Block based Method for Real-Time Compound Video Compression

Bing Han[a], Dapeng Wu[a], and Huipin Zhang[b]

[a]Department of Electrical and Computer Engineering
University of Florida Gainesville, FL 32611
[b]Cisco Systems, Santa Clara, CA 95054
Correspondence author: Prof. Dapeng Wu, wu@ece.ufl.edu, http://www.wu.ece.ufl.edu

ABSTRACT

In this paper, we propose a novel compound video compression method for real-time applications of computer screen video transmission. The compound video is a very special kind of video, which usually has less motion and contains a mixture of text, graphics, and natural pictures in each frame. A variety of algorithms has been proposed for compound image compression, which tries to remove the spatial redundancy of a single image. However, few works have addressed the problem of compound videos compression. Therefore, we review the previous works on compound image compression and discuss how to extend the existing algorithms to compound video compression. We propose a new video compression framework based on H.264. In order to improve the visual performance and network efficiency, we propose an adaptive quantization algorithm and a rate control algorithm for compound video compression. The experimental results show that the new compression method is able to compress the compound video in real-time with a relatively low computational complexity and with the motion compensation, the visual quality of the compound video is much better than simple compound image compression.

Keywords: Compound video compression, real-time transmission, adaptive quantization, rate control algorithm

1. INTRODUCTION

With the fast development of computer communication and networks, more and more applications require real-time transmission of computer screen data, such as web conferencing, remote computer access, and wireless computer projector. For example, web conferencing is to conduct live meetings or presentations using computers over the internet. Usually, the presenter needs to share his or her computer desktop or a specific application to the other participants in real time. Generally, the computer screen data is known as compound image or compound video. The raw compound data is usually too large for transmission over the internet. For example, a computer with 1024 by 728 screen resolution and 75 Hertz screen refresh rate can generate more than 100 MB data each second. Therefore, compression of the compound data is required for real-time transmission in current computer networks.

Previous researches focus on compound image compression instead of compound video compression. In other words, they study how to remove the spatial redundancy in a single compound image. Since the compound image is a mixture of text, graphics, and natural images, it could be better compressed if the prior knowledge is known. With the prior information, different compression algorithms are applied to different types of images. However, in the current computer operation systems, we can hardly get this prior information. Therefore, the most intuitive approach is first segmenting the text, graphics, and natural images and then compressing them separately with different compression algorithms. A variety of methods has been proposed based on this approach /cite.

Since most previous compression methods take the compound document as images, the temporal redundancy of the compound image sequence has not been well studied. In this paper, we will compress the compound data as a video sequence. With intra and inter prediction and motion compensation, it is known that video compression is much more efficient than static image compression when compressing a continuous image sequence. However, simply apply current video compression algorithm to compound video sequence will introduce problems... In order to improve the visual performance and achieve constant rate flow, We propose an adaptive quantization algorithm and a rate control algorithm specifically for compound video compression. Our proposed method is based on H.264, which is the state of art in video compression.

This paper is organized as follows. In Section 2, we review several previous works in compound image compression and describe the framework of the proposed compound video compression system. In Section 3, we discuss the compound imaging model and proposes an adaptive quantization algorithm and a rate control algorithm to improve the visual quality and network efficiency, respectively. The experimental results are shown in Section 4 and Section 5 concludes the paper.

2. COMPOUND VIDEO COMPRESSION

In this section, we will review some previous works on compound image compression, and then propose a new compound video compression method and present the system scheme.

2.1 Compound Image Compression

The compound image compression has been widely studied over the last decade. Because the sensitivity of human vision to natural image and text is different, the quality requirement of compound image compression is different from natural image compression. The previous works of compound video compression can be classified into two categories, layer-based methods and block-based methods.

Most layer-based approaches use Mixed Raster Content (MRC)[1] imaging model for image representation. MRC is a very common model used for scanned document images and it enables a multi-layer multi-resolution representation of a compound image. The basic MRC model represents a color image as two color-image layers, Foreground (FG) and Background (BG), and a binary Mask layer. The Mask describes how to reconstruct the final image from the FG/BG layers. DjVu[2,3] uses a wavelet based coder for the FG and BG layers, and JB2 for the mask. R. L. de Queiroz et al.[4] present a compound image codec that uses the H.264/AVC intra mode codec to encode the FG and BG layers and shows H.264 intra mode codec is much better than JPEG2000.

Block-based approaches are also studied for its low computational complexity, which is more suitable for real time applications. Li and Lei[5,6] propose a histogram based classification algorithm which classify each 16 by 16 image block into four types and then apply different compression algorithms to different types respectively. Ding et at.[7] propose a fast block classification algorithm for compound image coding. For each block type, they design an algorithm according to their different statistical properties to maximize the compression performance. Said and Drukarev[8] propose a classification algorithm working on object boundaries which is fast and require very low memory. Lin and Hao[9] propose a shape primitive extraction and coding based algorithm which segment the image into two type of blocks, text/graphic block and picture block. All shape primitives from both block types are lossless compressed by using a combined shape based and palette based coding algorithm.

2.2 System Scheme

The layer-based approaches use a mask of the whole image to represent the FG/BG layers and it is more accurate comparing to the block-based approaches, which use local statistical properties to classify the image blocks into several groups. However, the block-based algorithms run faster and use less memory, which is more suitable for real-time applications. In this paper, we try to combine these two approaches and use it in the block-based video compression method, such as H.264. The advantage of using video coding in compound document compression is that the temporal redundancy is already well studied in video compression, so it is much more efficient than simple still image compression.

Below is the framework of our proposed compound video compression scheme. In our compression system, there is a preprocessing before video compression. In the current video encoding algorithms, there is little preprocessing due to high computational complexity. However, in compound video compression, prior information is required to remove both the space and temporal redundancy. The preprocessing includes two steps. Firstly, we segment the image into two layers, the text/graphic layer and picture layer, which is represented by a binary mask. We also use a fast detection algorithm to find out the changed area in the current frame comparing to the previous frame and use another binary mask to indicate the updated area. However, unlike the layer-based approaches, we do not need to transmit the mask, which is an overhead, but only use it for adaptive quantization later in video compression. Besides, the traditional rate control algorithm in H.264 is not suitable for compound video compression, because there is less regular object motion and most are scene changes. To attack this problem, we propose a new rate control algorithm for compound video compression based on the binary mask of updated regions.

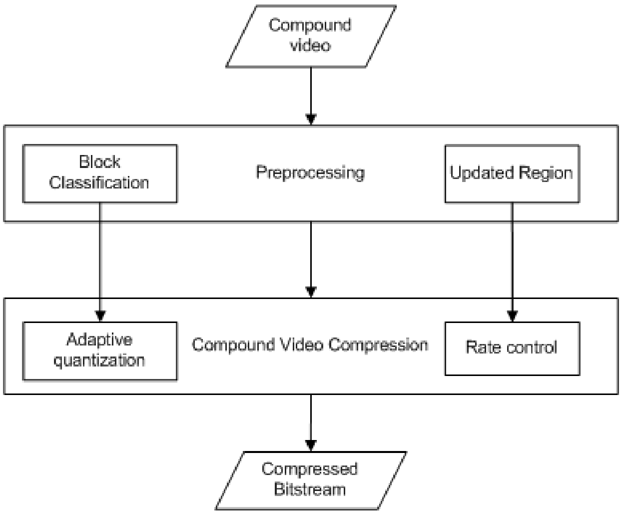


Figure 1. Compound video compression system flow chart

3. H.264 BASED COMPRESSION ALGORITHM

In this section, we will present the proposed algorithms used in H.264 for compound video compression in details. The preprocessing is to generate two binary masks, one is to separate the frame into two regions, text/graphic and natural picture, the other mask is to indicate the updated region in the current frame comparing to the previous frame. We further propose an adaptive quantization algorithm and a rate control algorithm based on the prior information we obtained from preprocessing.

3.1 Image Preprocessing

Since we need to generate both the text mask and update mask for each frame in the compound video sequence, the segmentation algorithms need to have low computational complexity.

To segment the text/graphic and natural picture, we simply count the number of pixels for each color in the frame and then classify by thresholding the number of pixels with the same color. The whole frame is scanned to count the number of different colors and the number of pixels for each color. If the pixel number of one color is more than a certain threshold λ, this color is marked as text/graphic color and all pixels of this color are classified as text/graphic pixels. The reason is intuitive that nature pictures usually have a continuous tone, which means it contains many different colors. While for text and graphics, they tend to use the same color for large areas. The color-counting algorithm is simple and fast, therefore, it is suitable for video compression.

The most intuitive method for finding out the updated region is to compare the two consecutive frames pixel by pixel. However, the computational cost of this algorithm is too high. In order to lower the computational complexity, we first divide the frame into 128 by 128 blocks. Then we scan each block and find the Message-Digest algorithm 5 (MD5) checksum of each block by using MD5 hash function. MD5[10] is widely used cryptographic hash function and it is commonly used to check the integrity of files. Any update will be reflected in their MD5 checksum. Therefore, for each block, we only need to compare the checksum of the two blocks instead of comparing them pixel by pixel. In this way, we can easily find out the update region in blocks.

3.2 Adaptive Quantization

In order to improve the visual performance, we exploit the characteristics of both human visual system and compound video signals. Because human vision is more sensitive to text/graphic, previous block-based compound image compression algorithms usually apply lossless coding algorithm to text/graphic blocks while applying lossy coding algorithm to picture blocks. In video compression, we do not need such kind of classification, instead, we can set coding priority by adaptively change the QP for each MB, based on the prior information we get from preprocessing. The study of HVS also shows that the HVS is more sensitive to low motion areas. Based on the above observations, we modulate the pixel level quantization step size for an MB with separate modulation factors respectively determined by the motion and the text/graphic segmentation result.

Explicitly,

$$Q_{MB} = Q_{Pic} f_{text} f_{static} \tag{1}$$

where Q_{MB} and Q_Pic are quantization step sizes of the MB and the picture respectively, f_{text} and f_{static} are modulation factors corresponding to text/graphic segmentation and region update respectively.

We use the following method to compute qp_{MB} for each MB.

$$f_{text} = \frac{M_{text} + \alpha_t \overline{M}_{text}}{\alpha_t M_{text} + \overline{M}_{text}} \tag{2}$$

$$f_{static} = \frac{M_{static} + \alpha_s \overline{M}_{static}}{\alpha_t M_{static} + \overline{M}_{static}} \tag{3}$$

where M_{text} and M_{static} are text index and static index of the MB respectively, $\overline{M}_{text}$ and $\overline{M}_{static}$ are average text index and average static index of the whole picture respectively, and α_t and α_s are the parameters that determine the two linear fractional models.

The text index and static index are computed respectively from the text mask and static mask obtained from preprocessing. The text index of the MB is sum of the text/graphic binary mask of the MB, and the static index is computed in the same way. The model parameter α_t and α_s take continuous value all in the range $[0, 1]$, representing the modulation factors f_{text} and f_{static} in the range of $(\alpha_t, \frac{1}{\alpha_t})$ and $(\alpha_s, \frac{1}{\alpha_s})$ respectively. As video coding parameters, they are already optimized in Zhang and Cote's work.[11] In this paper, we take the average value 0.5 for both α_t and α_s.

So quantization parameter of an MB qp_{MB} is given by

$$qp_{MB} = qp_{PIC} + round(6 \log_2(f_{text} f_{static})) \tag{4}$$

where qp_PIC represents the quantization parameter of the picture.

3.3 Rate Control

Most block-based hybrid video encoding schemes, such as H.264, are inherently lossy process. Rate control algorithm[12] dynamically adjusts encoder parameters in video compression, such as H.264, to achieve a target bit rate. The quantization parameter qp regulates how much spatial detail will be saved. When qp is very small, almost all the detail is retained.

The heart of the rate control algorithm is a quantitative model describing Figure 2 - the relationship between qp and actual bit rate and a surrogate for encoding complexity. However, the bits and complexity terms are associated only with the residuals because qp can only influence the detail of information carried in the transformed residuals.

The traditional model takes an algebraic form

$$ResidualBits = C_1 * MAD/qp + C_2 * MAD/qp^2. \tag{5}$$

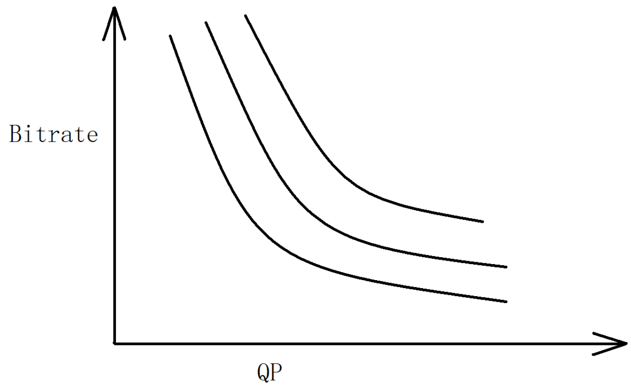


Figure 2. Increasing distortion and decrease quality

The coefficients C_1 and C_2 could be decided empirically. The mean average difference (MAD) of the prediction error reflects the encoding complexity associated with the residuals:

$$MAD = \sum_{i,j} |residual_{i,j}| = \sum_{i,j} |source_{i,j}| - \sum_{i,j} |prediction_{i,j}| \tag{6}$$

Ideally, MAD would be estimated after encoding the current picture, but that would require encoding the image again after qp is selected. Instead, it is assumed that this complexity surrogate varies gradually from picture to picture, and estimate it based upon data extracted from the encoder for previous pictures. This algorithm will fail when there is a scene change.

As we have discussed in the previous section, compound video is different from natural video in motion estimation. The object motion is very rare in compound video sequence and we only focus on scene changes. With the above rate control algorithm, when there is a scene changes, it will fail and cause network delay. Therefore, the traditional rate control algorithm is no longer suitable for compound video compression.

In order to solve this problem, we propose a new rate control model for compound video compression. At the scene change, we need to increase the in order to decrease the compressed frame size. Since the changed region is known from preprocessing, we can use the following formula to estimate qp for the current frame:

$$qp = qp_{init} + area_{changed}/area_{inital} * qp_{step} \tag{7}$$

where, $area_{changed}$ is the area that is updated in the current frame in terms of number of pixels and $area_{inital}$ is the number of pixels of the whole frame. qp_{init} is the initial qp value for the video sequence and qp_{step} is a constant number for adjusting qp. In our experiment, qp_init and qp_step are fixed. From our experiment, this new rate control algorithm can decrease the network burst by 50 percents at the scene change.

4. EXPERIMENTAL RESULTS

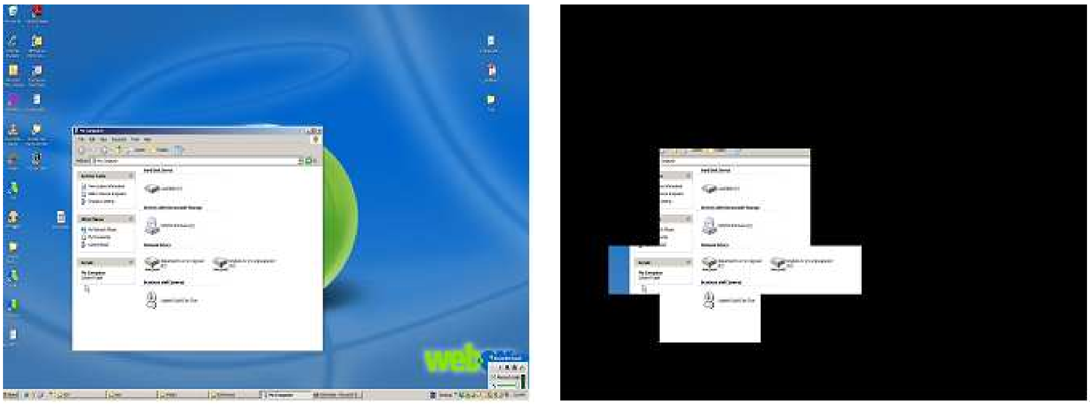
For our experiments, three 1024 by 768 true color computer screen image sequences are tested, including PowerPoint presentation, Outlook operations and mixed contents. The PowerPoint image sequence is very challenging to compress because the characters are directly drawn on background pictures and the Outlook image sequence is challenging because there is a lot characters to be compressed and it may cause large network burst.

4.1 Segmentation

The segmentation results in Fig. 3 and Fig. 4 are respectively the spatial segmentation and temporal segmentation. Fig. 3(a) is the 15th frame from the Mixed Content image sequence. It is the desktop wallpaper with a Windows file folder. Fig. 3(b) shows the segmentation result. The black area is marked as natural pictures, which contain most of the wallpaper and the icons, and the remaining area is the text and graphics part. Fig. 4(a) is the 16th frame from the Mixed Content image sequence and it is the consecutive frame of Fig. 3(a). Fig. 4(b) shows the updated region in Fig. 4(a) comparing to Fig. 3(a). The entire changing region is marked out in 128 by 128 blocks.

(a) The 15th frame in the mixed content video sequence (b) The binary mask to separate Text/Graphics and natural image

Figure 3. Spatial segmentation result



(a) The 16th frame in the mixed content video sequence (b) The binary mask to indicate changed areas

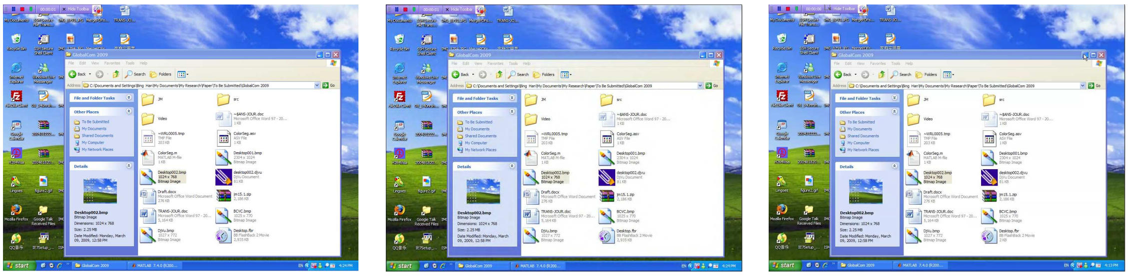
Figure 4. Temporal segmentation result

4.2 Compound Video Compression

Fig. 5-6 compare the visual quality of the reconstructed images compressed by BCVC and DjVu. In terms of reconstructed visual quality, BCVC performs much better than DjVu because motion compensation is widely used in video compression. For DjVu, or any other image compression algorithms, such as JPEG2000 and SPEC, they only focus on compression of a single static image. In a compound video sequence, many consecutive frames are the same. Therefore, we do not need to encode the first frame perfectly and can compensate it gradually. With a moderate frame rate, the image can become clear very quickly and it is not noticeable. The final reconstruction result by BCVC is much better than DjVu which only compress the frame once and then skip all the other frames.

4.3 Network Consistency

The BCVC is designed for real-time application of compound video transmission. Since the resolution of the compound video is usually very large, usually at least 1024 by 768, it is important to keep the compressed bit



(a) Original image from Mixed Content video sequence (b) Reconstructed image by DjVu (c) Reconstructed image by BCVC

Figure 5. Reconstructed image By BCVC and DjVu

(a) Part of the image from Mixed Content video sequence (b) Part of reconstructed image by DjVu (c) Part of reconstructed image by BCVC

Figure 6. Part of the reconstructed image for comparison

rate as constant as possible in order to avoid network delay. As we have discussed in Section 3, the previous rate control methods will fail in compound video compression, so we propose a new rate control algorithm to achieve a more constant bit rate.

Fig. 7 shows the comparison results of coded data stream with rate control and without rate control. It shows that with our rate control algorithm, the bit rate becomes much more constant is more suitable for real time transmission in current network environment. The biggest burst is decreased from 65KB to 30KB, which is only half of original network burst. We also checked the file size compressed by DjVu, which is usually around 100KB. So BCVC is much more suitable for real time transmission of compound video sequence.

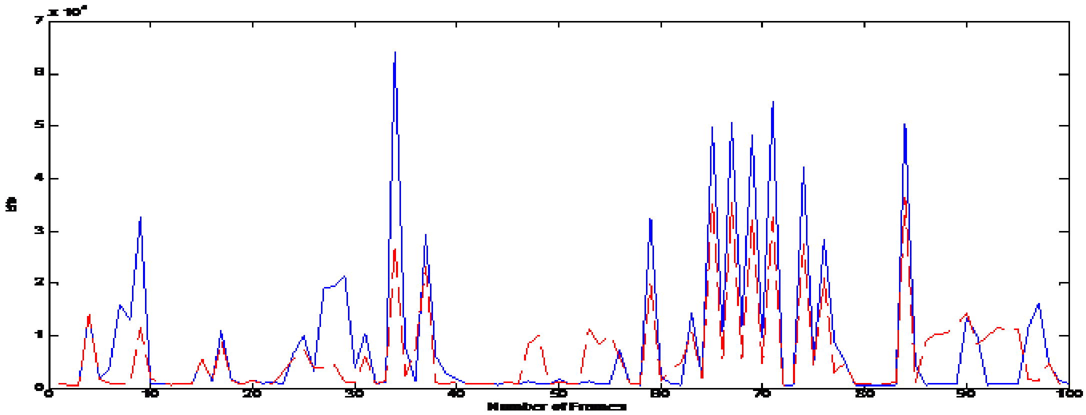


Figure 7. Comparison of codec with and without RC testing of Mixed Content video sequence at 300kbps

5. CONCLUSION

In this paper, we propose a novel method to compress the compound video sequence. Different from previous methods, which are based on single image compression, we take advantage of video compression and make it possible for real time compression and transmission. The proposed method is based on H.264 with preprocessing and we also propose an adaptive quantization algorithm and rate control algorithm in order to improve the compression performance specifically for compound video sequence. The experimental results show that the new method outperforms previous compound image compression methods in both visual quality and network consistency.

ACKNOWLEDGMENTS

Bing Han thanks Steven Huang and Xiang Li for many valuable discussions and suggestions.

REFERENCES

[1] Draft Recommendation, T., "44, Mixed Raster Content (MRC)," *ITU-T Study Group* **8**.

[2] Bottou, L., Haffner, P., Howard, P., Simard, P., Bengio, Y., and LeCun, Y., "High quality document image compression with" DjVu"," *Journal of Electronic Imaging* **7**, 410–425 (1998).

[3] Haffner, P., Bottou, L., Howard, P., and LeCun, Y., "DjVu: Analyzing and compressing scanned documents for Internet distribution," in [*Proceedings of the International Conference on Document Analysis and Recognition*], 625–628 (1999).

[4] de Queiroz, R., Buckley, R., and Xu, M., "Mixed raster content (MRC) model for compound image compression," in [*Proc. IS&T/SPIE Symp. on Electronic Imaging, Visual Communications and Image Processing*], **3653**, 1106–1117, Citeseer.

[5] Li, X. and Lei, S., "Block-based segmentation and adaptive coding for visually losslesscompression of scanned documents," in [*Image Processing, 2001. Proceedings. 2001 International Conference on*], **3** (2001).

[6] Li, X. and Lei, S., "On the study of lossless compression of computer generated compoundimages," in [*Image Processing, 2001. Proceedings. 2001 International Conference on*], **3** (2001).

[7] Ding, W., Liu, D., He, Y., and Wu, F., "Block-based fast compression for compound images," in [*2006 IEEE International Conference on Multimedia and Expo*], 809–812 (2006).

[8] Said, A. and Drukarev, A., "Simplified segmentation for compound image compression," in [*IEEE INT CONF IMAGE PROCESS*], **1**, 229–233, Citeseer (1999).

[9] Lin, T. and Hao, P., "Compound image compression for real-time computer screen image transmission," *IEEE Transactions on Image Processing* **14**(8), 993–1005 (2005).

[10] Rivest, R., "RFC1321: The MD5 message-digest algorithm," *RFC Editor United States* (1992).

[11] Zhang, H. and Cote, G., "Determining optimal configuration of video encoding parameters using numerical search algorithms," in [*Proceedings of SPIE*], **6822**, 682211 (2008).

[12] Li, Z., Pan, F., Lim, K., Feng, G., Lin, X., Rahardja, S., and Wu, D., "Adaptive frame layer rate control for H. 264," in [*Proceedings of the IEEE International Conference on Multimedia & Expo (ICME03*], **1**, 581–584.