CONTOUR APPROXIMATION FOR FASTER OBJECT BASED TRANSCODING WITH HIGHER PERCEPTUAL QUALITY

Oleg V. Komogortsev and Javed I. Khan
Media Communications and Networking Research Laboratory
Department of Math & Computer Science, Kent State University
233 MSB, Kent, OH 44242
okomogor@cs.kent.edu, javed@kent.edu

ABSTRACT
Object aware video rate transcoding can significantly improve the perceptual quality of relatively low bit-rate video. However, precise object detection in arbitrary video scenes is computationally challenging. This paper presents an interesting experimentation with a live eye-gaze tracker which suggests that object detection, particularly for perceptual encoding, may not have to be precise. Indeed, a simple approximation can not only reduce the complexity of the detection process, but also result in improved perceptual quality and yield very fast transcoding algorithms.

KEYWORDS
Transcoding, perceptual video encoding, human vision.

1. Introduction

In recent years, video transcoding has been gained increasing importance. The asymmetry in the Internet capacity, particularly at the egress networks is growing dramatically. The emerging digital video standards such as DTV or HDTV will bring an enormous flux of high quality video content. However, the relatively differential of bandwidth at network edges and the advent of small devices (such as Personal Digital Assistant) seems to indicate that in the near future, the Internet applications have to deal with increased bandwidth asymmetry. Consequently, there will be an increased need in higher video compression ratio. Most of the current video transcoding techniques are based on frame wide requantization [5,7,9,10,11,12]. Unfortunately, frame-wide requantization reduces the perceptual quality of a video very quickly. Research in first stage coding has already shown that object-based encoding plays a significant role in creating perceptually pleasant video at lower bit-rates [1,2,4,6]. MPEG-4 has been proposed to transport object-based coded video stream. However, a conversion of a regular video to an object based stream is computationally challenging [3,4] because of the high computational complexity of object detection. Thus live stream transcoding is still very difficult. It seems the first generation MPEG-4 systems will see most of its application in computer model generated synthetic video (where objects are already given), or for small format known content video [3] (such as head and shoulder videos).

1.1 Related Work

Among the latest methods employed for object detection in video, Ngo et al [8] described object detection based on motion and color features using histogram analysis. This technique was capable of processing less than 2 frames in one second. Unfortunately, many of the other techniques presented, such as [15], did not provide an evaluation of time performance. However, it depends on even more involved image processing methods, such as active contour model, which spends considerable effort determining the boundary shape, and is likely to be slower. More recently, some compressed domain techniques have been suggested, such as by Wang et al [14]. This system achieved about 0.5 sec/frame for the CIF size on a Pentium III 450 MHz. It should be noted that the stream transcoding has some prominent difference from first stage video encoding. First of all, in video transcoding, object detection has to be performed extremely fast at the rate of the stream. Secondly, the transcoder receives an already encoded video stream. An unprocessed input stream seldom contains pixels level information, it rather contains highly structured coded and refined information such as motion vectors, thus access to pixel level information means severe computational overhead while in the first stage encoding, the opposite is true. A real-time video transcoder must take advantage of such a scenario to be effective.

1.2 Computational Complexity

It seems, except from the compressed domain techniques, most of the object detection methods tried in video have been derived from image level algorithms and targeted towards scene comprehension type applications. Current approaches, even when they use compressed domain fields, mix pixel level data in

analysis for boundary approximation. Thus most methods are too slow to meet the needs of live perceptual stream transcoding. It seems a major source of computational burden originates from contour estimation. While precise contour detection has been considered as a critical part of scene interpretation, and understanding image research, it is possible they have less importance in perceptual encoding. To answer this question, we have recently performed a live eye-gaze study to observe the perceptual effectiveness of several fast algorithms for perceptual coding incorporating contour approximation of various degrees.

1.3 Perceptual Quality

Over the years, eye-gaze research has shed significant light on visual perception. The retinal distribution of photoreceptor cells is highly non-uniform. Correspondingly, only about 2 degree in our about 140 degrees vision span has sharp vision [1,2]. Scientists have identified several intricate types of eye movements, such as drift, saccades, fixation, smooth pursuit, involuntary saccades. An image quality perception is known to be highly correlated with the image quality around the eye fixation points. Visual sensitivity reduces exponentially with eccentricity from the fovea.

In this paper we will present a direct comparison of the object approximation techniques based on live eye-gaze. The study suggests indeed approximation not only significantly reduces the complexity of the object detection process, but also results in improved perceptual quality. The schemes can provide dramatic speed improvement in MPEG-2/MPEG-2 or MPEG-2/ MPEG-4 object based perceptual transcoding.

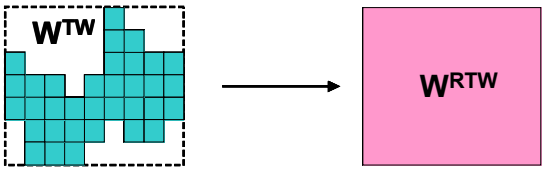

Fig. 1 Rectilinear approximation of the Tracking Window – W^{RTW}.

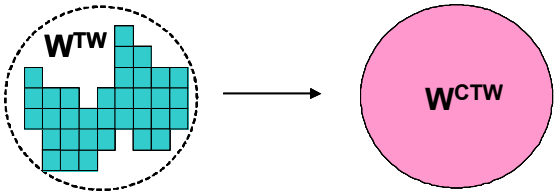

Fig. 2 Circular approximation of the Tracking Window – W^{CTW}.

2. Contour Approximation Approach

We started with a fast base BOF algorithm [13]. Though the algorithm itself is not the focus of this paper, we will describe it briefly. This is a strictly compressed domain method which tracks object with only motion vector analysis. It estimates the projection of various scene objects, background motion, and camera motions on the motion vectors in the coded stream. By observing the motion vectors, it then detects and tracks video objects. It is important to note that motion vectors are not simply codified information about a block's motion, as it may appear at first glance. Rather these also contain highly refined color, texture and shape information. However, what they cannot contain is precise shape information beyond block boundaries. Using Kalman filter prediction this fast algorithm automatically detects, and tracks the region covered by the scene objects for subsequent perceptual encoding. A detail of the algorithm is given in [13]. To study the impact of contour approximation we incorporate two contour approximation techniques with this algorithm. To measure the impact of the approximations on perceptual efficiency, we then play the live video to a subject and directly observe the eye gaze fixations on the video frames.

2.1 Approximated Tracking Window

The BOF algorithm for each object provides an object tracking window called W^{TW}. This window tracks the moving blocks corresponding to a moving object in the scene. We consider three versions of the algorithm. The first is the raw object tracking window and the two approximations are explained below.

2.2 Rectilinear Approximation

The first is the rectilinear approximation. All macroblocks in the original W^{TW} are sorted according to their coordinates and the rectilinear tracking window (W^{RTW}) is constructed using the min max corners. $W^{RTW}=\{(x_{min},y_{max}),(x_{min},y_{min}),(x_{max},y_{max}),(x_{max},y_{min})\}$, where $x_{max}=\max\{x_i\}$, $x_{min}=\min(x_i)$, $y_{max}=\max\{y_i\}$, $y_{max}=\max\{y_i\}$, where all $x_i, y_i \in W^{TW}$. The idea of such approximation is presented in Fig. 1.

2.3 Circular Approximation

The other form of approximation of W^{TW} is circular approximation or circular tracking window (W^{CTW})

shown in Fig. 2. This approximation is built in the following way. For all macroblocks inside of W^{TW} we find a pair of macroblocks that are the most distant from one another. Suppose that $MB_k[x_{k,}y_k]$ and $MB_m[x_{m,}y_m]$ are such macroblocks. Let the distance between them be $D_{km} = \sqrt{(x_k - x_m)^2 + (y_k - y_m)^2}$, then we define W^{CTW} as a circle with radius $R=0.5D_{km}$, and center at $(x_c = \frac{x_k + x_m}{2}, y_c = \frac{y_k + y_m}{2})$.

3. Experiment

3.1 Setup

Our MPEG-2 transcoding system was implemented with integrated Applied Science Laboratories' high speed eye-tracker model 501. The eye position video capturing camera worked at the rate of 120 samples per second. For this experiment, an eye fixation was defined as a period of time when an eye did not move more than 1 degree in 100msec. All tested video clips were 720x480 pixels and were captured with a Sony TRV20 digital camera at a frame rate of 30 frames per second. The duration of each video clip was around 2 minutes. The number of frames per GOP was 15. Number of "P" frames between any given two "B" frames was two. All video were projected on the wall of a dark room. The projected image physical dimensions were: width 60 inches, height 50 inches, and the distance between subject's eyes and the surface of the screen was around 100-120 inches.

3.2 Video Sets

Perceptual encoding is highly dependent on the video content. Therefore, we avoid providing any 'average' performance when we evaluate our system. Rather, we use test videos carefully selected to offer a special challenge to our system. In this paper we present three such videos. (a) "Video 1" contains a car driving in a parking lot. The object speed in that video is smooth and continuous. (b) "Video 2" has two radio controlled toy cars moving at different speeds with rapid unpredictable movements. In this video we asked a subject to concentrate on just one car. (c) "Video 3" has two relatively close up toy cars offering much larger area of focus. Cars move in different directions inconsistently. A subject was asked to concentrate only on one car.

Fig. 3-6 show some snapshots from the videos we tested. Fig. 3 and Fig. 5 present a circular approximation of tracking window. Fig. 4 and Fig. 6 show rectilinear approximation. Video transcoding and perceptual adaptation was done in real time. The original test videos bit-rate was 10Mb/s. As a result of perceptual compression the bit-rate was reduced down to 1Mb/s. During bit-rate reduction full resolution was maintained at the approximated windows macroblocks. The MPEG-2 TM-5 rate control was used to determine the quantization of the remaining macroblocks. The actual perceptually encoded video samples including the original videos can be obtained for direct visual appreciation from [16].

3.3 Gaze Containment Efficiency

Ideally, if all eye gazes fall within the tracking window, it is possible to design an optimum perceptual encoder. We define the quantity *average gaze containment* is a fraction of gazes successfully contained within a window:

$$\xi = \frac{100}{N} \sum_{t=1}^{N} \frac{|E^W(t)|}{|E(t)|} \qquad (4.3)$$

Where, E(t) is the entire eye sample set and $E^W(t) \subseteq E(t)$ is the sample subset contained within a tracking window W(t) for frame F(t). N is the number of frames in a video.

The right y-axes of Fig. 7 shows the results of gaze containment for W^{TW} W^{CTW} and W^{RTW} for the three videos. Compared to the strict object boundary based W^{TW}, both of approximations techniques increase average gaze containment significantly. As we can see from performance results for "Video 1", gaze containment increased from 49% to about 70% for rectilinear approximation and to about 89% for circular approximation. "Video 2" results indicate gaze containment increase from 52% for original tracking scheme to 93% for both circular and rectilinear approximation. "Video 3" case shows gaze containment increase from 53% to 84% for W^{CTW} and 76% for W^{RTW}.

We can see from the results that the circular approximation gives the best containment results for all three videos.

It is interesting to note that the base BOF algorithm for W^{TW} itself uses block approximation. Thus a true object contour is expected to contain less than 50% of the eye-gazes.

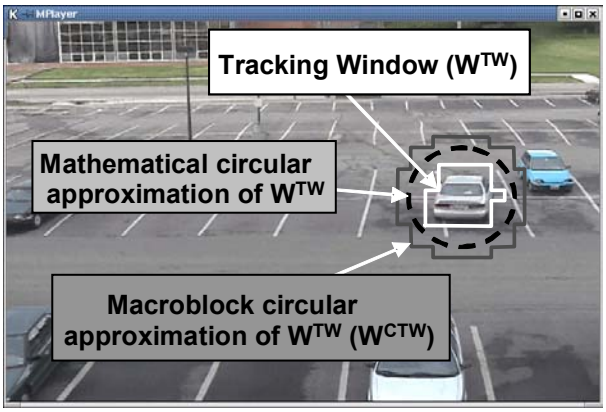

Fig. 3. "Video 1". Circular approximation. Frame 233.

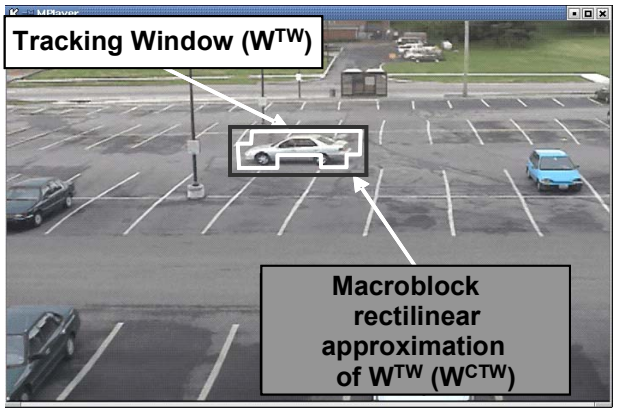

Fig. 4. "Video 1". Rectilinear approximation. Frame 1343.

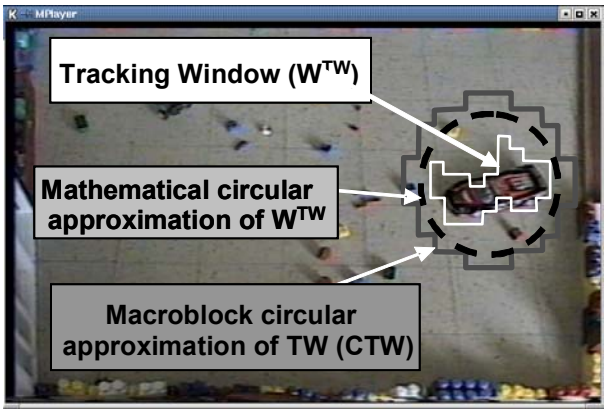

Fig. 5. "Video 2". Circular approximation. Frame 509.

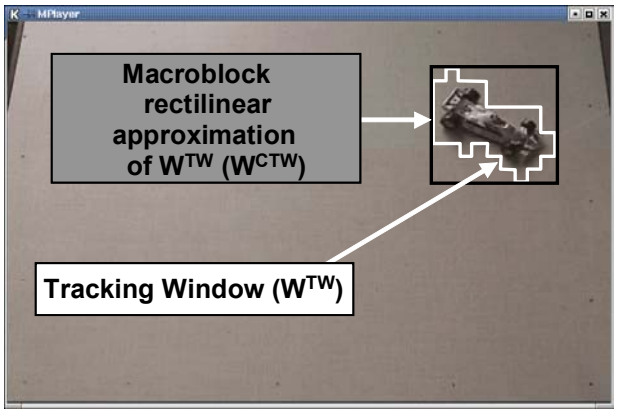

Fig. 6. "Video 3". Rectilinear approximation. Frame 255.

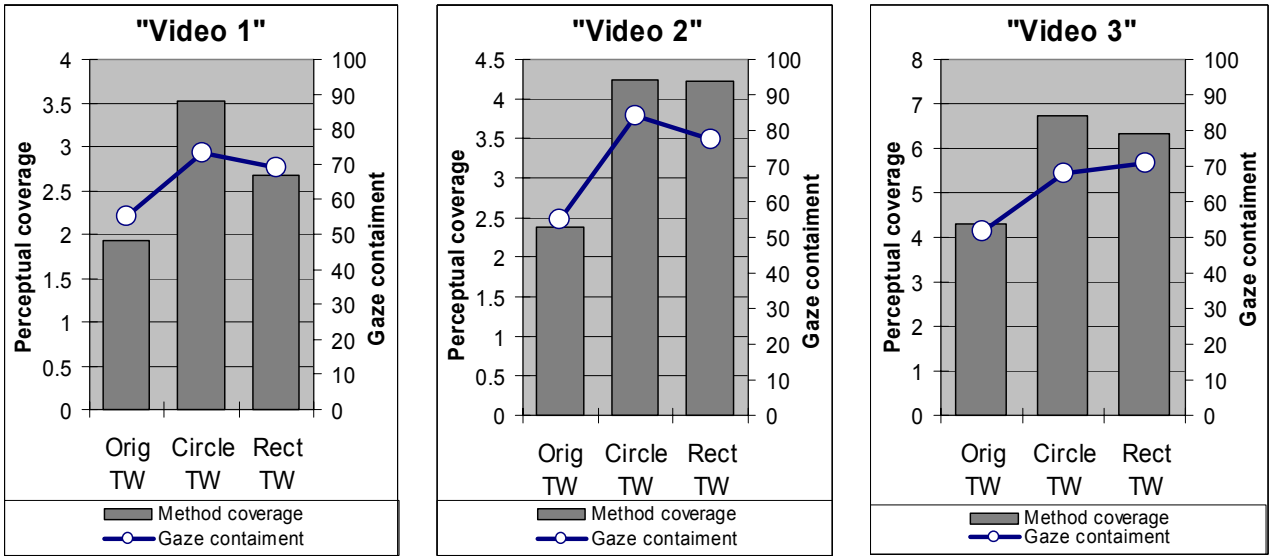

Fig. 7 Gaze containment and video frame coverage by different W^{TW} approximation methods.

3.4 Perceptual Coverage Efficiency

With a larger size of a tracking window more eye gazes can be contained, however, there will not be any perceptual redundancy to extract. Therefore, we were curious to see how small constructed windows were. We define a second performance quantity called *average perceptual coverage*.

$$\chi = \frac{100}{N} \sum_{t=1}^{N} \frac{|\Delta(W(t) \cap F(t))|}{|\Delta(F(t))|} \qquad (4.4)$$

Where, F(t) is the size of the total viewing frame, and W(t) is a tracking window which is being evaluated (delta for area or volume).

The bars plotted on the left y-axes of Fig. 7 show the perceptual coverage values for all three tested cases.

"Video 1" case shows perceptual coverage increase from 1.9% of the total visual plane for original tracking method to 2.9% for circular approximation and 2.7% for rectilinear approximation. "Video 2" perceptual coverage equals to 2.5% in case of W^{TW}, 3.8% for W^{CTW} and 3.5% for W^{RTW}. From test results of "Video 3" we can see perceptual coverage increase from 4.1% for original tracking to 5.4% for W^{CTW} and 5.7% for W^{RTW}.

It seems that circular approximation slightly outperforms rectilinear one in by the number of gazes contained. The increase in the area size for the circular approximation is negligible comparing to the total video frame size. Thus, we can conclude that circular tracking window W^{CTW} is the optimal choice for object approximation used for perceptual object video coding.

4. Conclusions & Current Work

Many of contemporary object-based perceptual coding techniques are based on the implicit assumption that the object area inside the object contour should be coded with a higher resolution. That assumption is perhaps imperfect. It is highly likely that the mental process of visual perception drives eyes to scan areas beyond particularly the area slightly outside the precise object boundary. The high emphasis on object boundary has lead to many involved schemes for video object detection. It seems the efforts spent in exact contour extraction, is perhaps counter productive in perceptual coding. The proposed approximations can lower cost and improve performance effectiveness. Without significant increase of the perceptual coverage we were able to achieve around 90% gaze containment in the proposed simple approximations. Clearly, other approximations techniques to suit the video coding constraint can also be designed.

This research has been supported by the DARPA Research Grant F30602-99-1-0515.

References:

[1] Z. Wang, and A. C. Bovik, Embedded foveation image coding, *IEEE Trans. Image Proc*., Oct. 2001

[2] Z. Wang, Ligang Lu, and Alan C. Bovik, Rate scalable video coding using a foveation-based human visual system model, *ICASSP 2001*.

[3] Aizawa, K., H. Harashima, & T. Saito, Model-based image coding for a person's face, *Image Commun*, v.1, no.2, 1989, pp 139-152.

[4] Hotter, M., & R. Thoma, Image segmentation based on object-oriented mapping parameter estimation, *Signal Process*., v. 15, 1998, pp.315-334.

[5] Keesman, Gertjan; Hellinghuizen, Robert; Hoeksema, Fokke; Heideman, Geert, Transcoding of MPEG bitstreams, *Signal Processing: Image Communication,* Volume: 8, Issue: 6, pp. 481-500, September 1996.

[6] Javed I. Khan, Q. Gu, Network Aware symbiotic video transcoding for instream rate adaptation on interactive transport control, *IEEE Int. Symp. on Network Computing and Applications*, IEEE NCA' 2001, Oct, 8-10, 2001, Cambridge, MA, pp.201-213.

[7] Youn, J, M.T. Sun, and J. Xin, Video transcoder architectures for bit rate scaling of H.263 bit streams, *ACM Multimedia 1999'*, Nov., 1999. pp243-250.

[8] Ngo, Chong-Wah, Ting-Chuen Pong and Hong-Jiang Zhang, "On clustering and retrieval of video shots", *ACM Multimedia 2001*, Oct., 2001. pp51-60.

[9] U. Chong and S. P. Kim, Wavelet trancoding of block DCT-based images through block transform domain processing, *SPIE Vol. 2825, 1996*, pp901-908.

[10] Niklas Björk and Charilaos Christopoulos, Video transcoding for universal multimedia access; *Proceedings on ACM multimedia 2000 workshops*, 2000, Pages 75 - 79

[11] J. Youn, M.T. Sun, and C.W. Lin, Motion vector refinement for high performance transcoding, IEEE, *Transactions on Multimedia, Vol. 1, No. 1, pp.30-40*, March 1999.

[12] P. Assuncao and M. Ghanbari, A frequency-domain video transcoder for dynamic bit rate reduction of MPEG-2 bit streams, *Trans. On Circuits Syst. Video Technol., vol. 8, no. 8, pp. 953-967*, 1998.

[13] Khan Javed I. Zhong Guo, & W. Oh, Motion based object oracking in MPEG-2 stream for perceptual region discriminating rate transcoding, *Proceedings of the ACM Multimedia, 2001*, October 2001, Ottawa, Canada, pp. 572-576.

[14] Wang R., H.J. Zhang and Y.Q. Zhang, A confidence measure based moving object extraction system built for compressed domain, *ISCAS 2000 - IEEE International Symposium on Circuits and Systems*, May 28-31, 2000, Geneva, Switzerland, pp.V21-24 2000.

[15] Kuehne, Gerald, Stephan Richter and Mark Beier, "Motion-based segmentation and contour-based classification of veideo objects", ACM Multimedia 2001, Oct., 2001. pp41-50.

[16] O. Komogortsev, & Javed I. khan, TR2002-06-01 Perceptually Encoded Video Set from Dynamic Reflex Windowing, Technical Report, Kent State Univrsity, Jun 2002, http://medianet.kent.edu/techreports/TR2002-06-01/TR2002-06-01-videoset-KK.htm.