image layer (mask) [1].

Mask layer is the decisive layer in reconstructing the image from the FG and BG layers.

When the pixel value in the mask layer is 1, the corresponding pixel from the FG is selected and when its 0, the corresponding pixel from the BG is selected and the final image is reconstructed as shown in Fig. 1.
MRC Framework for Scanned Data[4]

Fig. 2 MRC framework
In MRC after the original single resolution image is decomposed into layers, they are processed and compressed using different algorithms.

Fig. 2 shows how the original document X is input to the pre-processor which yields output Y (3-layer MRC model).

It also constructs an edge sharpening map and estimates the original edge “softness”. This information is termed as **side information**.

This data is then encoded and then the encoded data goes to the decoder, where the reconstructed data Y', along with the side information is used by the post-processor to assemble the reconstructed version X' of the original document.
Decomposition approaches yielding the same reconstructed image

This figure describes the decomposition process. The basic approaches are: **region classification** (RC) and **transition identification** (TI).

In case of RC decomposition, the regions containing graphics and text are represented in a separate FG plane.

Everything is represented in FG plane including the spaces in between letters and other blank spaces.
The mask as shown is uniform with large patches, clearly differentiating the text and the graphic regions and the background contains the document background, complex graphics and/or continuous tone pictures [1].

TI decomposition is quite similar to RC decomposition as can be seen from the same figure. However, in case of TI, mask and FG planes represent graphics and text in different manner.

The FG plane pours the ink of text and graphics through the mask onto the BG plane. So the mask should have the necessary text contours. Hence the mask layer contains text characters, line art and filled regions, while the FG layer contains colors of the text letters and graphics.

In RC decomposition, the mask layer is very uniform and very well compressible.
But the FG can contain edges and continuous tone details. So it cannot be compressed well with typical continuous tone coders such as JPEG.

As the mask layer in case of TI contains text objects and edges, it can be efficiently encoded using standard binary coders such as JBIG and JBIG-2.

The FG plane can be very efficiently coded even with coders such as JPEG because it contains large uniform patches. Also the FG plane can be sub-sampled without much loss in image quality [1].
RD plot modification in multiple MRC layers\textsuperscript{[1]}

Fig. 3  RD plot modification
The benefit of using MRC model for compression can be observed by analyzing its rate-distortion (RD) characteristics.

As shown in Fig. 3, if image (a) is compressed with a generic coder A with fixed parameters except for a compression parameter, it will operate under a RD plot as shown in (b). It is seen that another coder B under the same circumstances is found to perform better than coder A if its RD plot is shifted to the left as shown in (c).

The logic for MRC is to split the image into multiple planes as shown in (d), and to apply coders C, D and E to each plane with RD plots similar to that of coder B. Thus the equivalent coder may have better RD plots than A, but there would be an overhead associated with a multi-plane representation.
### List of existing MRC based encoders

<table>
<thead>
<tr>
<th>Encoder</th>
<th>Associated Company</th>
<th>Subsampling Factor</th>
<th>FG/BG Encoder</th>
<th>Mask Encoder</th>
</tr>
</thead>
<tbody>
<tr>
<td>Digipaper</td>
<td>Xerox</td>
<td>N/A 3</td>
<td>FG: Color Tags BG: JPEG</td>
<td>Token-based</td>
</tr>
<tr>
<td>DjVu</td>
<td>LizardTech, AT&amp;T</td>
<td>6 3</td>
<td>Wavelet-based</td>
<td>Modified JBIG2</td>
</tr>
<tr>
<td>LuraDocument</td>
<td>LuraTech</td>
<td>3 3</td>
<td>JPEG2000</td>
<td>CCITT G4</td>
</tr>
<tr>
<td>JPEG-matched MRC</td>
<td>HP</td>
<td>2 2</td>
<td>JPEG</td>
<td>JBIG</td>
</tr>
<tr>
<td>JPEG2000-matched MRC</td>
<td>HP</td>
<td>2 2</td>
<td>JPEG2000</td>
<td>JBIG</td>
</tr>
</tbody>
</table>
In RER, **Adaptive Error Diffusion** method is used to encode edge detail into the binary mask layer of the MRC document.

The MRC document is then decoded using a **Nonlinear Predictor** to determine the relative amount of foreground and background color to be applied to each pixel.

To yield minimum document image distortion, a method for jointly optimizing the parameters of the RER encoder and decoder is proposed.

Simulation results indicate that RER method can reduce document image distortion to a great extent at a fixed bit rate.

Also RER method is totally compatible with MRC standard and can be efficiently implemented in standard MRC encoders and decoders.
Training model of the optimized encoder and decoder
Fig. 4: Comparison of MRC (a) and RER (Resolution Enhanced Rendering) (b) Encoders$^2$
In figure 4(b), the RER encoding module creates the **dithered mask** 

\( D_s \) as the output by taking in the FG, BG, binary mask layer, and the original document as the input. The \( D_s \), FG and BG are separately compressed by using the binary image encoder and continuous-tone encoder that are used for the MRC encoder.

In the encoding module, an edge detection procedure is performed on the binary mask layer to determine the pixels that lie on the boundary between FG and BG, to compute dithered mask.

If \( K_s \) is 1, then \( s \) is a boundary pixel, else it is a non-boundary pixel. \( \lambda_s \) is obtained by Linear Projection. Then at each boundary pixel, \( D_s \) is generated from \( \lambda_s \) by switching on the error diffusion procedure. The error diffusion procedure is switched off at each non-boundary pixel and the binary mask value, \( M_s \), is the output, \( D_s \). Thus the binary output image of the dithered mask is produced.
In the previous slide, $X_s$ is the pixel color in the raster document at location $s$.

$F_s$ and $B_s$ are the decimated FG and BG colors at $s$. $M_s$ and $D_s$ are the binary and dithered masks respectively. $F_s^{-} (\tilde{F}_s)$ and $B_s^{-} (\tilde{B}_s)$ be the interpolated FG and BG colors at $s$.

$K_s$ is the binary output of the boundary detection procedure.

$\lambda_s$ is the output of the linear projector [2].

The solution of $\lambda_s$ is,

$$\lambda_s = \frac{(X_s - \tilde{B}_s)^t(\tilde{F}_s - \tilde{B}_s)}{(\tilde{F}_s - \tilde{B}_s)^t(\tilde{F}_s - \tilde{B}_s)}$$

(1)
Fig. 5: Comparison of MRC (a) and RER(b) Decoders\textsuperscript{[2]}
The figure above shows how the RER decoder works. It is assumed that the decoded FG and BG layers have lower resolution than the original. Also any interpolating algorithm can be applied to interpolate the two layers.

Let $F_s^-$ and $B_s^-$ be the decoded FG and BG layers that have been interpolated respectively, both having the same resolution as before.

$X_s^- (\hat{X}_s)$ is the reconstructed color pixel for each non-boundary pixel and is chosen between the FG and the BG colors, according to the binary mask as in the normal MRC.

To show the reconstructed pixel color if the pixel is on the boundary, a nonlinear prediction algorithm is performed [2]. It works by computing, $\lambda_s^- (\hat{\lambda}_s)$, a minimum mean-squared error (MSE) estimate of $\lambda_s$.

Using this estimate, we can have the following relation,

$$\hat{X}_s = \hat{\lambda}_s \hat{F}_s + (1 - \hat{\lambda}_s) \hat{B}_s$$  \hspace{1cm} (2)
RER predictor using tree-based nonlinear estimator[2]
RER Predictor

- A binary mask is extracted in a 5 x 5 window about the pixel in question. The binary vector formed, Zs, is then input to a binary regression tree predictor known as **tree-based resolution synthesis (TBRS)**.

- A two step process, **classification and prediction**, is used by the TBRS predictor to estimate the value of λs. Firstly it classifies the vector Zs into one of the M classes using a binary tree predictor. Each class, then has a corresponding linear prediction filter to estimate the value of λs from Zs by using the following relation,

\[
\hat{\lambda}_s = A_m z_s + \beta_m \tag{3}
\]

where, m is the determined class of vector Zs; A_m and β_m are the corresponding linear prediction parameters of class m.
Fig. 6:  Example of binary tree prediction model
Binary Tree Prediction Model

- Fig. 6 above shows the structure of the binary tree prediction model and its classification procedure.

- In (a), the tree is to be traversed down from the root which is at the top. A binary decision is to be made at each non-terminal node that the input vector \(Z_s\) passes through [2].

- The termination is at one of the leaves at the bottom, where each leaf represents a distinct class of the input vector space \(Z_s\). At any non-terminal node \(i\), the input vector is determined to go to the left child or to the right child using the specific splitting rule

\[
e_i^t (Z_s - \mu_i) \geq 0
\]

(4)

where, \(e_i\) is a pre-computed vector specifying the orientation of a hyperplane for splitting, \(e_i^t\) is the transpose of \(e_i\) and \(\mu_i\) is a reference point on the hyperplane.
If the projection of \((Z_s - \mu_i)\) is negative, then \(Z_s\) falls into the left side of the hyperplane passing through the reference point \(\mu_i\) and perpendicular to the vector \(e_i\) and consequently goes down to the left child of node \(i\). Or, it falls to the right side of the hyperplane and goes down to the right child.

Fig. 6 (b) illustrates an example of the recursive classification procedure in the input vector space \(Z_s\).

Each polygon region at the bottom level represents a different class of \(Z_s\). The distinct regions of the document corresponding to mask edges of different shapes and orientations can be separated by the classification step [2].

Thus RER leads to minimum document image distortion. Simulation results indicate that at a fixed bit rate, RER can reduce distortion to a great extent. Also RER is fully compatible with the standard MRC encoders and decoders.
MRC Decomposer$^{[6]}$
Data Filling

- There may be unused pixels in FG or BG layers because the final selected pixels can be from either layer depending on the mask.

- So in order to increase compression ratio, these pixels can be replaced by any color. This could be seen in the previous slide [4], [6]. This is the function of the pre-processor shown in the previous slide.

- Once the pre-processor and input are fixed, segmenter finds a binary mask, with which the pre-processor can derive the output layers according to the input.

- Thus, due to data filling the redundant data is replaced by another data and this helps in improving the compression to a great deal. Hence data filling improves the compression ratio.
JPEG matched MRC compression of compound documents

Schematic of MRC encoder [7]
In order to maintain tractable run-time memory requirement, the algorithm works on independent stripes of image data rather than a full image [7].

For each of the stripes there are practically ample of possible decompositions and associated stripe encoding parameters. But we need to obtain a near-optimal decomposition in terms of good compression and reconstructed image.

The analysis algorithm should analyze the input stripe thoroughly and also should consider the characteristics of the particular coders that are to be used to code the decomposed layers after the analysis.

JPEG has been chosen for coding the FG and BG layers as it is omnipresent and efficient image codec. The decomposition scheme developed is matched to the way JPEG operates. The mask layer is coded with JBIG.
Fig. 7 Images representing the magnitude of the error after decompression
(a) original compound image (600 dpi page);
(b) after JPEG compression;
(c) after MRC-JPEG compression [8]
Fig. 8 Enlarged portion of the text region of original and reconstructed images at a compression ratio of 70:1.

(a) Original,
(b) JPEG, and
(c) MRC using JPEG + MMR \textsuperscript{[1]} + JPEG.
Fig. 9: Enlarged portion of the pictorial region of original and reconstructed images at a compression ratio of 70:1.

(a) Original,  
(b) JPEG, and  
(c) MRC using JPEG + MMR$^{[12]}$ + JPEG.
Results of different coding methods applied to compound image ‘Toy Store’ [8]

<table>
<thead>
<tr>
<th>Coding method</th>
<th>Compression ratio</th>
<th>Encoding/decoding time</th>
</tr>
</thead>
<tbody>
<tr>
<td>JPEG</td>
<td>24.6:1</td>
<td>5 s / 4 s</td>
</tr>
<tr>
<td>WinZip</td>
<td>21.0:1</td>
<td>18 s / 7 s</td>
</tr>
<tr>
<td>JPEG LS</td>
<td>21.5:1</td>
<td>4 s / 7 s</td>
</tr>
<tr>
<td>MRC-JPEG</td>
<td>166.3:1</td>
<td>16 s / 11 s</td>
</tr>
</tbody>
</table>

- The figures 7, 8 and 9 show the evaluation of the visual quality by showing the error after decompression [8].
- These results correspond to the compression ratios in Table above.
- It can be seen from the figure that JPEG produces large errors around the text whereas MRC decomposition and coding can improve both the compression ratios and quality to a great extent.
SPEC System
(SHAPE PRIMITIVE EXTRACTION AND CODING)

- In case of a compound image compression, most of the algorithms use the standard 3-layer MRC model.

- But this model is computationally expensive and has some image-processing overhead to manage multiple algorithms and the imaging model.

- SPEC [10] decomposes the image into two sets of pixels: text/graphics and pictures.

- Its unique features are that segmentation and coding are tightly integrated in the SPEC algorithm.

- Also shape and color serve as two basic features to the segmentation and the lossless coding.

- SPEC also provides a lossless coding scheme with low complexity algorithm for segmentation and good compression ratio along with excellent visual quality.
[Flow Chart of the SPEC system][10]
SPEC - Segmentation

Flow-chart of Segmentation [10]
The two steps that comprise segmentation are block classification and refinement segmentation.

In step one, the 16 x 16 non-overlapping blocks are scanned to count the number of different colors.

The block is classified as the picture block if the color number is larger than a certain threshold value $T_1$ ($T_1=32$ is used for SPEC) \[11\]. Else it is classified as text/graphics block. This classification can be extremely fast.

Thus the segmentation process segments the image into two parts – text/graphics pixels (shape primitive pixels and all pixels of text/graphics) and pictorial pixels (the remaining pixels in the picture blocks).
Fig. 10: Flow-chart of Coding [10]
The SPEC coding process is shown in Fig. 10. The pictorial pixels are compressed with the lossy JPEG algorithm, while the text/graphics pixels are compressed with a new lossless coding algorithm.

In lossless coding, the decision of shape-based or palette-based coding depends on the complexity of the block and the attainment of the minimal code length condition.

Text/graphics pixels are removed from the pictorial block before the lossy JPEG coding in order to improve the compression ratio.

On the decoding side, first the JPEG coded pictorial pixels are decoded as BG and then the text/graphics pixels are decoded as FG.
References


## Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BG</td>
<td>Background</td>
</tr>
<tr>
<td>CCITT, ITU-T</td>
<td>Telecommunication Standardization Sector of the International Telecommunications Union</td>
</tr>
<tr>
<td>Dpi</td>
<td>Dots per inch</td>
</tr>
<tr>
<td>FG</td>
<td>Foreground</td>
</tr>
<tr>
<td>HP</td>
<td>Hewlett-Packard</td>
</tr>
<tr>
<td>ICIP</td>
<td>International Conference on Image Processing</td>
</tr>
<tr>
<td>ITU-T</td>
<td>Telecommunication Standardization Sector (ITU-T) coordinates telecommunication standards on behalf of ITU (International Telecommunication Unit)</td>
</tr>
<tr>
<td>JBIG</td>
<td>Joint bi-level image experts group</td>
</tr>
<tr>
<td>JPEG</td>
<td>Joint Photographic Experts Group</td>
</tr>
<tr>
<td>M</td>
<td>Mask</td>
</tr>
<tr>
<td>MMR</td>
<td>Modified Modified read</td>
</tr>
<tr>
<td>MRC</td>
<td>Mixed raster content</td>
</tr>
<tr>
<td>MSE</td>
<td>Mean square error</td>
</tr>
<tr>
<td>RC</td>
<td>Region classification</td>
</tr>
<tr>
<td>RD</td>
<td>Rate distortion</td>
</tr>
<tr>
<td>RER</td>
<td>Resolution enhanced rendering</td>
</tr>
<tr>
<td>SPEC</td>
<td>Shape primitive extraction and coding</td>
</tr>
<tr>
<td>TBRS</td>
<td>Tree-based resolution synthesis</td>
</tr>
<tr>
<td>TI</td>
<td>Transition identification</td>
</tr>
<tr>
<td>TIFF-FX</td>
<td>Tagged image file format fax extended</td>
</tr>
</tbody>
</table>
Thanks