KOMOGORTSEV, OLEG M.S., May, 2003 COMPUTER SCIENCE

DYNAMIC GAZE SPAN WINDOW BASED FOVEATION FOR PERCEPTUAL MEDIA STREAMING (pp. 70)

Director of Thesis: Javed Khan

Human vision provides numerous opportunities for video data-compression. Human vision extends about 140 degree, but only about 2 degrees have sharp vision. A fascinating body of research exists in vision and psychology geared towards the understanding of the human visual perception system. This thesis presents a novel eye-gaze enhanced media transcoding system for streaming video. This scheme includes a video server, a real-time performance capable media transcoder, a video player and an eye tracker. The system intakes live perceptual information related to a subject's eye position. Eye and head movements are detected via an eye-tracker, and a magnetic head tracker. A unique challenge of this real time perceptual adaptation scheme is the incorporation of fast eye movement mechanisms into a complex MPEG-2 transcoding scheme. An important factor in this perceptually adaptive encoding method is the delay between the time an eye-gaze sample is taken and the time the coding response arrives on the screen. This delay is particularly critical if the video involves network transmission. The delay also usually increases when large format media is to be perceptually transformed due to the coding complexity. This thesis investigates this feedback delay compensation problem and proposes a novel gaze interaction based foveation windowing scheme to solve it. The proposed technique is able to contain 90% of the gazes within 20-

25% window coverage area. The media transcoder developed on this scheme is one of the first eye-gaze based perceptual transcoders. It can be used between the server and the video player in a networked environment. The architecture of the transcoder is designed to allow transmission of both stored and live media. Though the architecture is independent of any media type, this system currently handles ISO/IEC 13818-2 MPEG-2 standard.

DYNAMIC GAZE SPAN WINDOW BASED FOVEATION FOR PERCEPTUAL MEDIA STREAMING

A thesis submitted

to Kent State University in partial

fulfillment of the requirements for the

degree of Master of Science

by

Oleg Komogortsev

May, 2003

Thesis written by

Oleg Komogortsev

B.S., Volgorad State University, 2000

M.S., Kent State University, 2003

Approved by

_____ , Advisor

_____ , Chair, Department of Computer Science

_____ , Dean, College of Arts and Sciences

TABLE OF CONTENTS

LIST OF FIGURES

Acknowledgements

I would like to express my thanks to my advisor Dr. Javed Khan, for constant guidance and support.

I want to thank members of the Medianet lab group Darshan Patel, Wansik Oh, Seung S. Yang and all others for helping with my research work

Special thanks to Jessica Erin for helping me with English grammar and syntaxes.

CHAPTER 1

Introduction

Current visual data compression schemes are based on statistical redundancy analysis. This thesis researches a novel video compression scheme that attempts to explore the human perceptual characteristics. The human vision offers a tremendous scope of perceptual data compression. Only about 2 degrees in our 140 degree vision span have sharp vision. There are two kinds of photo-receptor cells in a human eye: rods and cones. They play a crucial role in our vision. Cones provide color information, and are effective only for daylight vision. The rods are more sensitive to light than the cones; they function primarily during night vision. Cones provide fine-grained spatial resolvability of the visual system. These two photoreceptor cells are connected to the ganglion cells. Ganglion cells are the output cells of the retina, and their axons form the optical nerves. Photoreceptor cells are not uniformly distributed, but are concentrated in the central part of the retina (or fovea). Human acuity perception is related to the sampling density of the photo-receptors in the retina, and the mapping of the photo-receptors to the ganglion cells. The density of cells falls off sharply outside the fovea. The diameter of the highest acuity circular region subtends only 2 degrees, the parafovea (zone of high density) extends to about 4 to 5 degrees, and acuity drops off sharply beyond. At 5 degrees acuity is only 50% [19].

This thesis proposes a technique that considers tracking characteristics of the human eye. Most of the previous research in this area has used eye-trackers as a passive instrument with standalone image/video presentation systems to gain understanding of the visual acuity distribution. This work explores how an active media transmission scheme can be built with integrated eye gaze tracking. A particularly important factor in such integrated perceptual encoding scheme is the delay between the time an eye-gaze is tracked and the time the coding response arrives at the screen. This delay is particularly significant in systems that involve network transmission. It is also substantial when large format media is to be perceptually encoded. In particular, this thesis addresses the problem of how this delay issue can be addressed by a novel dynamic window mechanism.

1.1 Related Work

A large number of studies have been performed which investigated various aspects of perceptual compression. The overall research problem can be divided into several research issues. This section summarizes important past research that is particularly relevant to this work.

1.1.1 Contrast Sensitivity and Spatial Degradation Models for Image Encoding

A Specific focus has been the study of contrast sensitivity and spatial degradation models around the foveation center and its impact on the perceived loss of quality by subjects [6, 8, 13, 14, 16]. These studies suggest a potential for significant bit-rate reduction from perceptual compression.

For example [12] presented 256x256x8 gray scale images to subjects, and gave them some visual task (such as reading, face detection/evaluation, clutter evaluation) to evaluate his system performance. The images he used covered roughly a 20-degree field of vision. The purpose of the experiment was to observe the visual sensitivity degradation in the para-foveal region. The researcher was able to achieve up to a 94.7% bandwidth reduction without any perceived loss in the subject's ability to perform the visual tasks.

Niu [16] studied the potential of wavelet-based decomposition for creating dual resolution still image coding. One valuable aspect of Niu's work was that he reported results that show a relationship between the foveal window size, allowable quality degradation, and corresponding ability of human eye to detect such degradation. He discovered that if periphery quality was degraded to a level 7, (only first seven bands of wavelet coefficients was retained), and the dual-resolution encoded image with ROI (region of interest or foveal window) size of 2 degree was presented for 150 ms to a subject, then the subject could detect degradation about 60% times. ROI window of size 5 degrees reduces the detection level to less than 20% of the times. The foveal window size has a proportional relationship with the possible image compression. Niu estimated that if the 'Zerotree' quantization of wavelet coefficients is used for subsequent image coding then about 75-62% bit reduction is possible with the 2-degree window. Possible compression was reduces to approximately 44-31% with the 5-degree window.

Loschky and McConkie [14] repeated Niu's experiment with multi-resolution display and higher resolution original images (768x512). They observed similar (60% with 2-degree ROI window and 20% with 5-degree) results. However, giving the subject a

secondary task (searching for some object as opposed to simple detection of degradation) resulted in higher detection rate of artifacts. The concluding comments of this research indicated that the high-resolution window size has to increase in order to maintain the high level of performance. Also, this study noted whether the edge between the high and lower resolution areas of the image is sharp or softened made no difference in the detectability of the visual artifacts. This observation conflicts with several other works. Overall this study suggests that a 4.1-degree foveal window is sufficient to achieve near-normal visual performance, indicating that a subject was not able to detect perceptual compression if the coding system used foveal window of that size. However, the authors of this paper did not consider if the dimensions of the foveal window should change in the case that an additional observing task would be given to the subject. Several scientists reported that the foveal window size could have a reverse impact on the visual search process such as by reducing the eye saccade lengths [27], or by resulting in longer eye fixations [28].

1.1.2 Perceptual Coding Techniques for Image Encoding

A number of coding techniques suitable for varying the spatial resolution of the image plane has been suggested: Wavelet-based Spatial Coding [16, 24], Spatial Domain Multi-resolution Coding [7]; Multi-resolution Medical Image Coding [11], Retinal Coding [13], Progressive Coding [17], etc.

The author of [7] builds a sequence of reconstructed images from the original image. This sequence consists of the same images, except the pixel dimensions of each image differ. Several visual degradation functions such as linear, non-linear and human visual

system (HVS) acuity-matching are used to create a fovea compressed image from sequences of images as described above.

The example of Retinal Coding would be the Retinal Reconstructed Images (RRI) coding [13]. Instead of uniform rectilinear spaced organization of image information, this work regards spatial organization and the density of ganglion cells in the human retina. It considers the circular symmetry of the human eye and organizes the image bit distribution based on viewing distance, point of eye fixation, size of the input image and number of cells around the point of fixation. In the decoding phase, the RRI organized samples are projected back on the image screen using B-spline based image reconstruction. The experiment reports nearly 2 times compression on image data. The coding/decoding complexity is inverted. The B-spline sampling has $O(n^2)$, complexity, but the surface reconstruction hs $O(n^2s^2)$, where s is the number of samples per surface.

[10] presented pyramid coding and used a pointing device to identify focus. Their work includes a hierarchical stage-by-stage motion vector estimation technique, which fits with the original pyramid-scheme and enables predictive coding. Their work used a HVS model for the inclusion of spatial frequency data in coding. It is based on a functional model that connects the threshold of contrast sensitivity of the human eye with the retinal eccentricity and spatial frequency. It then determines which spatial frequency component is visible to the human eye based on the distance. Accordingly, only the visible components are added in the code. It reported compression results on several image sequences, which reduced the frame size about 7 times. The method also used blending near the pyramid block boundaries. The method required low-level image

processing, which is extremely time consuming. It required full decompression of the image. In almost all practical situations, even the raw captured video requires compression. Complexities of the recognition filters/ algorithms are super-linear. Thus, the complexity of the procedures becomes intractable with high-fidelity large format video. Object model is not available for most situations. If the object model is available, even then the acuity distributions for various objects are needed to be determined case by case as well. Thus computational complexity of video coding scheme is a formidable challenge.

1.1.3 Perceptual Video Encoding

Several investigations studied video encoding in particular [4, 10, 20, 22, 23, 25, 26]. In [25] research, twenty-four observers viewed 15 forty-five-second clips of NTFS video while the direction of the gaze was monitored. Video frames where divided on clusters. The size of each one was 6% of the video frame. Dominant cluster contained between 78% of subject gazes for the video samples with considerable motion, and 43% for the slow motion. This result shows that many people have a tendency to look at the same part of the image. [26] concluded that the benefits of eye-gazed compression are modest, and the high cost of implementation makes the gaze contingent processing not suitable for general purpose image processing. This paper used the same approach as in [25], but in these experiments they tried to encode image based on the area where subjects previously looked. This scheme didn't work as well as it was expected. Subjects were able to notice blurred areas on the image. The author notes that it might be due to the fact that subjects looked at the different areas each time they observed a video sequence. The conclusion is

that pre-encoding based on previous areas of attention is not as efficient as expected; as a subject may look at different areas of the image upon repeat viewing.

Many of the early works have been inspired by the objective of designing a good display system [4, 14, 20]. For example, [4] used a live-eye tracker to determine the maximum frequency and spatial sensitivity suitable for human eye perception using HDTV displays with fixed observer distance. [20] experimented with two models of degradation. They used very low-resolution 8-second video clips. The video clips were gray scale, wavelet encoded and 16 frames per second. The size of the image was 256x256 pixels. Both used stand alone presentation, fixed observer distance, and fixed sized acuity windows. Integration of live eye-tracker with video live encoding was problematic. Also in this work, the authors called to attention that there is no significant difference between linear and acuity matching resolution degradation.

A particularly interesting work by [5] studied facial video. Instead of eye-gaze, it used image analysis to monitor the face image, and used its center as the point of focus. They suggested a *contrast sensitivity function* (CSF) to degrade the resolution from the detected face center before presenting it to the subject. This work reported almost 50% bit reduction using this technique. However, the geometric center of a known object, (such as the center of face is used here), was not necessarily the center of foveation.

1.1.4 Latest Research

Earlier experiments used a lower bit-rate, smaller formats. A short time ago, [23] presented a fast DCT based transcoding technique that can be used for variable spatial resolution coding. They focused on the frame prediction problem that arises in such

compressed domain transcoding. [22] investigated how to control the bit-rate for MPEG-4/ H.263 stream for foveated encoding optimally. Their simulation used a set of given fixation point(s) and predicted about 8-52% bit-rate saving for I pictures and about 68% for P pictures for 352x288 video sequences.

1.2 Thesis Overview

1.2.1 Proposed Perceptual Video Encoding System

This thesis describes a recently completed live foveation integrated media transmission scheme. That system integrates a real-time live media transcoder with a live-eye tracker. The system intakes live perceptual information related to a subject's eye position and head-movement via an eye-tracker and a magnetic head tracker and correspondingly controls the spatio-temporal resolution of the presentation. The eye-tracker tracks the eye-gaze with respect to the human head. The magnetic head tracker detects the movement of the head with respect to the scene plane, and together they determine the eye-movement with respect to the presentation.

The Percept Media Transcoder (PMT) unit interfaces the perceptual information derived from the perceptual sensors, and applies it to the media specific perceptual encoding. The PMT architecture has been designed so that multiple media types and media specific perceptual transcoding modules can be plugged into it without requiring the reorganization of the overall media distribution systems networking. That architecture enables one to use any standard-based media server and presentation system. The perceptual compression scheme described in this thesis is incorporated into a full-logic MPEG-2 high-resolution region-based motion-vector reprocessing transcoder.

1.2.2 Challenges

The perceptual encoding system approach approximately divides the problem into two challenges. The first (and more well studied problem) is how to associate the para-foveal degradation at the boundary of the eye containment zone with specific spatial resolution (quantization value, color) on the display based on the specific media type and modality. The second challenge is that the interaction delay manifests the challenge of how to predict and estimate the gaze containment so that the spatial resolution can be applied in the right place.

Many of the previous research in eye acuity and sensitivity did address the first part. Almost no literature exists on the second issue. Most, previous studies used point-gaze(s) and placed emphasis on the spatial degradation from the point-gaze as a principle source of data reduction. However if the assumption of the availability of precise and instantaneous tracking is taken away, the size of this proximity would play a much more dominate role in bit-reduction (and probably also in perception) of video.

The idea of certain forms of the containment window immerged in several previous research studies, such as object based perceptual compression. However, these were static and fixed size windows on the object (statically detected by scene analysis, or pointing device). Still these were (i) window of fixed size, and (ii) the impact of control loop delay was not considered. Such a delay is critical to consider in a feedback based perceptual video transcoding scheme. The delay is created between when the eye position is detected and the time a perceptually encoded frame is displayed. It is important to note that a network transmission not only imposes control loop delay, but also the delay is

dynamically varying. As it will be shown later, feedback delay seems to be playing a dominant role in the actual determination of fovea region. A typical network delay ranges from 20 msec to a few seconds. Saccades can move the eye position more than 10-100 degrees in that time potentially wiping out the entire advantage of designing an accurate acuity window within the 2 degrees of foveation.

1.2.3 Objectives

The goal of the thesis is to investigate the feedback delay between the eye-tracking, coding, and displaying the encoded image and develop an approach that can operate with such dynamically varying delay inside of perceptual compression scheme.

Instead of relying only on the *acuity matching* model, the integrated approach of *gaze proximity prediction and containment* is proposed. Previous research has found it extremely difficult to model the precise eye movements. This is even more difficult in a moving scene. This thesis explores a method which instead of dealing directly with individual eye position based precise acuity, uses a gaze proximity zone or a *foveation containment window*, and approximate acuity. The main objective is to ensure that the majority of the eye-gazes remain within the window with a statistical guarantee. The advantage of this scheme is that it can compensate for the unpredictability and instability arising from fine grain foveation tracking and the delay inherent in media encode/decoder loop in any unicast/broadcast scenario.

However, it should be clarified that this thesis does not attempt to propose any new video coding scheme. The scope of this work has been confined to the industrial MPEG-2 video stream. MPEG-2 standard provides crucial capabilities such as VCR control,

multiple program multiplexing, de-multiplexing, expandability and compatibility. It is interesting to note the original MPEG-2 TM-5 model does include static human visual systems (HVS) characteristics, including a crude control for dynamic perceptual considerations (*macroblock activity* factor).

1.2.4 Thesis structure

This thesis is organized in the following way.

Section-2 describes human eye movement characteristics that play important role in the proposed gaze containment algorithm. That section contains the construction of the HVS acuity model, and the dynamic eye-movements tracking method. It shows how to combine these two topics to determine the dynamic gaze vicinity. The mechanism of predicting future eye movements is described further in the section-2.

Section-3 provides the applied techniques for perceptual transcoding for an MPEG-2 stream.

Section-4 presents a series of experiments to show characteristics of this novel system. For this purpose a set of rigorous tests was defined and eye containment performance results are presented.

Section-5 brings the discussion about the perceptual encoding limitations addressed by the proposed system. The future area of research is discussed at the end of the section.

CHAPTER 2

Eye Focus Tracking by Reflex Windowing

2.1 Visual Dynamics

The proposed eye-containment system is intricately related to the movements of human eye. An overview of movement characteristics is presented below.

Human beings ability to perceive information is affected by different kinds of eye movements. Each of these movements plays its role in the process of gaining information, and it is important to identify them for better understanding of our vision. Scientists have identified several elaborate types of eye movements, such as drift, saccades, fixation, smooth pursuit and involuntary saccades.

Saccades: Saccades are the eye moments that occur between two points of fixations (to be explained shortly) are called saccades. They are accomplished by eye movements of a single type – identical and simultaneous very rapid rotations of the eyes. Amplitude of the saccade usually doesn't exceed 20 degree. For angels less that 1 degree the duration of the saccade is 0.01-0.02 sec; for angles of 20 degrees it may reach 0.06-0.07 sec. The maximum velocity reached by the eye during a saccade of 20 degrees is 450degr/sec.

Fixations: Several types of eye movements also take place when the object of perception is stationary relative to the observer's head. Human's eye moves in three way during fixation: by small involuntary saccades, equal for the two eyes, drift and by

tremor. During long fixation 97% of time is drift and only 3% small involuntary saccades [18].

Drift: A drift is an irregular and relatively slow movement of the axes of the eyes, in which the image of the fixation point for each eye remains inside fovea. Drift movements prevent the formation of the empty field. Drift is always accompanied by a tremor. The average duration of the drifts is from 0.3 to 0.8 sec in case when a subject is observing a stationary object. Drift speed varies chaotically from zero to approximately 30 minutes of angle per second.

Tremor: A Tremor is an oscillatory movement of the eyes of high frequency but low amplitude. The amplitude of the tremor is 20-40 seconds of the angle. Frequency of the tremor movements is 70-90 oscillations per second.

Involuntary saccades: Small involuntary saccades usually arise when the duration of fixation on a particular point of a stationary object exceeds a certain length of time (0.3-0.5 sec) or when, because of drifts, the image of the point of fixation becomes too far removed from the center of the fovea.

This Thesis particularly concentrates two major types of eye movements: fixations and saccades.

2.2 Perceptual Compression Scheme Overview

To create a perceptually encoding scheme that works properly, two problems must be addressed: the human visual system (HVS) acuity matching and eye gaze tracking. The approach of this work is to first derive a para-foveal window based on acuity - eye

sensitivity function, then add corrections that take into account the reflex eye-movements between the time the eye is tracked, and the perceptually encoded frame is displayed.

To address the eye acuity matching issue $W_A(t)$ *acuity window* (AW) is constructed. AW presents bit distribution, which matches HVS during eye fixation.

To predict future eye movements $W_R(t)$ *reflex window* (RW) is built. RW represents a container for saccadic eye-movements. It represents the area, where AWs should be placed in the future.

$W(t) = f(W_A(t), W_R(t))$ *visual window* (VW) is a combination of AWs and RW. VW provides imperceptible HVS enhanced video compression for the real time encoding scheme.

Next three subsections correspondingly describe the design on these three windows. Subsection-2.6 describes the processing used to convert the individual eye tracker sample data stream into the model parameters, so the visual window is predicted dynamically as per the model.

2.3 Eye Acuity Model

As previously mentioned, the perception of image quality depends of the spatial distribution and mapping of cones, rods and ganglion cells, and the mapping of the visual fields across the visual context. A large number of functions have been suggested for *contrast sensitivity function* (CSF). Some of them are based on anatomical considerations and some are based of psycho visual empirical studies. In this work CSF presented by equation 2.3 is used, which has been modeled after the CSF function presented by [5].

This equation reflects entropy losses of the visual system. Also, this function addresses the issue of cones, rods and ganglion cells distribution. It is supported by two sets of data provided by [1] and [2]. Fig 2.1 shows the acuity distribution in the visual plane, created by proposed CSF function.

$$S(x, y) = \frac{1}{1 + k_{ECC} \cdot \theta_E(x, y)} \tag{2.3.1}$$

Here S is the visual sensitivity with respect to the frame coordinates (x,y) , k_{ECC} is a constant (in this model $k_{ECC} = 0.24$), and $\theta_E(x, y)$ is the eccentricity in visual angle. Within any lossy video compression method, the acuity quantity S has to be mapped to the spatial degradation functions of the given encoding scheme.

2.4 Reflex Window

The reflex window's objective is to contain the eye fixations by estimating the probable maximum possible eye velocity due to saccades. Given a set of past eye-positions, the reflex window predicts a zone the eye will be at a certain point in future with target likelihood. Reflex window is modelled as an ellipse with focuses $\{T_d \cdot V_x(t), T_d \cdot V_y(t)\}$ where T_d is *feedback delay* and $V_x(t)$ and

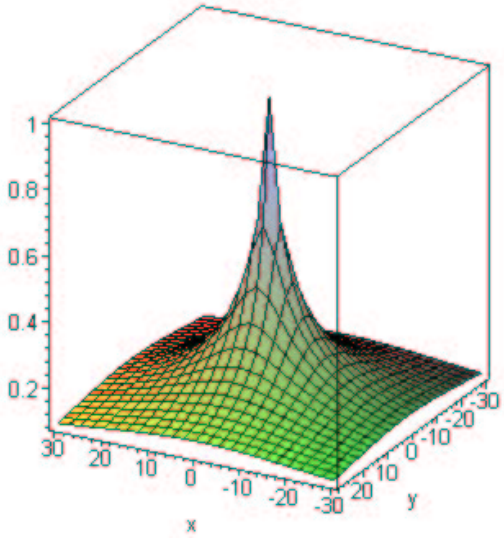

Figure 2.1: Visual sensitivity function of the human eye.

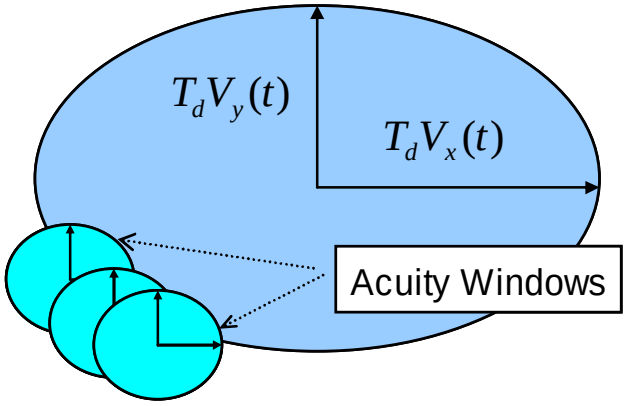

Figure 2.2: Reflex window covered by a set of acuity windows.

$V_y(t)$ are *containment assured eye velocity* (CAV). Fig 2.2 shows the diagram representation of reflex window.

2.5 Combining Acuity Window with Reflex Window

Finally, the combined window that can take into account both the acuity distribution as well as the eye motion is modeled. The eye-velocity determines the reflex ellipse. The eye is expected to be anywhere within this region. The acuity window will be added to the boundary of the reflex window to create a visual window. Fig 2.2 explains the idea. The visual sensitivity is calculated as a function of eccentricity. We assume the subject's eye will be directed anywhere within the RW with equal probability, the eccentricity is then measured as:

$$\theta_E(x,y) = \frac{180}{\pi}\arctan\left(\frac{\sqrt{\left(\frac{x-x_C}{x_R/y_R}\right)^2 + (y-y_C)^2} - y_R}{VD}\right) \tag{2.5.1}$$

In this equation x and y here are horizontal and vertical pixel positions on the video frame. VD is the viewing distance in the units of pixel spacing. Quantities x_C and y_C are the coordinates of the center of the RW window, and $x_R = T_d \cdot V_x(t)$ and $y_R = T_d \cdot V_y(t)$ are the dimensions of the reflex window.

Thus the prediction corrected sensitivity function is:

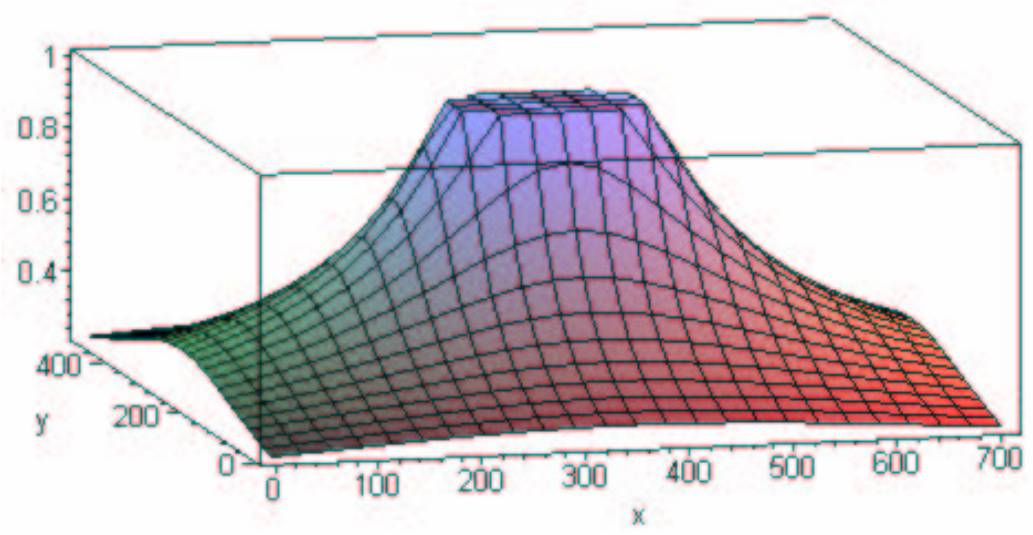

Figure 2.3: Visual sensitivity 3D map (704x480), RW size – (200x100), RW center -(352,240).

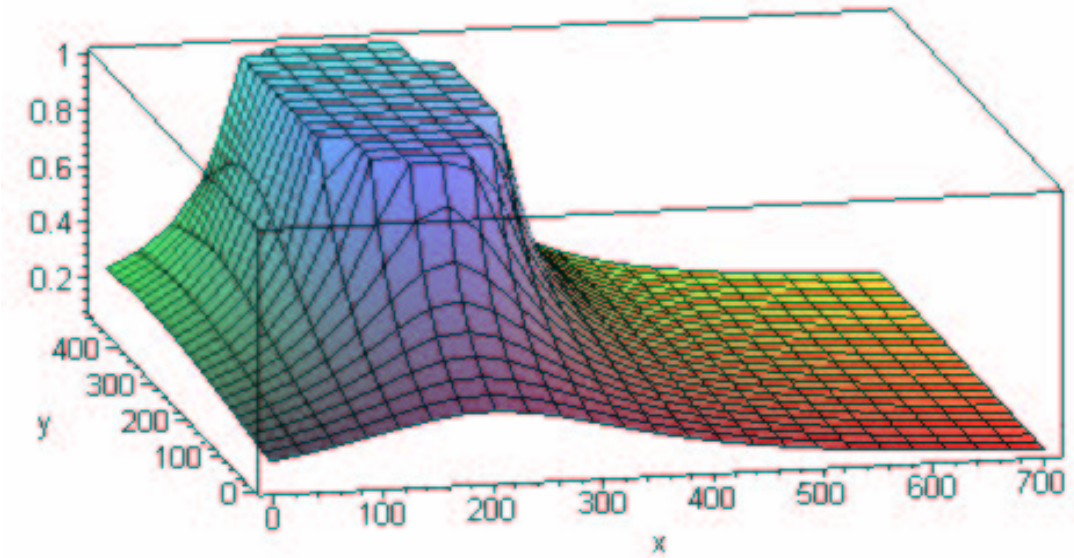

Figure 2.4: Visual sensitivity 3D map (704x480), RW size – (100x200), RW center –(200,350).

$$S(x,y) = \frac{1}{1 + k_{ECC} \cdot \frac{180}{\pi} \arctan\left(\frac{\sqrt{\left(\frac{x - x_C}{V_x(t)/V_y(t)} \right)^2 + (y - y_C)^2} - V_y(t) \cdot T_d}{VD} \right)} \quad (2.5.2)$$

Visual sensitivity is calculated for each pixel using formulas 2.3.1 and 2.5.1. As a result we would have a perceptually encoded image with the given sensitivity resolution.

Fig 2.3 shows Visual Window created by sensitivity function S(x,y), which is combined from AW and RW. RW center is at (352,240). RW dimensions are 200 pixels by 100 pixels. Fig 2.4 shows combined VW, with RW center at (200,350). RW dimensions are 100 pixels by 200 pixels.

2.6 Dynamic Reflex Window Construction

The eye velocity prediction method is described in this section. Based, on the past positional variances future eye velocity components are estimated. These velocity components are used in formula 2.5.2 for a given prediction accuracy goal.

In the proposed model, eye velocity was calculated as a path that the eye gaze traveled while the system was processing frame F(t). Having calculated the length of this path, it was possible to estimate eye velocity in pixels per frame. It should be noted that the number of eye samples that path was created from might be different for each particular

frame. Such factors as equipment sampling frequency, network delay and encoding time of a specific frame influence the amount of eye samples received at a given time.

2.6.1 Running Average Eye Velocity

Suppose there are n eye gazes detected during encoding of t-th frame. Each eye gaze $S_i(t)$ detected for the frame F(t) has (x,y) position (in units of pixels). The estimated horizontal and vertical components of the eye velocity for the frame F(t) are then calculated as:

$$\hat{V}_x(t) = \sum_{i=1}^{n-1} x(t_{i+1} - T_d) - x(t_i - T_d) | \tag{2.5.3}$$

$$\hat{V}_y(t) = \sum_{i=1}^{n-1} y(t_{i+1} - T_d) - y(t_i - T_d) | \tag{2.5.4}$$

These values are called *running average velocity* (RAV).

In a simpler explanation these velocity values represent eye movement during frame F(t). Notation $x(t_i\text{-}T_d)$ and $y(t_i\text{-}T_d)$ means that eye sample that system received at the moment "t" had been detected by eye-tracker at the moment "$t\text{-}T_d$". This eye sample was processed T_d msec later after it was detected by eye tracker. Fig 2.5 presents the concept of different types of eye gazes. In Fig 2.5 the number of delayed eye gazes is two (one eye delayed eye gaze is RW center). That means that "n" in formulas 2.5.3 and 2.5.4 is equal to 2. In real implementation the center of reflex window is placed on the last available eye-gaze. The equations for RW center would be placed at: $x(t_n - T_d)$ and $y(t_n - T_d)$. Delayed eye gazes will be represented by: $x(t_i - T_d)$ and,

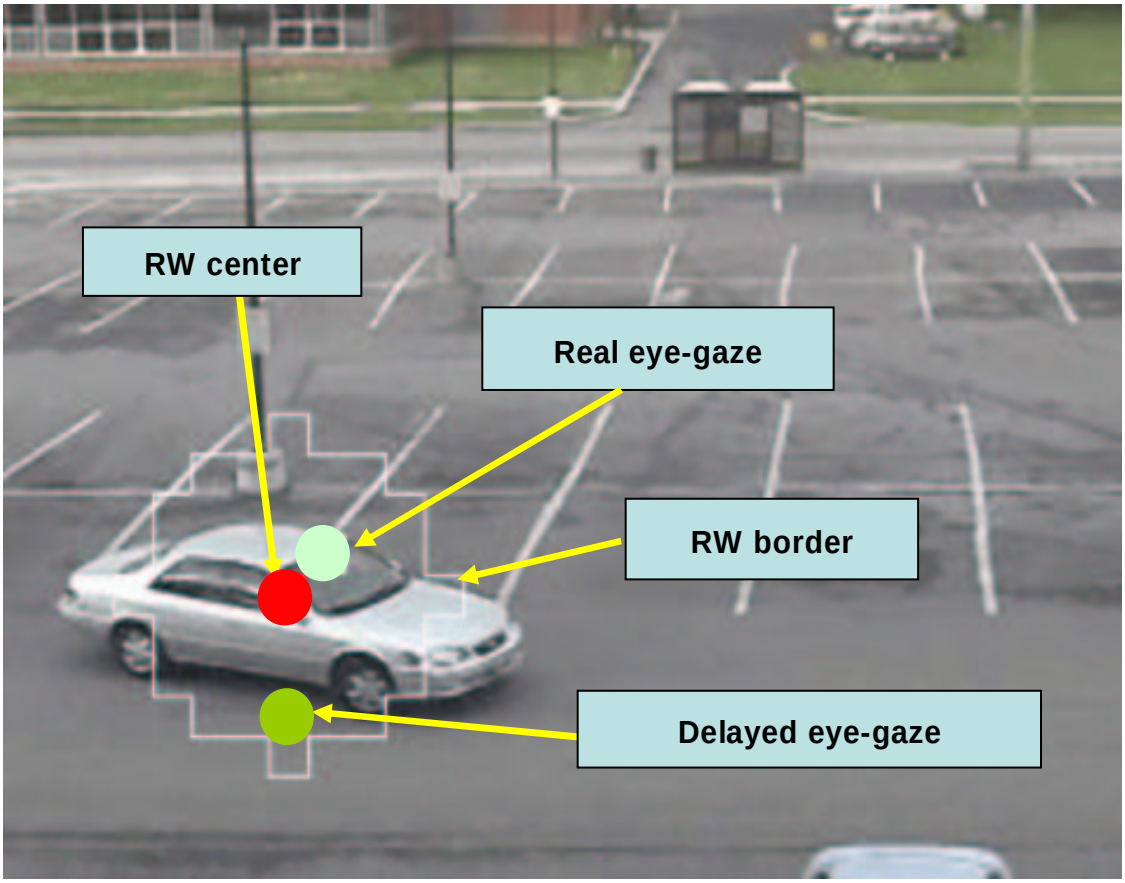

Figure 2.5: Example of Reflex Window and its components.

where $1 \le i \le n$, and n is number of eye detected eye while encoding frame $F(t)$. Real eye gazes coordinates would be those that will be detected while encoding frame F(t+T_d): $x(t_i + T_d)$ and $y(t_i + T_d)$, where $1 \le i \le n$ is number of detected eye samples for frame $F(t + T_d)$.

Values $\hat{V}_x(t)$ and $\hat{V}_y(t)$ are originally calculated as pixel values. Given that L (inches) is the distance between subject eyes and the display, H_{ph} (inches) is the horizontal image size, W_{ph} (inches) is the vertical image size, H (pixels) is the horizontal image size, and W (pixels) is the vertical image size. $\hat{V}_x(t)$ and $\hat{V}_y(t)$ are converted to angular (degrees) values:

$$V_{x\_angular}(t) = \arctan\left(\frac{H_{ph} * \hat{V}_x(t)}{L * H}\right) \tag{2.5.5}$$

$$V_{y\_angular}(t) = \arctan\left(\frac{V_{ph} * \hat{V}_y(t)}{L * W}\right) \tag{2.5.6}$$

How fast can human eyes move? Fig 2.6, Fig 2.7, Fig 2.8 show a sample of RAV measurement of a subject for different frames. Here x-axis shows frame numbers and y-axis shows the eye speed during each frame.

2.6.2 Containment Assured Eye Velocity

Knowing the point where the center of RW is placed and having eye velocity values collected, it is possible to derive the algorithm, which will calculate the size of RW. The

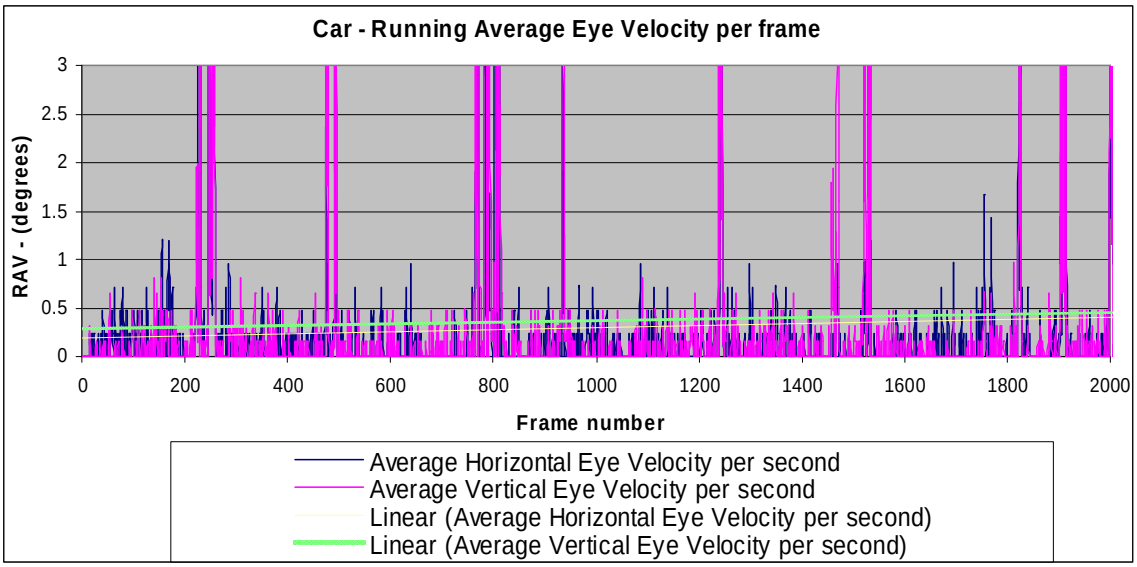

Figure 2.6: "Car". Running average angular eye velocity for every video frame.

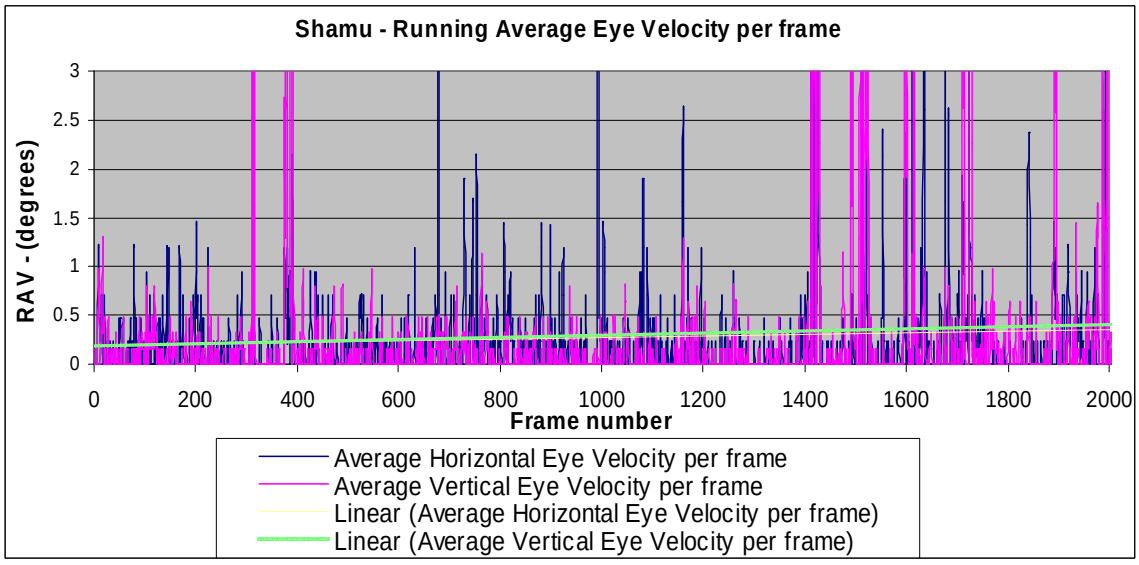

Figure 2.7: "Shamu". Running average angular eye velocity for every video frame.

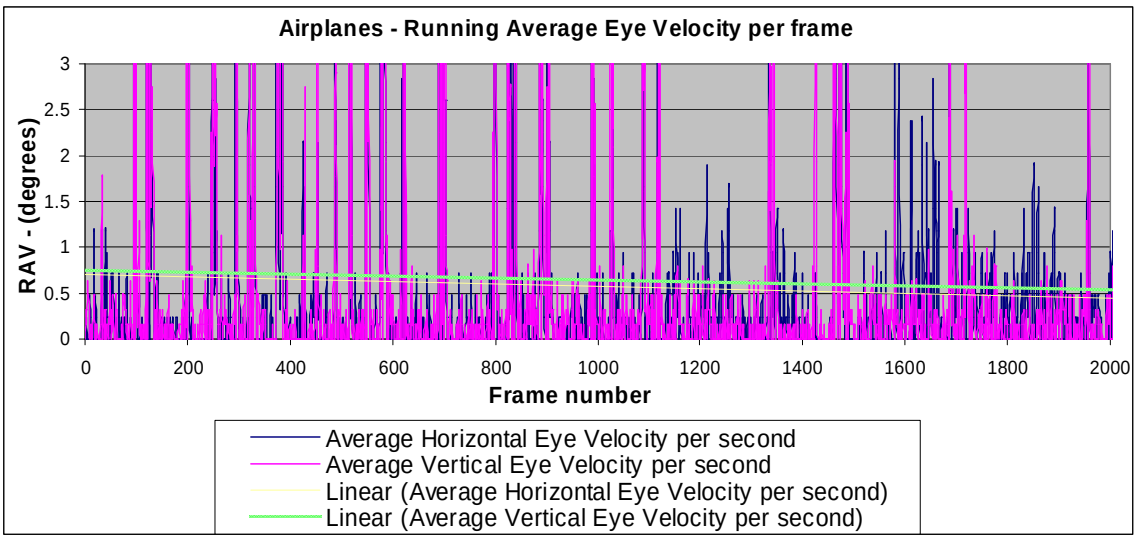

Figure 2.8: "Airplanes". Running average angular eye velocity for every video frame.

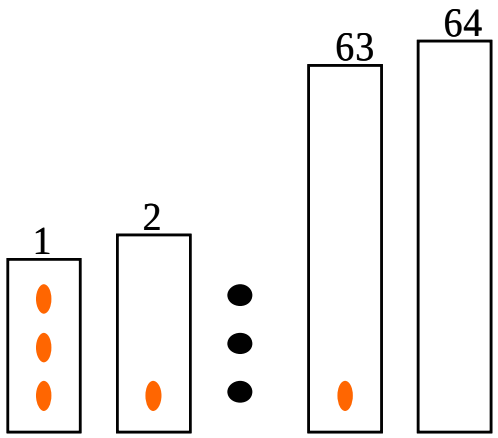

Figure 2.9: Example of velocity counters update mechanism.

idea behind the algorithm is to take into consideration velocity values over some period of time. An analogy of the eye velocity histogram is built by a set of velocity counters. The formal description is provided bellow.

There are two sets of counters. Each counter is designed to increase its value once a particular eye velocity sample is received by the system. For each arriving eye-sample $E_i(t)$ the x and y speed components are accounted separately. Eye velocity counters are presented by two sets: $C_X = \{c_{x,0},..., c_{x,W} : c_{x,i} \geq 0\}$ and $C_Y = \{c_{y,0},..., c_{y,H} : c_{y,i} \geq 0\}$. Were W and H are width and height of the video image in pixels correspondingly. Each counter $c_{x,i}(t)$ is associated with fixed RAV value $\hat{V}_x(t)$ (same goes for y component). A RAV sample $\hat{V}_x(t)$ belongs to the $c_{x,i}(t)$ counter if $\hat{V}_x(t) = i$ pixels per frame ($\hat{V}_y(t)$ belongs to $c_{y,j}(t)$ if $\hat{V}_y(t) = j$). $\hat{V}_x(t)$ and $\hat{V}_y(t)$ are integer values. Each RAV sample $\hat{V}_x(t)$ and $\hat{V}_x(t)$ updates corresponding counter. For each frame F(t) there is one horizontal $\hat{V}_x(t)$ and one vertical $\hat{V}_y(t)$ RAV sample coming. Depending on the RAV sample value the corresponding counter is incremented. In the case of $\hat{V}_x(t) = i$ and $\hat{V}_y(t) = j$

$$c_{x,i} = c_{x,i} + 1 \text{ and } c_{y,j} = c_{y,j} + 1 \qquad (2.5.7)$$

RAVs history limit is taken into consideration, where any samples older than “k” frame units (this value is referenced as RAVs velocity values (VV) in the graphs) are discarded. That means that corresponding counters for the RAVs that system had for

frame F(t-k) are decreased by one. This is accomplished by setting up a circular queue and count update as following: assuming that RAVs values for frame F(t-k) were $\hat{V}_x(t-k) = i$ and $\hat{V}_y(t-k) = j$ then RAVs counter reduction will look like this:

$$c_{x,i} = c_{x,i} - 1 \quad \text{and} \quad c_{y,j} = c_{y,j} - 1 \tag{2.5.8}$$

ϖ is *target gaze containment parameter*. ϖ corresponds to the amount of the eye gazes to be contained within RW. $\varpi \in (0,..,1]$. For example $\varpi = 0.8$ would mean that 80% of gases should be contained within the RW.

To calculate required eye velocity following inequalities should be solved.

$$\sum_{i=1}^{\max(m)} x_i \le \varpi \sum_{i=1}^{W} x_i \tag{2.5.9}$$

Then future horizontal eye velocity would be $V_x(t) = m$.

Similarly,

$$\sum_{i=1}^{\max(n)} y_i \le \varpi \sum_{i=1}^{H} y_i \tag{2.5.10}$$

Then $V_y(t) = n$.

$V_x(t)$ and $V_y(t)$ are called *containment assured velocity* (CAV). A Reflex Window constructed by CAVs will assure target gaze containment for the future eye gazes.

Fig 2.9 illustrates the process with an example. Suppose the image size is W=64 and H=64. Let's consider set C_x only. After RAV sample $\hat{V}_x(y)$ arrives, $C_{x,1}$ is going to be increased by one if $\hat{V}_x(t)=1$ pixel. X_2 is going to be increased by one if $\hat{V}_x(t)=2$. Similarly X_{64} is going to be increased by one if $\hat{V}_x(t)=64$. Let's say now 5 RAVs arrive with corresponding values of: 1, 1, 2, 1, 63 (pixels/frame). Let's assume that the target containment is 80% ($\varpi = 0.8$). Then, maximum value for m, which can be derived from equation 2.5.9 would be equal to 2. That means that next frame in the video stream is encoded with assumption that eye velocity is going to be 2 pixels per frame for duration of that frame.

CHAPTER 3

Transcoder System: MPEG-2 Full Logic Transcoding

Once the visual window is obtained, it is applied to the target media. In our MPEG-2 example we have developed foveal rate controller transcoder that operates as a piecewise constant rate (PCR) controller similar to TM-5. It works in four modes. In normal mode it operates in PCR mode with a carryover. In cases of extreme congestion, it can retract to GOP wise PCR. Also, its region-based perceptual encoding operation can be switched on and off. Unlike compressed domain transcoding schemes, the described transcoder is a full logic transcoder with motion vector inference. Compressed domain transcoders are fast though they suffer from inter frame drift within GOP due to the accumulation of reference error in the predictions of "I" frames. This is particularly problematic for region based encoding, as it deliberately takes some bits from "I" references. In this thesis a full decoder and a transformation matching motion vector inferencing encoder is employed. The motion vector inferencing encoder avoids a very computation intensive motion vector (MV) estimation process. Instead of recalculating MVs, it reads the motion vector matrix from the incoming stream for each video frame. The original motion vectors however, cannot be used for the coding of the re-encoded frames. The nature of MV recalculation depends on the nature of frame transformation. Each transcoder transformation is thus designed as a pair-wise function/algorithm $\{T^{F}(), \text{ and } T^{MV}()\}$, where $F_{out}=T^{F}(F_{in})$, and $MV_{out=}T^{MV}(MV_{in})$. The inference MV matrix is used before the

prediction in the re-encoding stage. The predictions are freshly recalculated in the encoding stage, thereby avoiding any drift due to re-encoding. At the same time it avoids the costly MV search, allowing an increase in transcoding speed.

3.1 MPEG-2 Bit Rate Control Mechanism

MPEG-2 employs a complex double feedback based transcoder rate control mechanism for rate control. An MPEG-spatial fidelity control scheme that is fully compatible with the widely used TM-5 model was developed as a result of this work. It can generate the piecewise constant bit-rate (CBR). Our enhancement enables it to perform additional spatial bit-allocation. However, the overall bit-rate still maintains the CBR per GOP basis. The generated bit-stream remains fully MPEG-2 conformant. Thus any off-the-shelf MPEG-2/MPEG-4 player can decode and play it.

Due to the *variable length coding* (VLC), it is not possible to predict the exact amount of bits needed to encode a macro-block for a given choice of coding parameters. Secondly, the perceptual content and activity in a particular region of the video frame dictates the inherent amount of bits that may be required to encode a macroblock. Also, the bit requirements per macro-block depend on the picture type (I, B or P) as well other subjective factors. The proposed scheme is a double-loop feedback control mechanism where the output bit-rate is continually sensed to determine overall piecewise constant rate, with appropriate accounting for variations in frame/picture type like TM-5. A second internal feedback loop further tracks the effectiveness of key conversion factors/constants for additional stability. The output bit-rate is controlled by the

quantization-step given by ISO/IEC 13818-2 tables [21] estimated on basis of static human visual sensitivity (HVS) analysis.

3.2 MPEG-2 Quantization Mechanism

The rate controller system has two modes of operation: *normal* mode and *frugal* mode. In normal mode, the objective for a feedback system is to maintain the output bit rate at piece-wise per GOP (group-of-picture). In frugal mode, it moves into a variable-rate encoding mode with proper proportioning for frame types, and the macro-block activity, without any carryover. The saving earned during the frugal mode, however, is stored and can be (optionally) carried over to the point where normal mode is resumed to attain overall target rate. The control mechanism maintains three virtual buffers for separate tracking of bits consumed by the I, B, and P frames. To encode a frame of type x, for each macroblock, first a quantity called buffer fullness d_j^x is determined. Then it is used to determine the modulation factor Q_j.

$$\mathrm{Q_j} = \left\lfloor \frac{31 \times e_j^x}{r} \right\rfloor \text{ where, } \mathrm{r} = \left\lfloor \frac{2 \times c(t)}{frame\_rate} + 0.5 \right\rfloor \tag{3.2.1}$$

Here, r is called *reaction parameter* and is estimated from the current overall bit rate goal c(t). The quantity e_j^x is the *effective buffer fullness* and is computed from *virtual buffer fullness* d_j^x. The notation refers to the jth macroblock inside of x type frame. These quantities are determined as following:

$$e_j^x = d_j^x - d_0^x \cdot S(t), \text{ and} \qquad (3.2.2)$$

$$d_j^x = d_0^x + B_{j-1} - \frac{(j-1)\cdot T^x(t)}{mb\_count}$$

In normal mode the *effective buffer fullness* is given by the *virtual buffer fullness*, but during frugal mode, it is decoupled from initial buffer fullness, and is only estimated based on the frugal state target bit rate. A value of 1 to the state function *S(t)* moves the system to the frugal state, and zero to normal state. In the frugal mode, the bit generation temporarily reduces. However, the virtual buffer fullness quantity is continually updated. This enables the carryover of the savings made during frugal mode operation when the system returns to normal mode.

Virtual buffer fullness is determined from three quantities: (i) the number of bits generated so far by encoding previous j-1 macroblocks inside this frame (B_{j-1}), (ii) the initial fullness of buffer before beginning the encoding of this frame (d_j^0), and (iii) the target bits allocated to this frame (T^x). The initial values for the buffer fullness are computed at the beginning of encoding a frame. For the encoding of first frame of a GOP these are given by: $d_0^I = 10 \times \frac{r}{31}$, $d_0^P = k_P \cdot d_0^I$, *and* $d_0^B = k_B \cdot d_0^I$. Here k_B and k_P are universal constants. They depend on the quantization matrices. For standard MPEG-2 quantization matrix their values are k_P =1.0 and k_B =1.4. For subsequent frames the final fullness of the previous frame is passed on as the initial fullness of the next frame buffers.

During the frame encoding process, the number of bits required for each macroblock is measured immediately after the macroblock was encoded. Once the DCT is done, all

subsequent coding procedures for current macroblock including VLC have to be completed before the next macroblock can be quantized.

3.3 Target Bit-Rate Calculation for Different Frame Types:

To calculate the target bit allocation for each frame, first a rough bit allocation for the entire GOP is done at the beginning of each GOP. This estimation is done from the stream target bit rate, the frame rate, and the total number of frames in the GOP. Each GOP initially has one "I" and n_B and n_P of "B", and "P" frames respectively.

$$R_{GOP} = \left\lfloor \frac{(1 + n_{P-remaining} + n_{B-remaining}) \times c(t)}{frame\_rate} + 0.5 \right\rfloor \tag{3.3.1}$$

To account for the variations in the frame types complexities, a TM-5 like adjustment is made. This is performed with the quantities called *global complexity measures* [X_I: X_P: X_B]. These are computed by averaging the actual quantization values used during the encoding of all the macroblocks including the skipped ones) and the actual number of bits generated S_X, where $X_X = S_X \cdot Q_X$. These averages are maintained for each frame type (x=I, P, and B) and updated at the end of the encoding of the each frame. Finally, the actual target bit-rate for each frame type is computed using the following usual TM-5 models (where k's is a pre-defined constants):

$$T^I(t) = \left\lfloor \frac{R(t)}{1 + \frac{n_P \cdot X_P}{k_P \cdot X_I} + \frac{n_B \cdot X_B}{k_B \cdot X_I}} + 0.5 \right\rfloor \tag{3.3.2}$$

$$T^{P}(t) = \left\lfloor \frac{R(t)}{n_P + \frac{n_B \cdot k_P \cdot X_B}{k_B \cdot X_P}} + 0.5 \right\rfloor \tag{3.3.3}$$

$$T^{B}(t) = \left\lfloor \frac{R(t)}{n_B + \frac{n_P \cdot k_B \cdot X_P}{k_P \cdot X_B}} + 0.5 \right\rfloor \tag{3.3.4}$$

Once each frame is encoded the number of bits used is measured and the encoded frame is subtracted from the initial GOP size ($R_{new} = R - S_X$) to estimate the remaining available bits. Also, the number of frames n_B or n_P gradually decreases. The target size for subsequent frames in the GOP, which are either type P or B, are estimated from the remaining bits R, and the remaining number of frames. Finally $\mathrm{Q_j} = \left[31 \times d_j^x \cdot r^{-1}\right]$ is computed by dividing the buffer-fullness by the TM-5 *reaction parameter*. When the system is in the normal mode, the rate control mechanism does not need to sense the target bit rate at every frame. However, when system moves into frugal mode it senses the current target-rate on per-frame basis.

3.4 Eye Acuity to Bit-Rate Mapping

At the top level, finally the quantization factor *mquant* for each macroblock is calculated as a product of two primary factors (a) the *buffer fullness* and (b) the *macroblock activity*. The *mquant* for the jth frame is computed as a product of two parameters: $mquant_j = Q_j \times a_j$. The final value of $mquant_j$ is coded either in the slice or in the macroblock header [21]. In the original MPEG TM-5 design, the motivation behind

the *activity factor* (a_j) was that human visual perception is less sensitive to distortions in noisier textured areas and more sensitive to distortion in image areas with uniform texture. Each macroblock uses the sensitivity function as an inverse proportional modifier to the activity factor. In the first step, based on the foveal analysis we determine the sensitivity factor s_i for each macroblock and reduce macroblock activity level in inverse proportion. The first stage keeps the foveal bit-allocation at a normal value, but then proportionally reduces the parafoveal bit-allocation. That way it reduces the overall bit-rate decreasing per-frame bit-allocation. However, to bring the per-frame bit-allocation to the target level, in the second stage the overall dividend gained from para-foveal reduction is uniformly distributed to elevate all the macro-blocks, including the para-foveal macro blocks. The above equation shows the activity function.

$$a_i^{fov} = \frac{a_i}{s_i}.2^{\,sum \sum_{all\ macroblocks} \frac{\log s_i}{n}} \qquad (3.4.1)$$

The remaining scheme works similar to TM-5. There are 31 quantization levels to control the amount of bits allocated for each macroblock. These 31 levels will be distributed accordingly to stimulation in the acuity model represented above. Highest quantization level is used for highest acuity point, the lowest level is used the lowest acuity point. Remaining levels are distributed between the highest and lowest ones according to a pre- defined acuity function. This scheme provides the greatest amount of bits possible for the most sensitive acuity area and the smallest amount of bits for the area which does not need high visual quality.

CHAPTER 4

Experiment

4.1 System Setup

The proposed system was implemented with an integrated Applied Science Laboratories High speed Eye tracker Model 501 [3]. The system had the following characteristics: the eye position video capturing camera had working sampling frequency of 120 samples per second. It had a rated precision of 0.5 degree. Its accuracy (spatial error between true eye position and computed measurement) was 1 degree. Errors could increase to less than 2 degrees in the periphery of the visual field. Its allowable eye movement along the horizontal axis was 50 degrees or more and along the vertical axis, is 35 degrees or more depending on optic placement and eyelids. The field was generally oval in shape). The eye position data output was averaged over 10 eye position fields. An eye fixation was defined as a mean of X and Y eye position coordinates measured over a minimum period of time, during which the eye did not move more than some minimum displacement. For this experiment it was assumed the minimum time period was 100msec, and the minimum displacement was about 1 degree per second.

The subject was accustomed to all the videos before the eye data was gathered and processed by the proposed algorithm. All three videos were 720x480 pixels and were captured with a Sony TRV20 digital camera at high resolution, (and more than 500 lines at a frame rate of 30 frames per second). The number of frames per GOP was 15. Number

of “B” frames between any given two “P” frames was two. Each video was projected on the wide-screen in the dark room. The physical dimensions of the image were: width 60 inches, height 50 inches. The distance between the subject’s eyes and the surface of the screen was 100 inches.

A particularly challenging aspect of experimentation with perceptual system is the difficulty of modeling the subjective aspects of the human interaction. There is no agreed method. In this experiment, a set of objective parameters was designed as an attempt to provide such measurement: containment factor, goodness of containment, perceptual coverage. To understand the impact of subjective characteristics of human perception system the experiments on a three carefully selected test videos, each offering various subjective challenges were performed.

4.2 Gaze Containment

The first experiment conducted was to determine how effectively eye gazes are contained within the reflex window. To show that, the quantity called gaze containment was defined. E(t) represents the set of real eye gazes for the frame F(t). According to the proposed encoding scheme RW(t) is constructed using $E(t-T_d-k)$, …. ,$E(t-T_d)$. Constant “k” represents the number of the latest RAV samples that the algorithm uses for RW(t) construction. $E^W(t) \subseteq E(t)$ is the eye gazes subset contained within RW(t).

Gaze containment is the fraction of eye gazes successfully contained within the reflex window:

$$\xi(t) = \frac{\left|\mathrm{E}^{\mathrm{RW}}(t)\right|}{\left|E(t)\right|} \tag{4.2.1}$$

In the experiment, the subject was watching the video while the eye-tracker was colleting the eye information. After that, the eye gaze log file was supplied to the analyzing software that calculated the reflex window for different feedback delay values. In this experiment, the target gaze containment parameter was set to ω=0.9 (90%).

To plot the results of this experiment gaze containment was averaged over every thirty frames. That presented a more general picture of the system's performance. The formula for plotting was derived from equation 4.2.1:

$$\xi_{AV}(k) = \frac{1}{30}\sum_{i=1}^{30}\xi(i) \tag{4.2.2}$$

$\xi_{AV}(k)$ is averaged gaze containment over one second. Where $\xi(i)$ is a gaze containment for frame F(i). F(i) represents frame number "i" for the second "k".

Fig 4.1, Fig 4.2, Fig 4.3 and Fig 4.4, Fig 4.5, Fig 4.6 plot the results for $\xi_{AV}(k)$. As evident in most of the graphs, reflex window algorithm was able to contain 100% in many cases. The containment results for two feedback delays of 166 msec and 1000 msec and two cases for number of RAV samples were considered. In the graphs the x axis represents encoding time line and y axis represents the amount of eye gazed contained within RW during presentation time. It should be noted that the number of eye gazes detected for each second of play time might

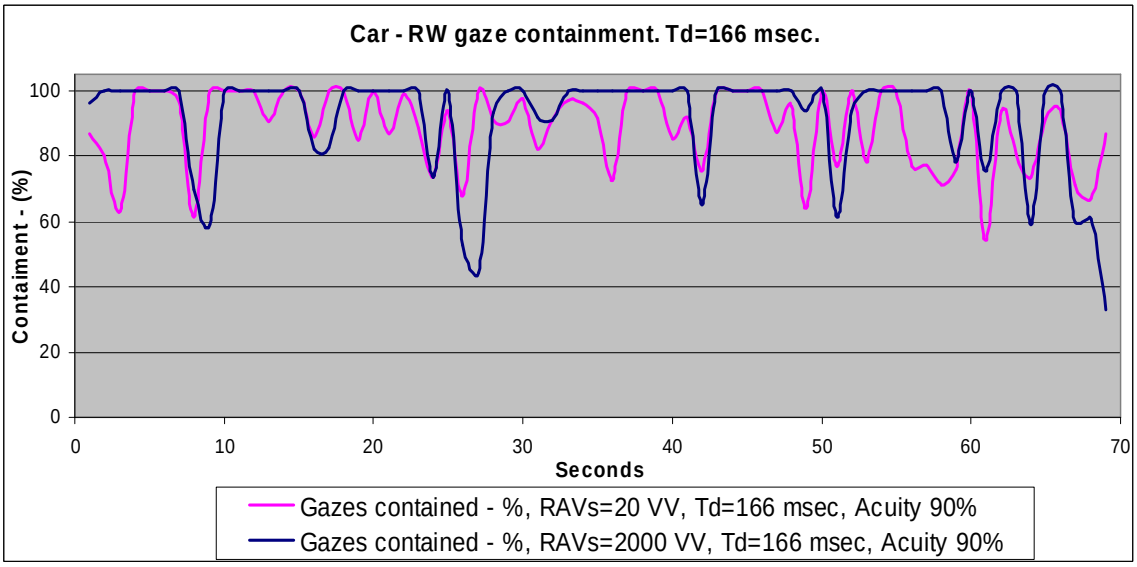

Figure 4.1: "Car". Percentage of the eye gazes contained for 166 msec delay scenario and two different schemes, where 20 RAV samples and 2000 RAV samples are considered.

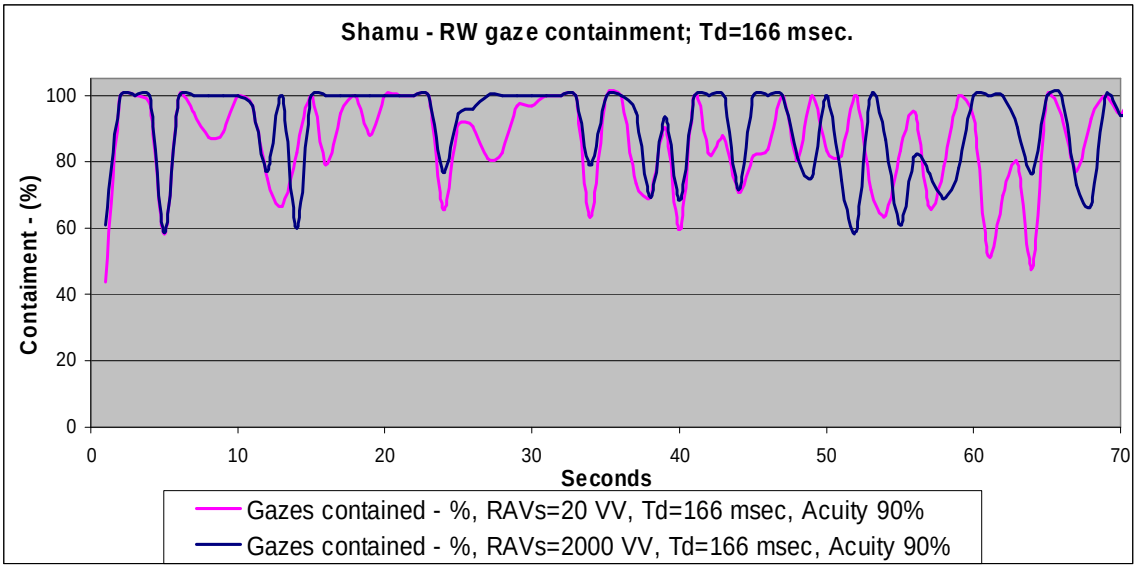

Figure 4.2: "Shamu". Percentage of the eye gazes contained for 166 msec delay scenario and two different schemes, where 20 RAV samples and 2000 RAV samples are considered.

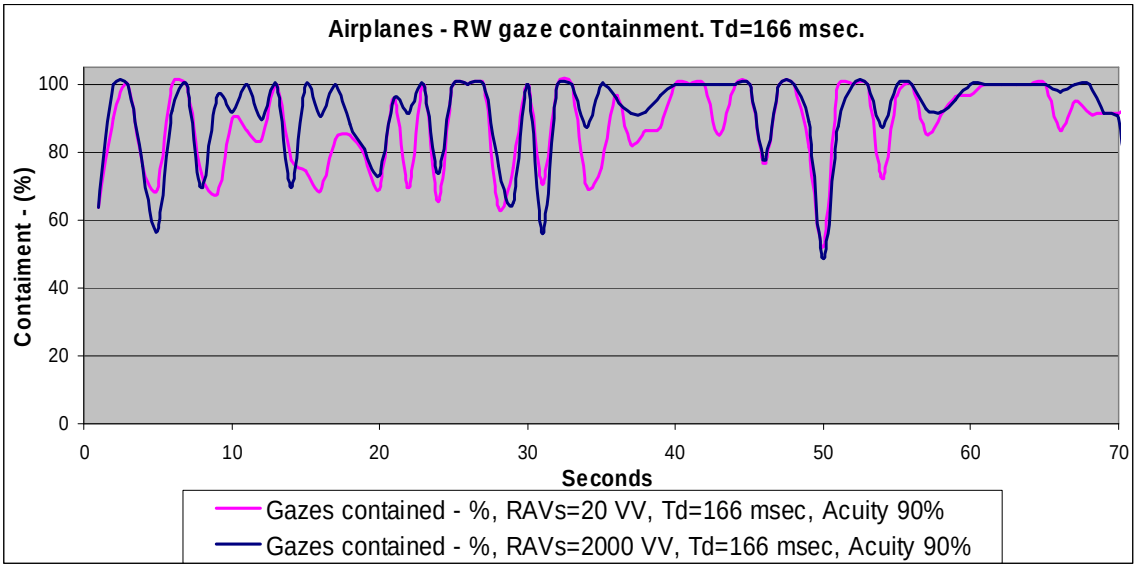

Figure 4.3: "Airplanes". Percentage of the eye gazes contained for 166 msec delay scenario and two different schemes, where 20 RAV samples and 2000 RAV samples are considered.

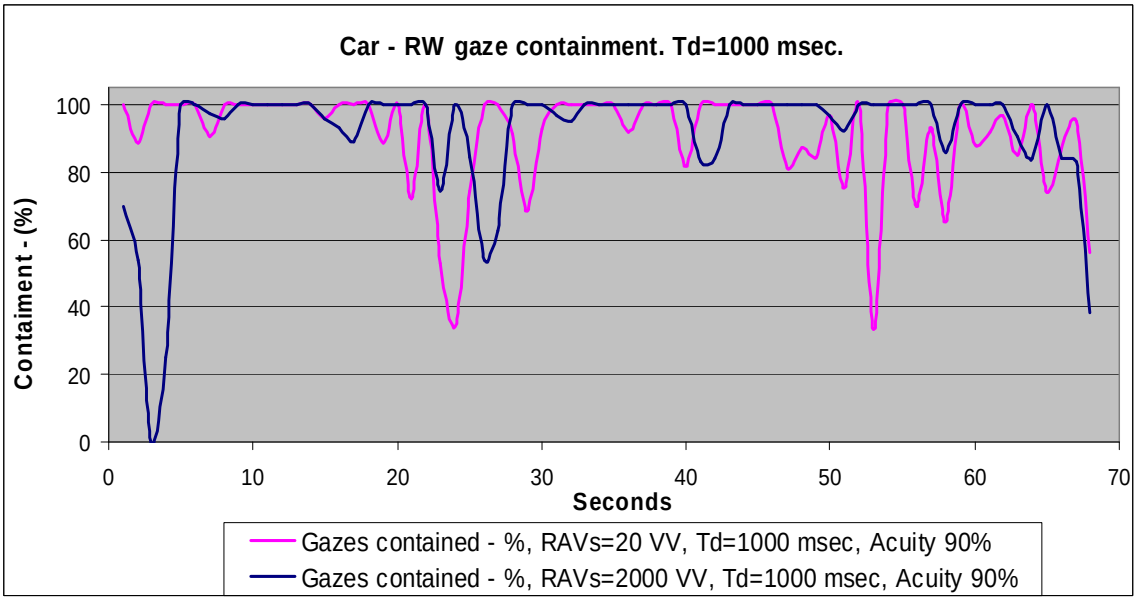

Figure 4.4: "Car". Percentage of the eye gazes contained for 1000 msec delay scenario and two different schemes, where 20 RAV samples and 2000 RAV samples are considered.

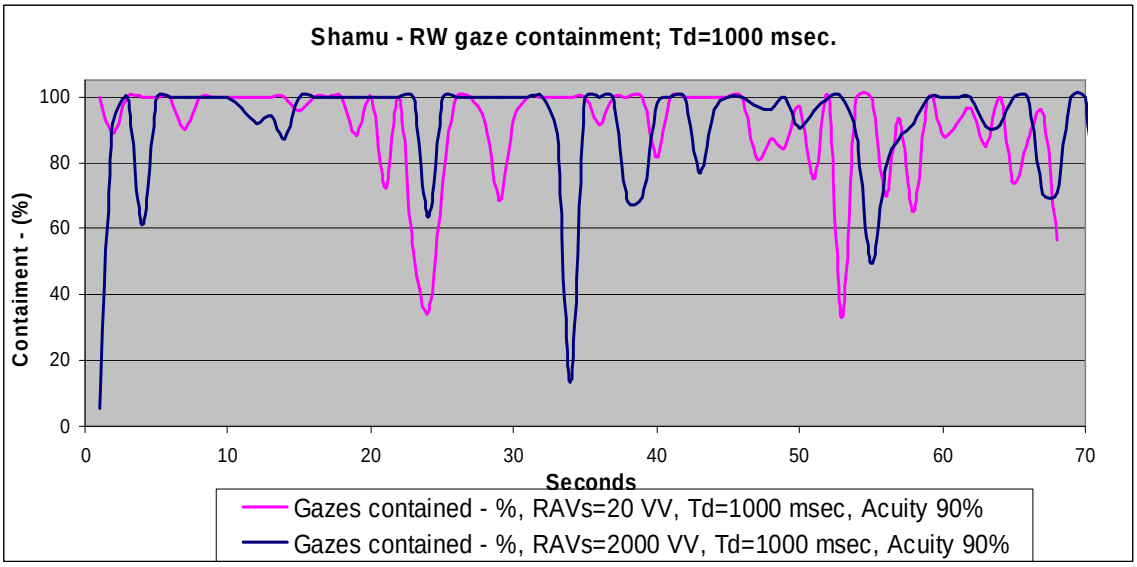

Figure 4.5: "Shamu". Percentage of the eye gazes contained for 1000 msec delay scenario and two different schemes, where 20 RAV samples and 2000 RAV samples are considered.

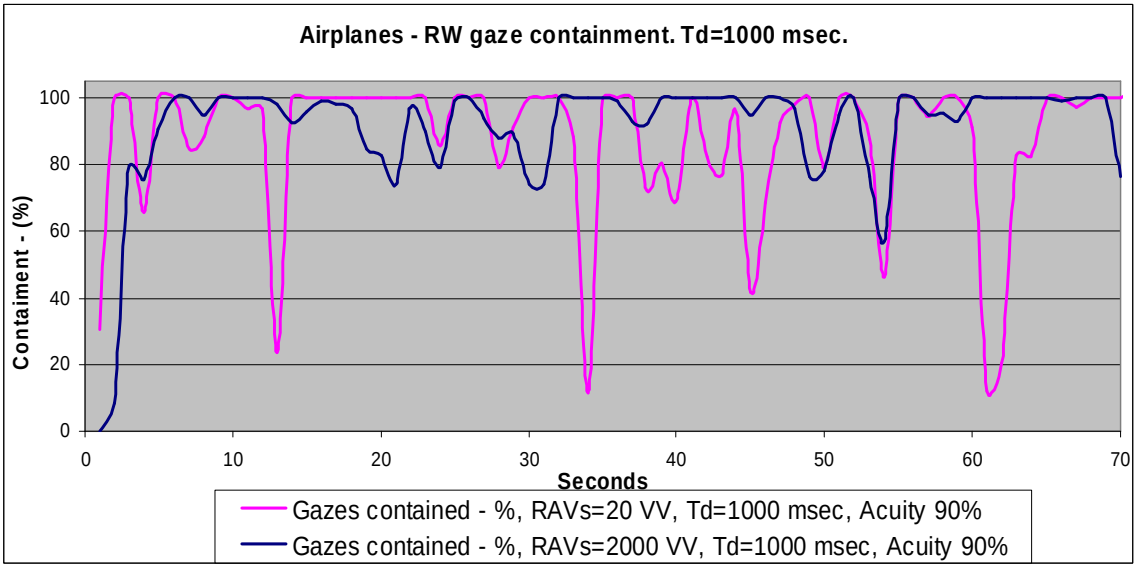

Figure 4.6: "Airplanes". Percentage of the eye gazes contained for 100 msec delay scenario and two different schemes, where 20 RAV samples and 2000 RAV samples are considered.

be different. This number depends on the capturing mode of the video camera, delay in the equipment and the network. It is possible to see from the graphs that proposed system performed very well showing 90% gaze containment on average. A lot of times gaze containment was 100% and it rarely dropped bellow 60%.

4.3 Eye Deviation for Reflex Window

While Fig 4.1, Fig 4.2, Fig 4.3 and Fig 4.4, Fig 4.5, Fig 4.6 show the hit and misses, it was further necessary to see the 'goodness' of RW construction algorithm, or how close were the hits or how far-off were the misses of the eye gazes in regard to RW. To measure that quantity called *deviation* was defined. For each eye sample $E(t_i)$, proposed method draws a line from $E(t_i)$ to the RW(t) center $m_{RW}(t)$. $p(t_i)$ is a point created by intersection of the line $(E(t_i), m_{RW}(t))$ with RW(t) boundary. The signed distance $d(p(t_i), E(t_i))$ between $p(t_i)$ and $E(t_i)$ is the deviation. The concept is explained in Fig-4.8.

$$\delta = \begin{cases} d(E(t_i), p(t_i)); & \text{if S(t) is outside RW}(t) \\ -d(E(t_i), p(t_i)); & \text{if S(t) is inside RW}(t) \end{cases} \tag{4.3.1}$$

If deviation is negative, than the gaze sample is inside the RW. If deviation is positive than the eye gaze sample is outside the RW. The magnitude identifies how far an eye gaze fell from the border. Fig 4.9, Fig 4.10, Fig 4.11 and Fig 4.12, Fig 4.13, Fig 4.14 plot the deviation variation for the test video set. As it is possible to see on these figures most of the time the eye gazes fell very close to the RW border. Throughout the experiment, the deviation remained very close to the zero line. As expected the smaller

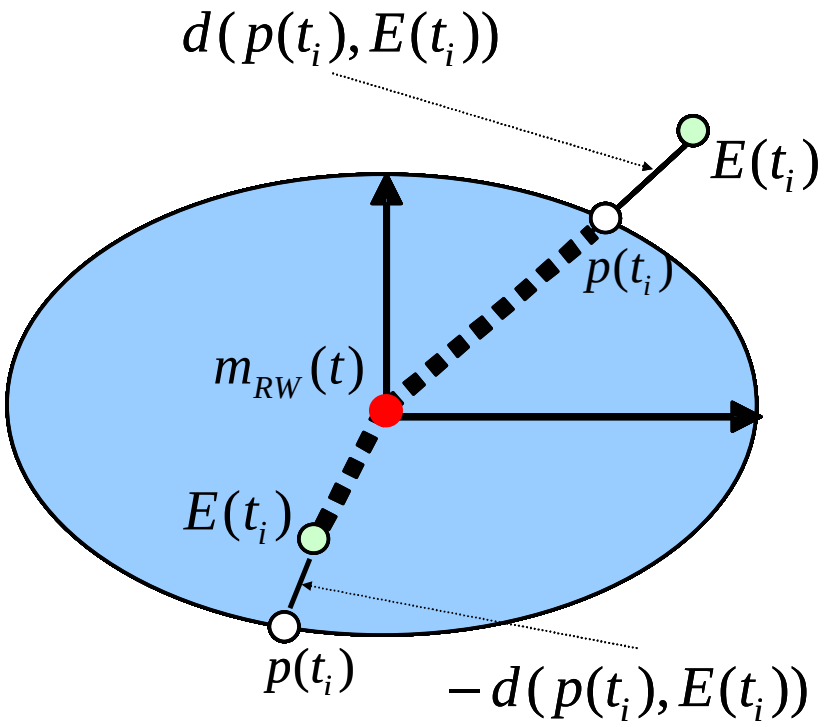

Figure 4.7: Example of deviation calculation.

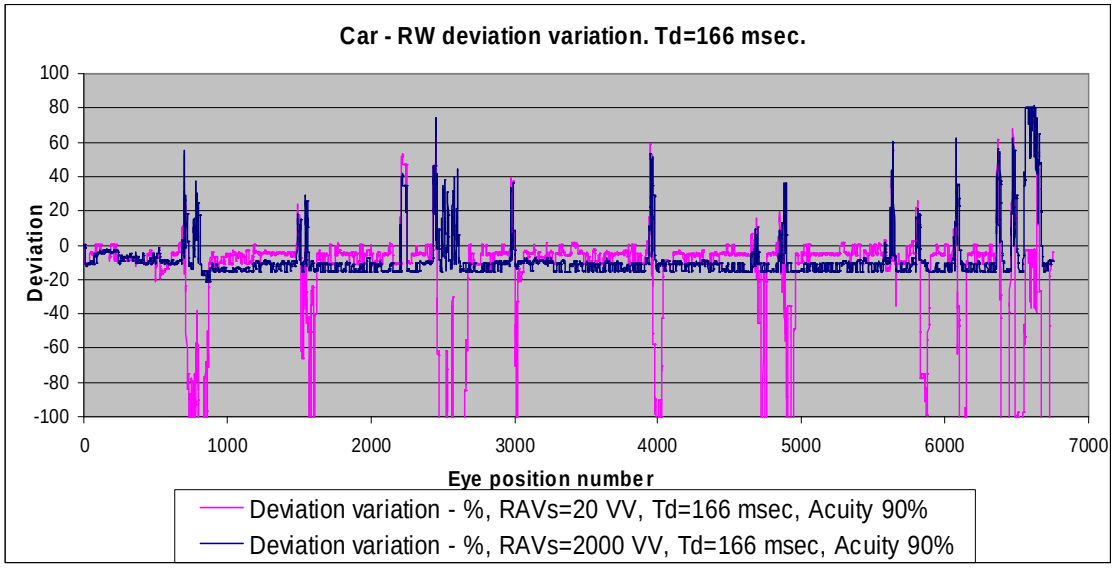

Figure 4.9: "Car". RW deviation variation for Td=166 msec and two different testing schemes, where 20 vs. 2000 RAV samples are considered correspondingly.

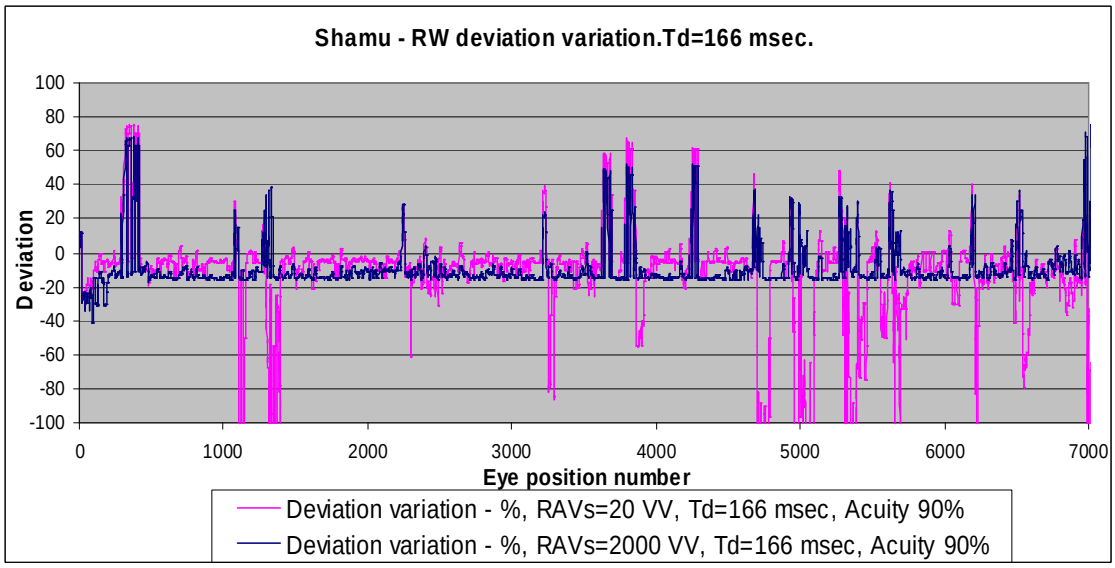

Figure 4.10: "Shamu". RW deviation variation for Td=166 msec and two different testing schemes, where 20 vs. 2000 RAV samples are considered correspondingly.

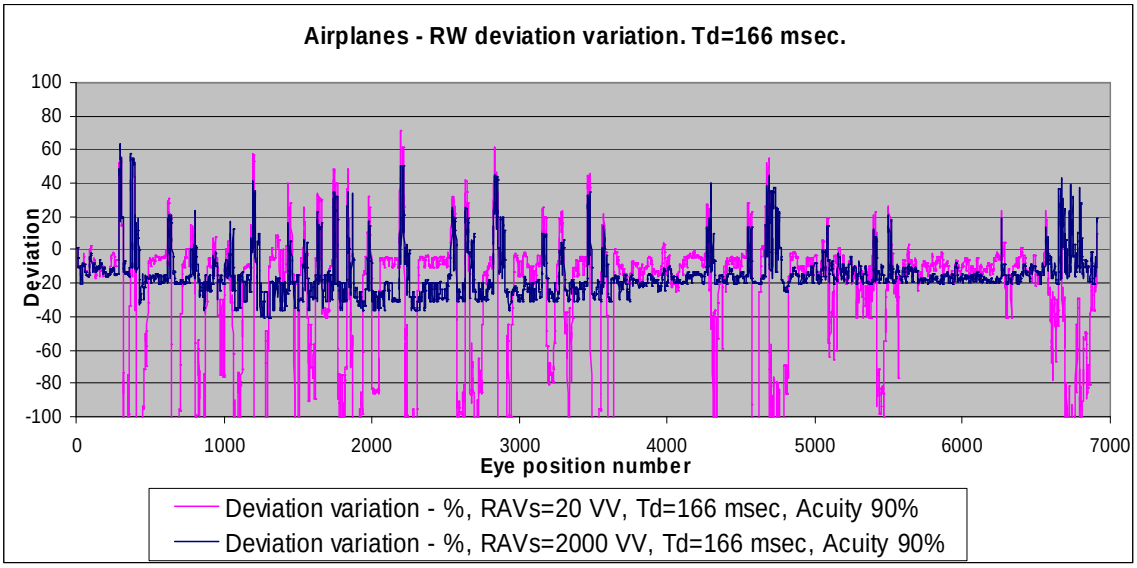

Figure 4.11: "Airplanes". RW deviation variation for Td=166 msec and two different testing schemes, where 20 vs. 2000 RAV samples are considered correspondingly.

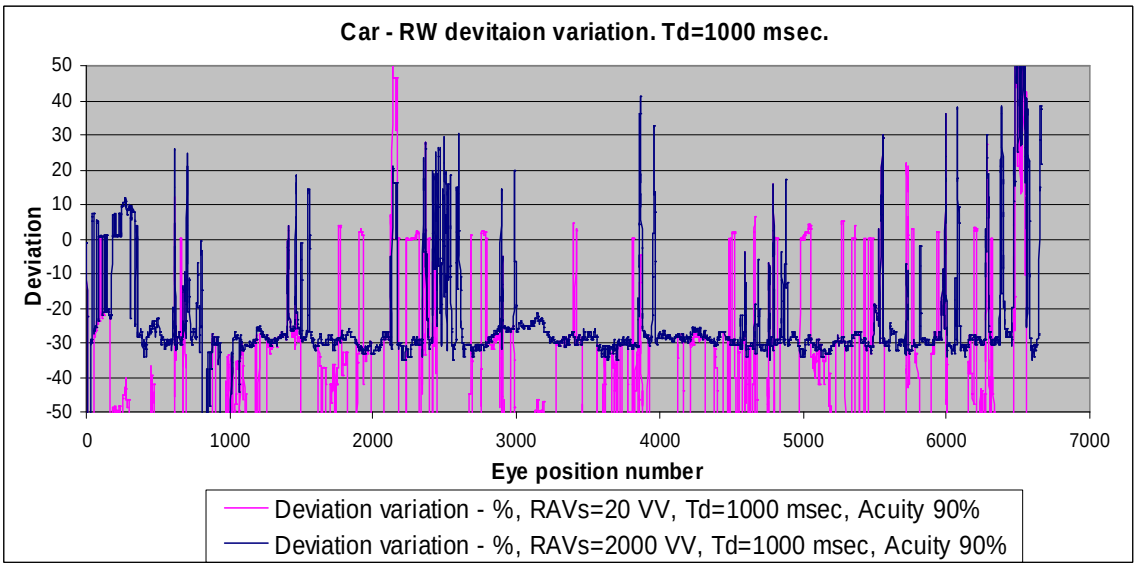

Figure 4.12: "Car". RW deviation variation for Td=1000 msec and two different testing schemes, where 20 vs. 2000 RAV samples are considered correspondingly.

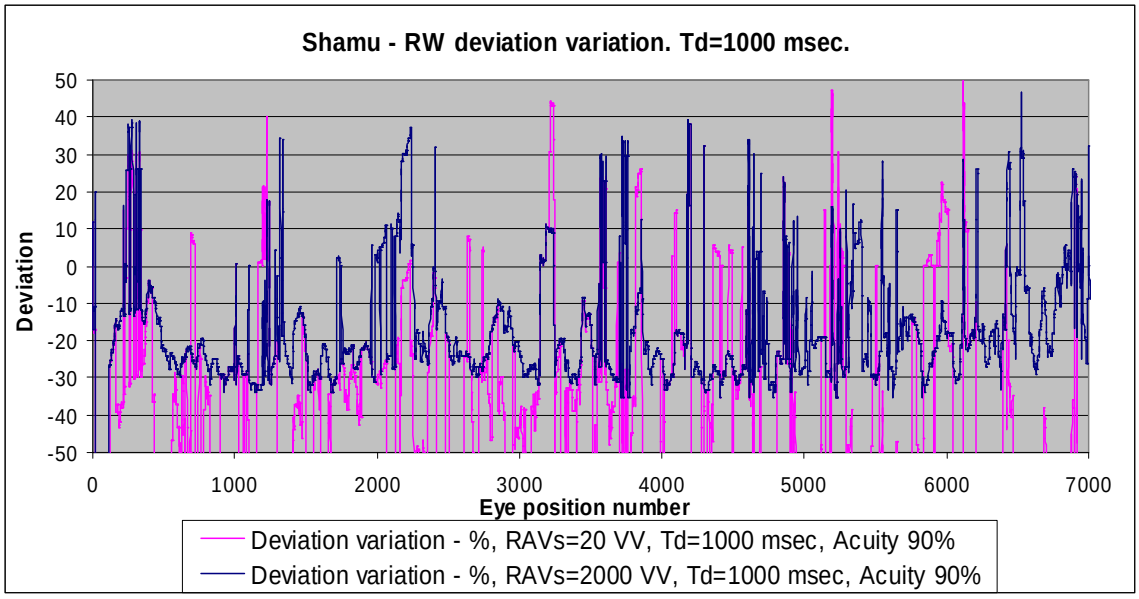

Figure 4.13: "Shamu". RW deviation variation for Td=1000 msec and two different testing schemes, where 20 vs. 2000 RAV samples are considered correspondingly.

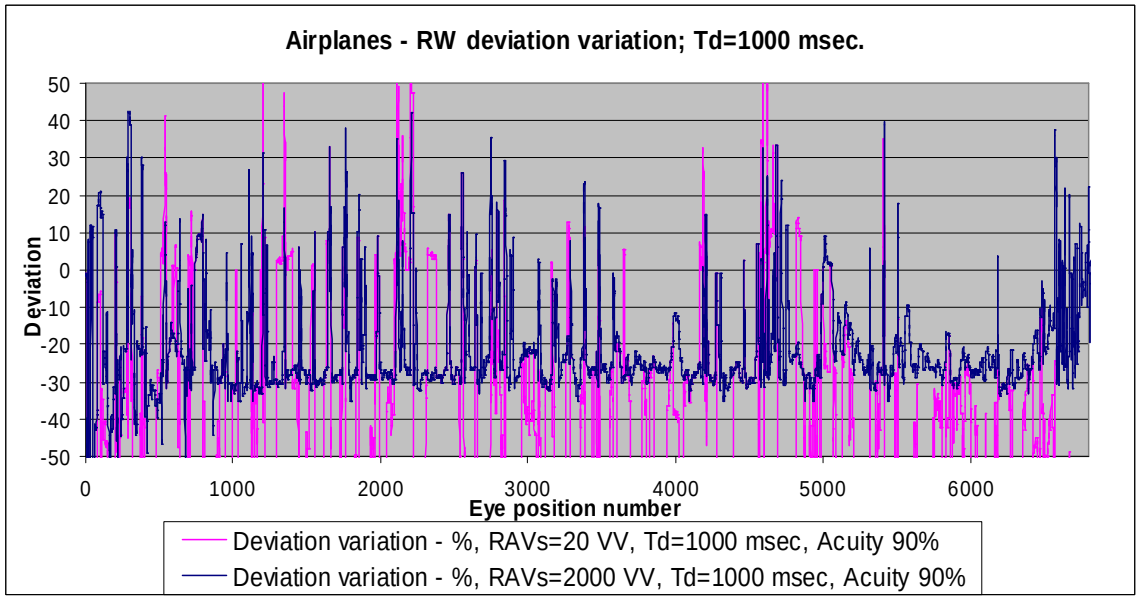

Figure 4.14: "Airplanes". RW deviation variation for Td=1000 msec and two different testing schemes, where 20 vs. 2000 RAV samples are considered correspondingly.

delay case with 166 msec feedback delay, gives good stability to the system - the RW border was on average within 20 pixels from the eye sample. It is possible to see, looking at the graphs that the proposed RW construction algorithm provides almost optimal solution for eye gaze prediction - any smaller RW could have resulted in larger number of misses.

4.4 Reflex Window Coverage Efficiency

As evident, a large window is always expected to generate better containment. For example, if a window covers the entire visual area, then all eye gazes are certainly be contained however this situation is not a desirable one as there will not be any perceptual redundancy to extract. To measure the coverage efficiency, a second performance parameter called "*perceptual coverage*" is defined.

Perceptual coverage is percentage of the image which requires coding at highest resolution.

$$\chi(t) = 100 \frac{\left|\Delta(RW(t) \cap F(t))\right|}{\left|\Delta(F(t))\right|} \qquad (4.4.1)$$

F(t) is the total viewing frame, and RW(t) is the predicted reflex window using E(t-T_d-k), …. ,E(t-T_d) eye gazes, where "k" is number of latest RAV samples that the algorithm uses for RW(t) construction. $\chi(t)$ is perceptual coverage. The intersection of the reflex window and the video frame towards the coverage is calculated to receive a more accurate result.

Fig 4.15, Fig 4.16, Fig 4.17 and Fig 4.18, Fig 4.19, Fig 4.20 respectively show the perceptual coverage for the cases corresponding for three test video set. Fig 4.15, Fig 4.16, Fig 4.17 graph show the difference between RW coverage of video frame by the algorithm, which considers 20 RAV samples vs. 2000 RAVs with 166 msec feedback delay in the system. As we can see the RW has the size of only 40% of the frame (Fig 4.18, Fig 4.19, Fig 4.20) in the case of 1000 msec delay. A system operating with about 166 ms delay would require only 5-20% of the video frame to be encoded with high resolution.

As was indicated before, a determining factor of the reflex window in the proposed algorithm is the *containment assured velocity* (CAV). It is interesting to see the CAV velocities estimated by the algorithm. Fig 4.21, Fig 4.22, Fig 4.23 and Fig 4.24, Fig 4.25, Fig 4.26 respectively plot the recorded CAV's for these cases in units of angular eye velocity. While, the smooth pursuit velocity was recorded in the range of .5-1.2 degree/sec, there were occasional fluctuations when the velocity shot up to 1 degree/sec and in some cases beyond 3 degrees/seconds. The larger fluctuations are believed to be caused by the eye-blinks of the subject during the viewing of the video. Two finer observations can be made. In both T_d=166 msec and Td=1000 msec, we can see that CAV looks like an averaged quantity on the first graph for RAVs=2000 velocity values, but for 20 RAVs it fluctuates significantly. We can also notice that for T_d =1000 msec RAVs=2000 case CAV is less than in the case of T_d = 166 msec and RAV= 2000.

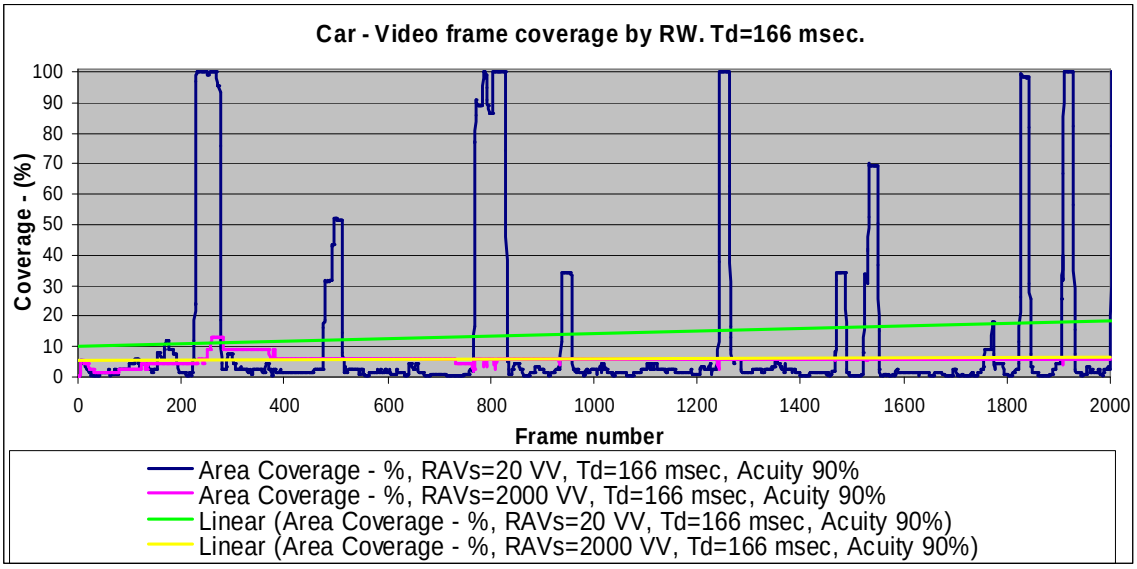

Figure 4.15: "Car". Video frame coverage by RW for Td=166 msec scenario and two different schemes, where 20 and 2000 RAV samples are considered.

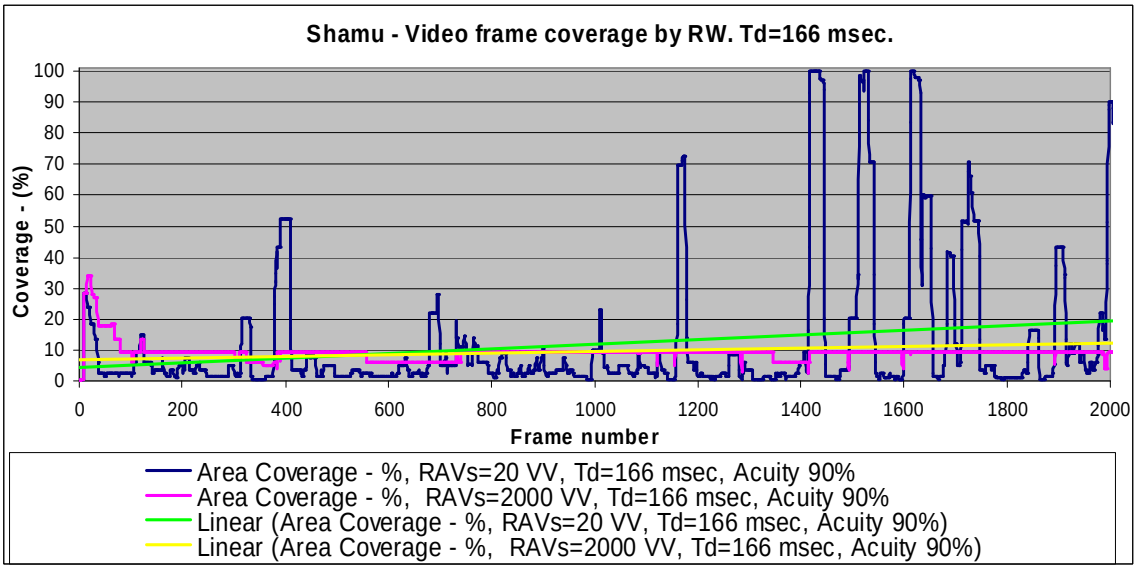

Figure 4.16: "Shamu". Video frame coverage by RW for Td=166 msec scenario and two different schemes, where 20 and 2000 RAV samples are considered.

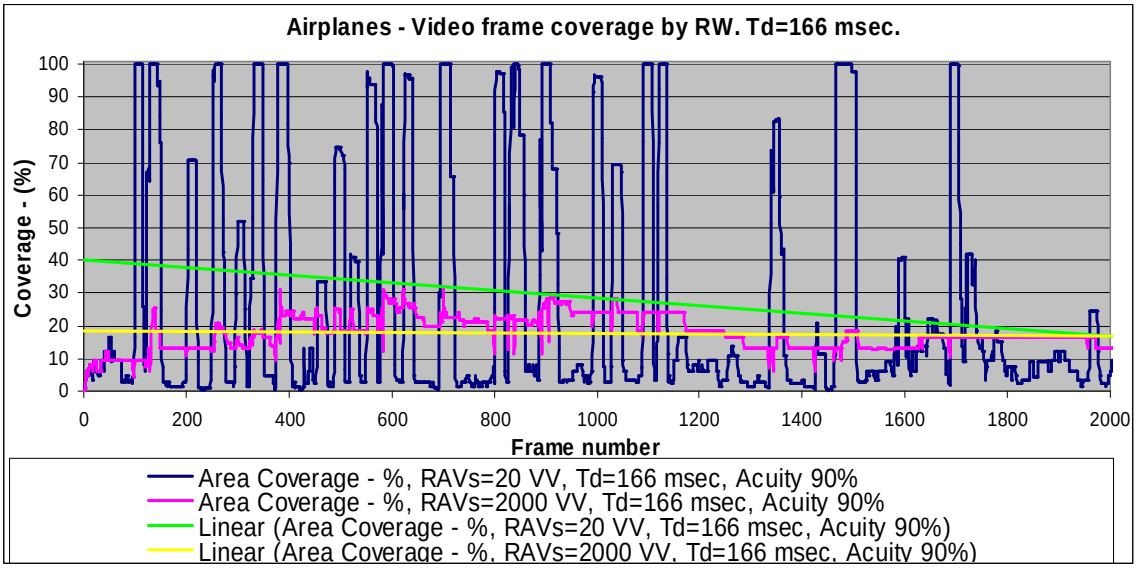

Figure 4.17: "Airplanes". Video frame coverage by RW for Td=166 msec scenario and two different schemes, where 20 and 2000 RAV samples are considered.

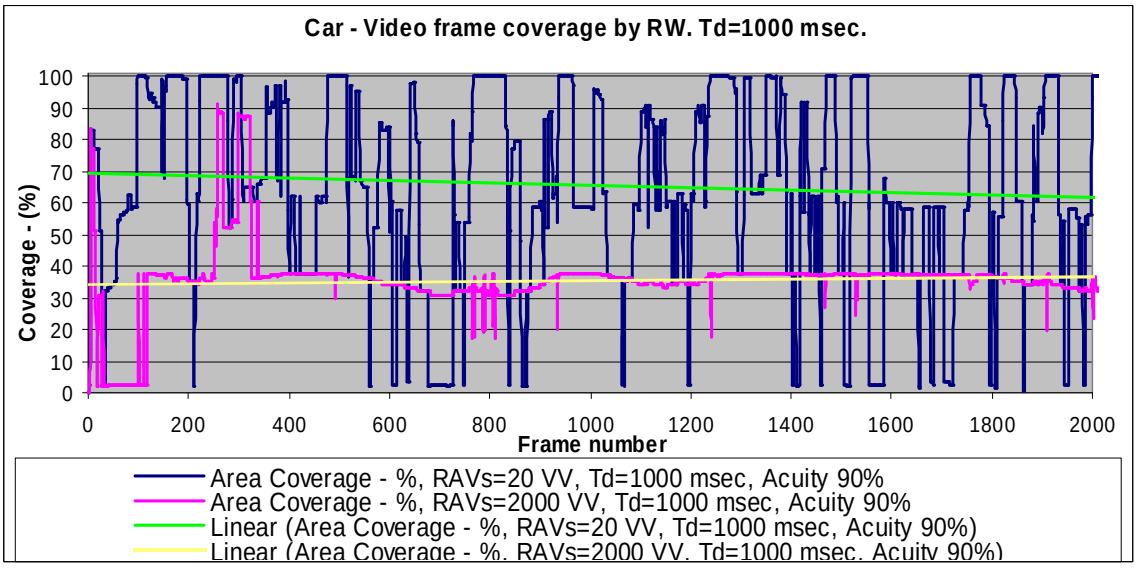

Figure 4.18: "Car". Video frame coverage by RW for Td=1000 msec scenario and two different schemes, where 20 and 2000 RAV samples are considered.

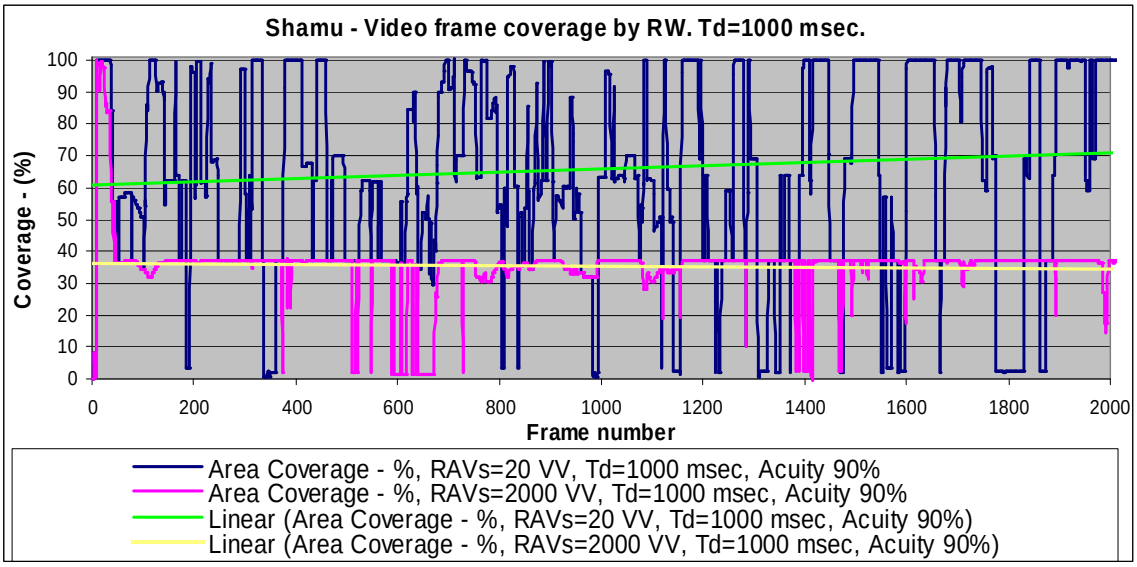

Figure 4.19: "Shamu". Video frame coverage by RW for Td=1000 msec scenario and two different schemes, where 20 and 2000 RAV samples are considered.

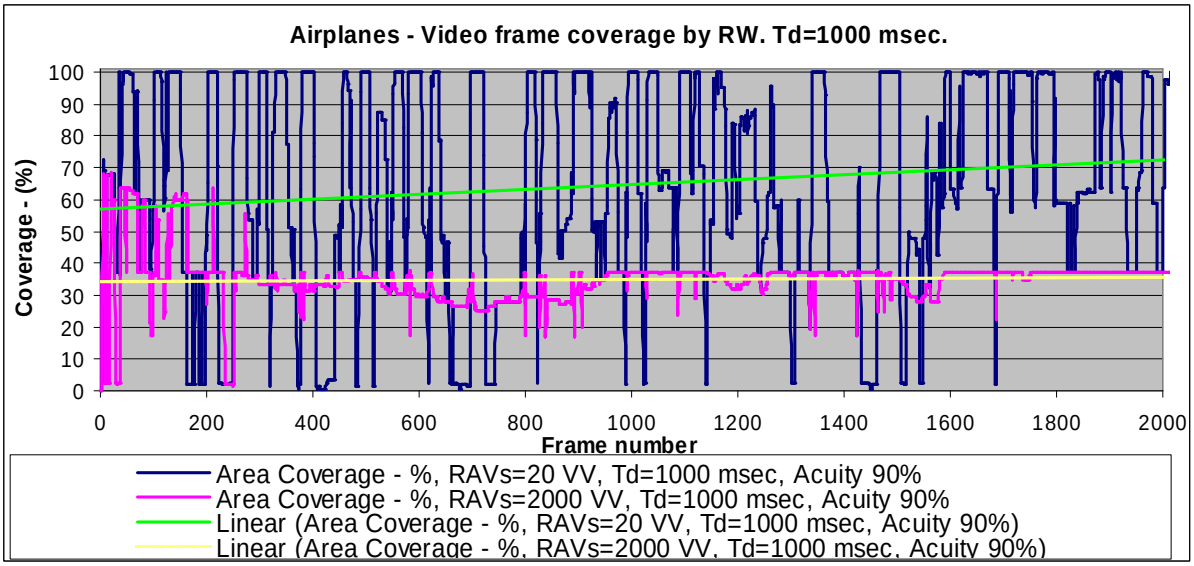

Figure 4.20: "Airplanes". Video frame coverage by RW for Td=1000 msec scenario and two different schemes, where 20 and 2000 RAV samples are considered.

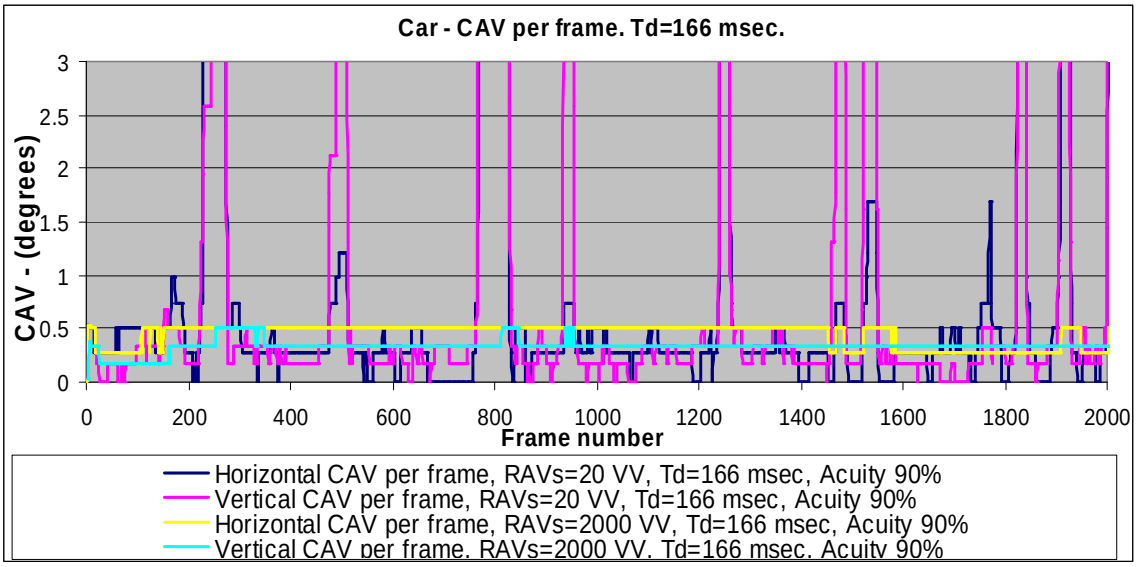

Figure 4.21: "Car". Containment Assured Velocity for Td=166 msec scenario and two different schemes, where 20 and 2000 RAV samples are considered.

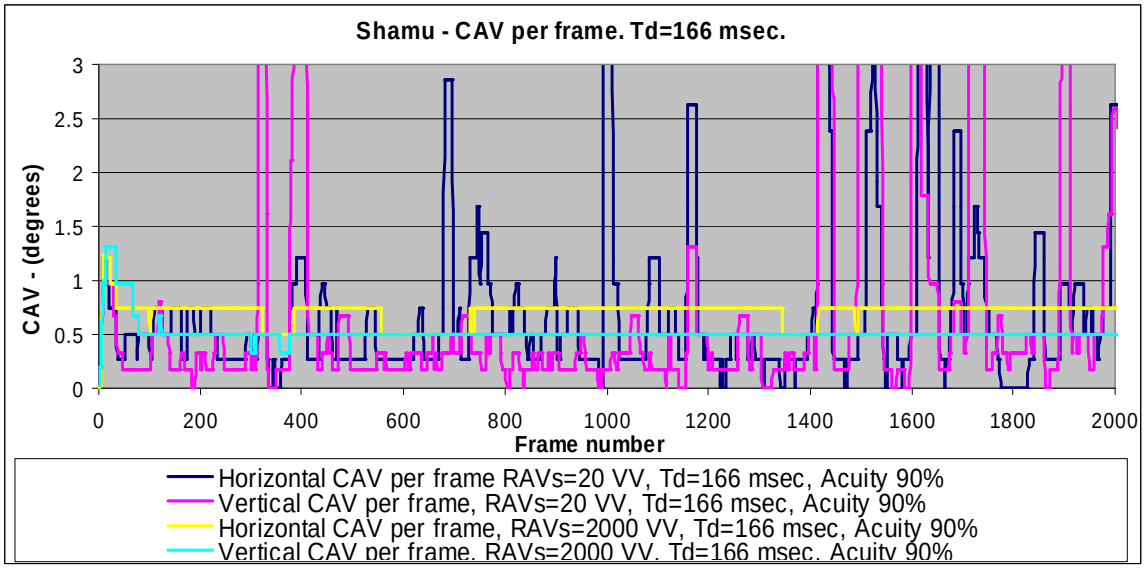

Figure 4.22: "Shamu". Containment Assured Velocity for Td=166 msec scenario and two different schemes, where 20 and 2000 RAV samples are considered.

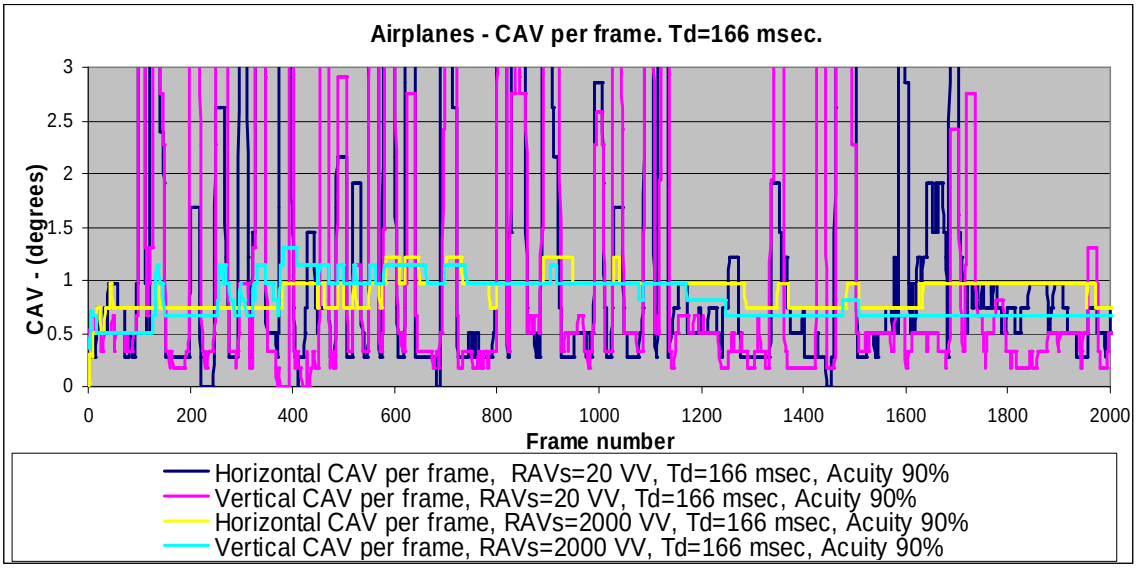

Figure 4.23: "Airplanes". Containment Assured Velocity for Td=166 msec scenario and two different schemes, where 20 and 2000 RAV samples are considered.

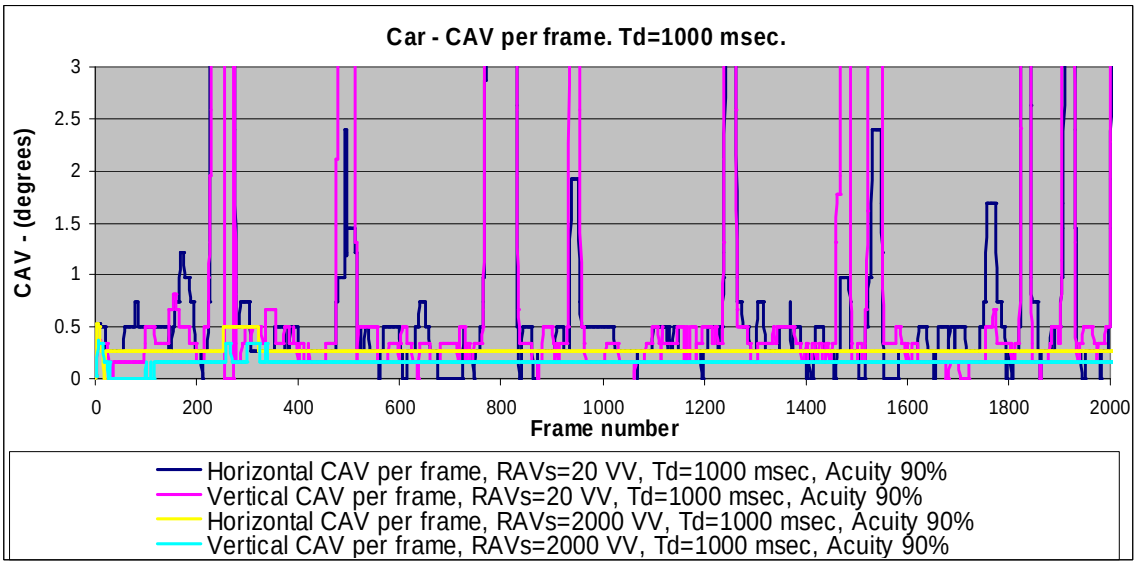

Figure 4.24: "Car". Containment Assured Velocity for Td=1000 msec scenario and two different schemes, where 20 and 2000 RAV samples are considered.

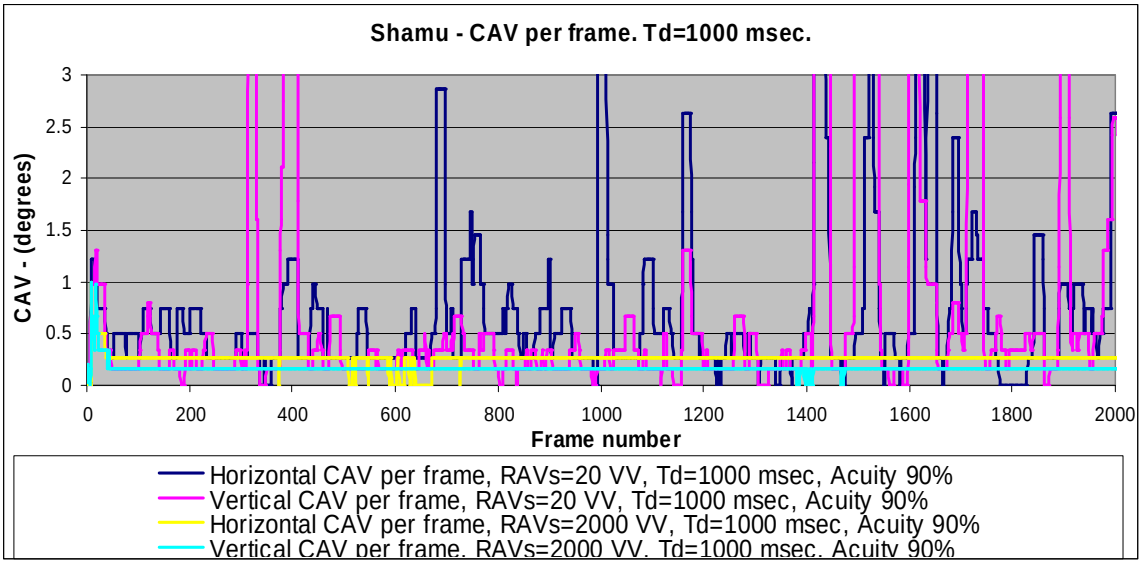

Figure 4.25: "Shamu". Containment Assured Velocity for Td=1000 msec scenario and two different schemes, where 20 and 2000 RAV samples are considered.

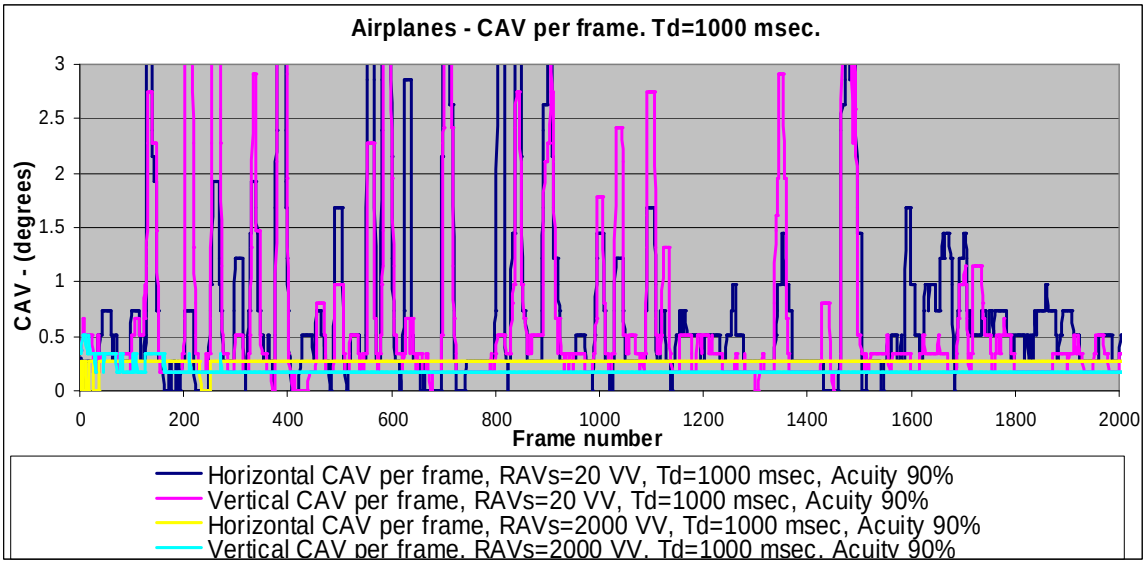

Figure 4.26: "Airplanes". Containment Assured Velocity for Td=1000 msec scenario and two different schemes, where 20 and 2000 RAV samples are considered.

4.5 Subjective Content Complexity and the Performance

Human eye movement is highly dependent on the video content. Inherently, some types of scenes offer more opportunity for compression and some offer less. A perfect compression algorithm should continuously analyze the complexity of a scene and provide the best performance possible. Unfortunately, there is no easy or established means to measure the complexity of the content. With the presence of subjective impact a gross average performance is generally not meaningful. Three test videos were carefully chosen for this work. Each of them represents different visual complexity class. Below a complexity description for each test video is written:

Car: This is a video of a moving car on a parking lot taken from a security camera point of view in one of Kent State University's parking lots. The visible size of the car is approximately one fifth of the screen. Car moves slowly, letting subject to develop smooth pursuit movement. Nothing on the background of this video distracts subject attention. Video duration is 1min 10sec.

Shamu: This video captures an evening performance of Shamu at Sea World, Ohio, during under a tracking spotlight. This video consists of several moving objects: shamu, trainer, and the crowd. Each of them is moving at a different speed during various periods of time. The interesting aspect of this video is that a subject can concentrate on a different objects and it would result in variety of eye-moments: fixations, saccades, pursuit. Such environment suits the goal of challenging RW construction algorithm with different types of eye movements. The fact that the video is taken during night time

provides an interesting aspect of the video perception by the subject. Video duration is 2 mins.

Airplanes: This video depicts formation flying of supersonic planes – performed by Blue Angels on Lake Erie, rapidly changing their flying speeds. The number of planes varies from one to five for duration of the video. Also, the camera action involves rapid zoom and panning. This video provides a challenge for the reflex construction algorithm to build a compact window to contain rapid eye-movements of the saccades and pursuit. Sometimes camera could not focus very well on the planes while capturing this video and subject has to search for the object. This aspect brings additional complication to the general pattern of eye movements for this video. This video duration is 1: min and 9 sec.

4.5.1 RAVs Impact on System Performance

The originals and various perceptually encoded versions of these videos are available from the website [15] for direct visual comparison. Fig 4.27 plots the RW coverage obtained by the proposed algorithm. It is possible to see that, with this algorithm the reflex window was tightest on the “Car” video due to the smooth moving nature of the object inside the video. In the case when the algorithm used last 20 RAVs for RW construction (RAVs=20), the RW on the average was about only 15% of the video frame. The performance of “Shamu” video with more rapid and complicated object movements was next best 17%. “Airplanes” as expected gave worst performance of 30% due to the rapid supersonic airplanes movements inside the video and focusing problems what subject experienced while looking at the videos. Interestingly, the number of RAV

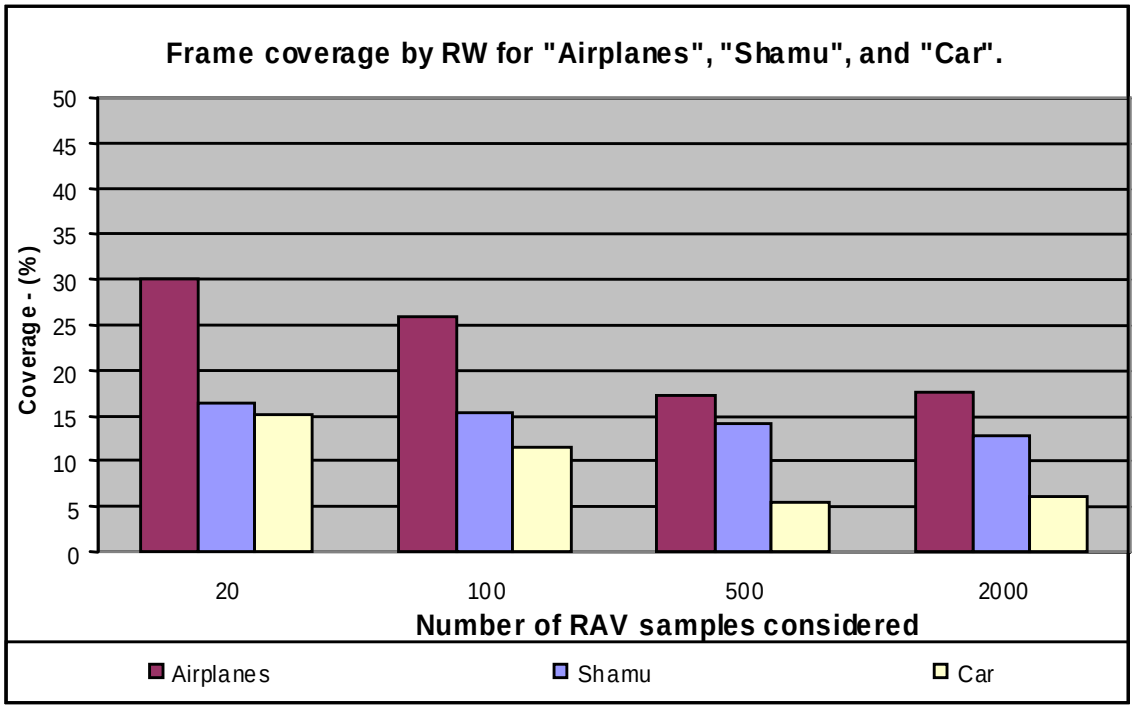

Figure 4.27: Video frame coverage for three videos. Td=166 msec. Axis "x" shows how many RAV samples where taking into consideration for RW construction.

samples considered seems to have effect on the RW coverage. Longer memory with larger number of RAVs seems to have considerable effect in improving the coverage efficiency. In the case when RAVs was equal to 2000 VV the perceptual coverage was reduced to 7%, 13% and 17% for the three videos respectively. Looking at the Fig 4.27 it is possible to see that perceptual coverage was almost the same for RAVs=500 and RAVs=2000. This result can be interpreted as there is a threshold in number of RAVs that should be considered for computing the most efficient RW. After collecting a certain amount of RAV samples system is able to predict the CAV velocity very well. It is also possible to assume that considering more than some number of RAVs (probably more than 2000) would lead to degradation in system performance.

4.5.2 Background Compression Factor

In this section describes the estimation for possible bit rate reduction for a simple bit-rate reduction scheme. This method assumes that the area of RW is encoded with highest possible quality and the rest of the video frame is encoded with a fraction of the RW quality.

$$\psi = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{1\cdot\Delta RW(i)+\rho(\Delta(F(i)-\Delta(F(i)\cap RW(i)))} \tag{4.5.1}$$

ψ is overall possible compression. F(i) represents video frame. RW(i) is predicted reflex window using $E(i\text{-}T_d\text{-}k)$, …. ,$E(i\text{-}T_d)$ eye gazes, where "k" is number of latest RAV samples that the algorithm uses for RW(i) construction. ρ is background

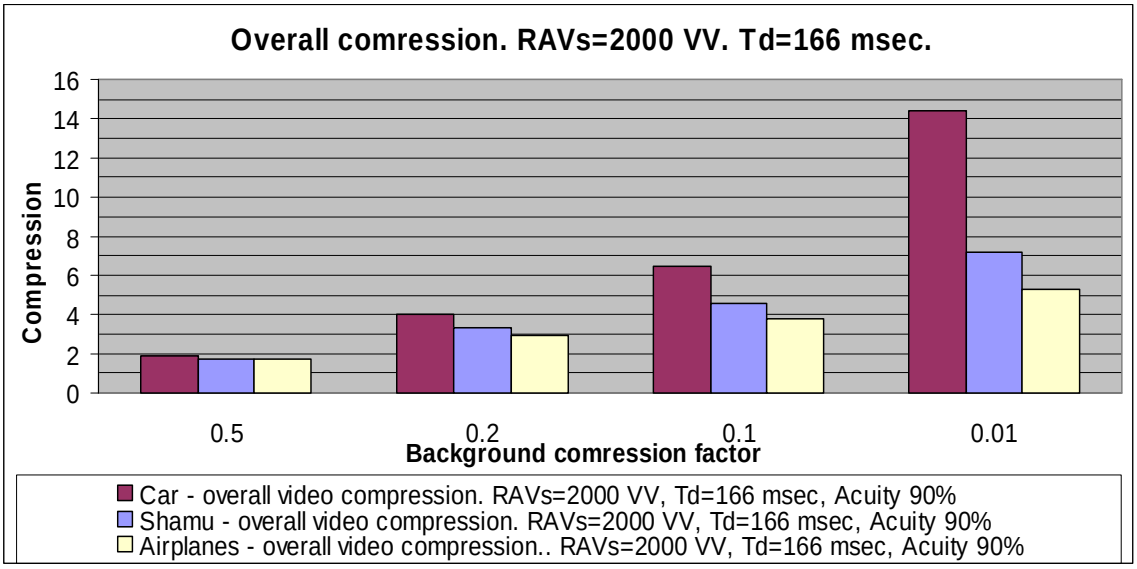

Figure 4.28: Video frame coverage for three test videos. Td=166 msec. Axis "x" shows how many RAVs were taking into consideration for dynamic RW construction algorithm.

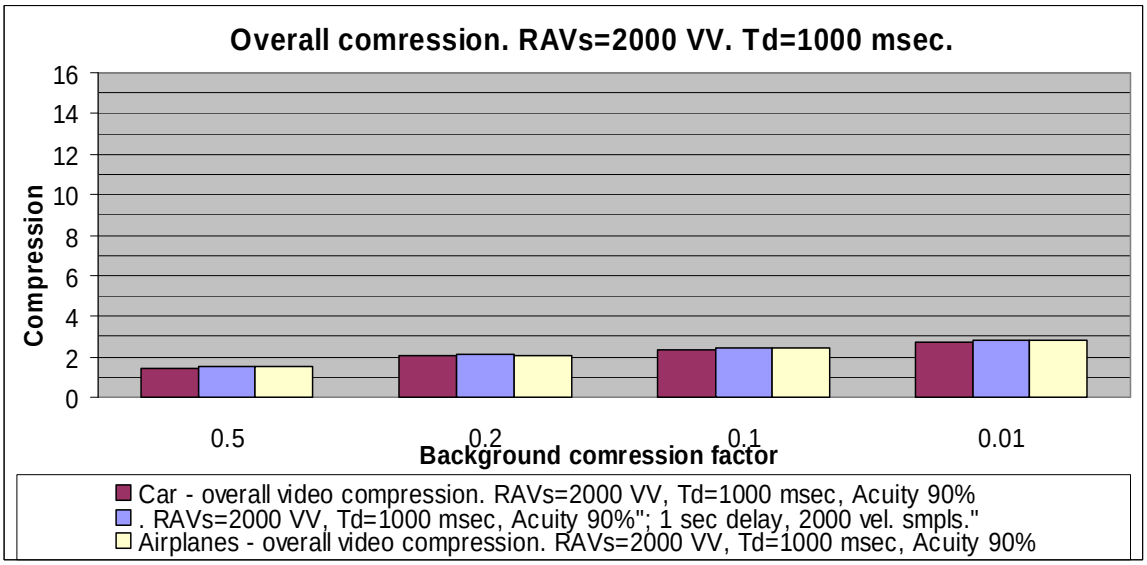

Figure 4.29: Video frame coverage for three test videos. Td=1000 msec. Axis "x" shows how many RAVs were taking into consideration for dynamic RW construction algorithm.

compression factor and it is a positive integer. It is assumed that RW(i) area is encoded with quality equal to “1”.

Fig 4.28 and Fig. 4.29 plot ψ for different ρ. In the case of RAVs=2000, T_d=166 msec and $\rho = 0.01$ it is possible to achieve compression of 14 comparing to original bit rate. It should be noted here that this compression does not take into consideration eye sensitivity function. That means that a subject would probably be able to see compression artifacts.

4.6 Perceptual Compression Efficiency

In this section, the overall compression factor is estimated in case when equation 2.5.2 is used. As it was mentioned before that the use of such function would provide perceptually undetectable compression. For this experiment the viewing distance used in the equation 2.5.2 was chosen to be VD=2*H. H here is the height of the image in inches. For this thesis experiments: H=50 inches, VD=100 inches.

Let C to define intrinsic compressibility, which presents perceptual compression efficiency. N is the number of frames in the video sequence. $S_i(x, y)$ – sensitivity function from equation 2.5.2, which defines visual window for the frame i.

$$C = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{\int_x\int_y S_i(x, y)dxdy} \tag{4.6.1}$$

Fig 4.30 shows the compression level that is possible to achieve using CSF function and accounting for the feedback delay in the system. Right y-axis of Fig 4.30 shows the

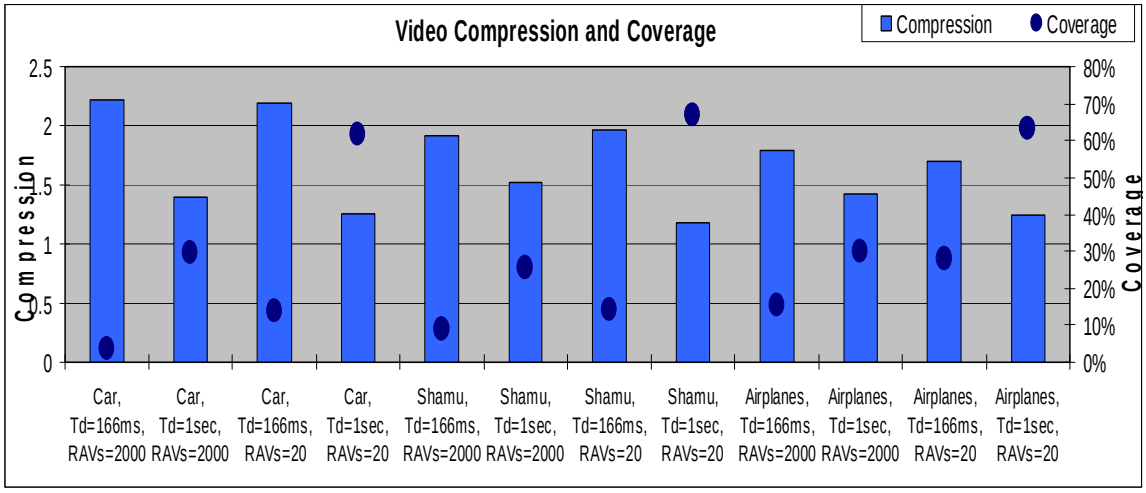

Figure 4.30: Compression estimation and perceptual coverage results for different test videos, Td values and RAVs.

overall compression level achieved by the system. Left y-axis of Fig 4.30 shows average perceptual coverage provided by the Reflex Window. X-axis shows video names, feedback delay values and number of RAV samples considered for RW construction. “Airplanes” video gave worst performance providing 1.7 times compression in the case of feedback delay of 166 msec, RAVs=2000. Compression level for the same video was a even lower in the case of feedback delay of 1000 msec. It was around 1.4. “Shamu” video clip had second best performance result – close to 2 times compression. “Car” video clip had the highest level of compression of 2.3 times in the case of 166 msec delay and RAVs=2000. In the case of feedback delay of 1 sec the compression was reduced to 1.4 times.

CHAPTER 5

Conclusions and Future Work

5.1 Conclusions

This thesis investigated a method of perceptual video compression based on interaction between real time video transcoding mechanism and an eye tracker. The purpose of the thesis was to study the impact of feedback delay created by the fact that a video transcoder is deployed in the network, presumably Internet. Depending on how far is the distance between eye tracker equipment and the transcoder the data transfer time between these two points might vary. In its current implementation the created system can accept and serve live or stored MPEG-2 ISO-13818-2 content over a network taking direct perceptual interaction with human eye. The feedback delay is value that should be provided to the system prior to the encoding process.

The current mainstream in eye-tracking based perceptual communication research has heavy concentration on the design of CSF. The experiment that is described in this Thesis with a live system strongly suggests that CSF plays less dominant role in overall video streaming encoding than it was previously assumed. The feedback delay in control loop (in network, in media encoding, or even the delay within the eye-tracker) creates a perceptual window several times larger than the para-fovea area previously proposed by CSF researchers. Considering this a simpler approximation of CSF will probably be as good as a detailed acuity model as far as the overall system performance is concerned.

Due to the need to predict future eye moments within some bounded area, there is no need to encode the area outside such boundary using extremely accurate acuity function. Much more gain can be achieved by reducing the reflex window size. It could be done by taking into consideration the duration of eye moments and prediction of what eye movement type is going to happen next. Video quality degradation around the reflex window has to be designed with the avoidance of encoding artifacts such as blocking effects to ensure best system performance. The appearance of such artifacts can reduce system performance more than a bad choice of a CSF function.

This Thesis presented the algorithm for the reflex window estimation. It showed that more than 90% of the gazes can be contained within 20-25% of the video frame. Compression of up to 2.3 times was achieved with consideration of eye sensitivity function and the feedback delay.

In concept this reflex window can be applied to perceptually encode any visual media type. The actual perceptual quality of the presentation will depend on the specific encoding technique used to map the visual window (which is mainly built from reflex window) on the specific media type. Once, the visual window is obtained there are numerous ways in which the region discriminating encoding can be performed with various computational-effort/quality/rate trade-off efficiency. Though this thesis provides the actual description of visual window bit-rate mapping in MPEG-2 standard, the main contribution of this work is how to succeed in keeping the eye gazes contained within the reflex window. The technique proposed in this system can be applied to the visual media of any type and it is coding standard independent.

Perceptual engineering based data compression scheme is applicable not only for video but for almost all visual presentations. This is particularly attractive for large field of view projection systems. The mainstream data compression technology has matured over the last two decades. These rely mostly on extraction of statistical redundancy (particularly spatial, temporal, and frequency domain redundancy). It is interesting to note that reported improvement in compression factor from newer methods has diminished in recent years. Is it possible that the statistical methods have reached a form of theoretical limit based on the pixel entropy of the content? Perceptual engineering may offer the next big wave of improvement. The field is still in its infancy. The potentials are enormous. More techniques, which can exploit intricate characteristics of our vision system such as directional eye velocity, interaction with the content, peripheral vision, can provide novel clues to push the compression limit further.

5.2 Limitations and Future Work

One of the major contributions of this thesis is that it studies a critical problem towards bringing eye-gaze based perceptual transcoding one step closer to reality. However, there are several other hurdles to be solved before such a system can be fully functional.

The issue of target gaze containment needs to be examined more. In this work the target gaze containment was set to 0.9, meaning that on average 90% of the gazes were contained within RW. There is significant potential that the coverage area can be further reduced by using additional scene analysis.

Another potential improvement is that the feedback delay T_d can be also dynamically adapted to the network changes. The topology of the Internet is constantly changing. It might lead to different data transfer times, during different period of times, which can influence the value of T_d. Making the transcoder capable of estimating the feedback delay dynamically and adjusting T_d value accordingly might significantly improve system performance.

This thesis proposes a way of predicting the velocity of future eye moments, by calculating CAV. This algorithm might be further enhanced, by careful consideration of eye movement types. Saccades for example have fixed duration. After saccade is over a fixation should take place. If system is capable of detecting a saccade, then knowing its duration and the fact that it is doable to reduce visual quality without subject noticing it during saccade movement, it is possible to reduce video bit rate without perceptually detecting it.

In this Thesis the shape of proposed RW is ellipse. The elliptic shape is built under assumption that eye can move in any direction from the center of RW with equal probability. Elliptic shape can be modified to something else, if RW construction algorithm takes into consideration the course of the eye gaze points over time. If the RW shape is modified in this way then there might be a possibility of RW size reduction, while maintaining same level of gaze containment.

References

[1] V. Virsu, J. Rovamo, “Visual resolution, contrast sensitivity, and the cortical magnification factor” in *Experimental Brain Research V. 37*, 1979.

[2] A. Johnston, “Spatial scaling of central and peripheral contrast sensitivity functions”, *JOSA A V.4 #8*, 1987.

[3] Applied Science Laboratories, “Eye tracker manual (model 504)”, Applied Science Group Inc, ASL324-M-998.

[4] Daly, Scott J., “Engineering observations from spatiovelocity and spatiotemporal visual models” in *Human Vision and Electronic Imaging III*, July *1998, SPIE*.

[5] Daly, Scott J.; Matthews, Kristine E.; Ribas-Corbera, Jordi, “Visual eccentricity models in face-based video compression” in *Human Vision and Electronic Imaging IV*, May *1995, SPIE*.

[6] Duchowski, A.T., “Acuity-Matching Resolution Degradation Through Wavelet Coefficient Scaling. IEEE Transactions on Image Processing 9, 8. August 2000

[7] Duchowski, A.T., McCormick, Bruce H., “Simple multiresolution approach for representing multiple regions of interest (ROIs)” in *Visual Communications and Image Processing '95, April* 1995, SPIE.

[8] Duchowski, A.T., McCormick, Bruce H., “Preattentive considerations for gaze-contingent image processing” in *Human Vision, Visual Processing, and Digital Display VI, April* 1995, SPIE.

[9] Duchowski, A.T., McCormick, Bruce H., “Gaze-contingent video resolution degradation” in Human *Vision and Electronic Imaging III*, July 1998, SPIE.

[10] Geisler, Wilson S.; Perry, Jeffrey S.; “Real-time foveated multiresolution system for low-bandwidth video communication**”** in *Human Vision and Electronic Imaging III, July 1998, SPIE.*

[11] Kim, Man-Bae; Cho, Yong-Duk; Kim, Dong-Kook; Ha, Nam-Kyu; “Compression of medical images with regions of interest (ROIs)” in *Visual Communications and Image Processing '95, April* 1995, SPIE.

[12] Kortum, Philip; Geisler, Wilson S., “Implementation of a foveated image coding system for image bandwidth reduction” in *Human Vision and Electronic Imaging*, April *1996, SPIE.*

[13] Kuyel, Turker; Geisler, Wilson S.; Ghosh, Joydeep, “Retinally reconstructed images (RRIs): digital images having a resolution match with the human eye” in *Human Vision and Electronic Imaging III*, July *1998, SPIE.*

[14] Lester C. Loschky; George W. McConkie, “User performance with gaze contingent multiresolutional displays” in *Eye tracking research & applications symposium,* November, 2000.

[15] TR2002-06-01 Perceptually Encoded Video Set from Dynamic Reflex Windowing, http://medianet.kent.edu/techreports.html, also mirrored at http://bristi.facnet.mcs.kent.edu/ ~javed/medianet/techreports.html.

[16] E.L. Niu, “Gaze-based video compression using wavelets”. University of Illinois at Urbana-Champaign. The Graduate College. August 1995.

[17] Tsumura, Norimichi; Endo, Chizuko; Haneishi, Hideaki; Miyake, Yoichi; “Image compression and decompression based on gazing area” in *Human Vision and Electronic Imagin*, April *1996, SPIE*.

[18] L. Yarbus “Eye Movements and Vision” Institute for Problems of Information Transmission Academy of Sciences of the USSR, Moscow 1967.

[19] Irwin, D. E. Visual Memory Within and Across Fixations. In Eye movements and Visual Cognition: Scene Preparation And Reading, K. Rayner, Ed. Springer-Verlag, New-York, NY,1992, pp. 146-165. Springer Series in Neuropsychology.

[20] Duchowski, A.T., McCormick, Bruce H., “Gaze-contingent video resolution degradation” in Human Vision and Electronic Imaging III, July 1998, SPIE.

[21] Information Technology- Generic Coding of Moving Pictures and Associated Audio Information: Video,ISO/IEC International Standard 13818-2, June 1996.

[22] S. Lee, M. Pattichis, A. Bovok, Foveated Video Compression with Optimal Rate Control, IEEE Transaction of Image Processing, V. 10, n.7, July 2001, pp-977-992.

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

[24] Duchowski, A.T., "3D wavelet analysis of eye movements" in *Wavelet Applications V*, March 1998, SPIE.

[25] Stelmach, Lew B.; Tam, Wa James; Hearty, Paul J.; "Static and dynamic spatial resolution in image coding: an investigation of eye movements" in *Human Vision, Visual Processing, and Digital Display II, June 1991, SPIE.*

[26] Stelmach, Lew B.; Tam, Wa James; "Processing image sequences based on eye movements" in *Human Vision, Visual Processing, and Digital Display V, May 1994, SPIE.*

[27] Wampers, M., Diepen, P.M.J.V. and d'Ydewalle, G. The use of coarse and fine peripheral level of detail degradation when used with head mounted displays, Georgia Institute of Technology, Graphics, Visualization & Usability Center., 1995.

[28] van Diepen, P.M.J. and Wampers, M. Scene exploration wth Fourier-filtered peripheral information. *Perception,* 27 (10). 1998, 1141-1151.