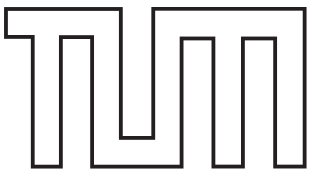


Technische Universität München
Chair of Media Technology
Prof. Dr.-Ing. Eckehard Steinbach

Master Thesis

Face Tracking for Optimized Bitrate Control in Low Delay Video Encoding

Author:	Chethan Ningaraju
Matriculation Number:	03659451
Address:	Schröfelhofstraße 10-05-07
	81375 Munich
Advisor:	Dr.-Ing. Eugen Wige, Muhammad Zafar Iqbal
Begin:	15/07/2016
End:	15/01/2017

With my signature below, I assert that the work in this thesis has been composed by myself independently and no source materials or aids other than those mentioned in the thesis have been used.

München, January 23, 2017 _____

Place, Date Signature

This work is licensed under the Creative Commons Attribution 3.0 Germany License. To view a copy of the license, visit http://creativecommons.org/licenses/by/3.0/de

Or

Send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California 94105, USA.

München, January 23, 2017 _____

Place, Date Signature

Abstract

In video conferencing systems, coding artifacts are especially disturbing in the face region at low bitrates. In this work, region-of-interest (ROI) based encoding techniques are proposed to preferentially code the face region with a higher quality to improve the perceived visual quality of the video conferencing system. The face region is detected in the video stream and marked as ROI before encoding.

The low-delay bitrate control proposed for H.264 video coding standard is extended to implement two different approaches of ROI-based encoding. In the first approach, QP offsets for ROI is computed based on the relative area of ROI to reduce the quantization parameter (QP) allocated for ROI macroblocks. In the second approach, region-based bit-allocation is performed to allocate a higher proportion of bits to ROI macroblocks. The two approaches offer a trade-off between complexity and output quality. In contrast to the previous ROI-based video coding approaches, this work uses the complexities of ROI and non-ROI parts in a video frame to allocate an optimal amount of bits for the corresponding regions. Experimental results demonstrate that the quality of ROI is improved without any noticeable degradation in non-ROI quality hence improving the overall perceived visual quality at low bitrates. The results also verify that there are no undesirable changes in the overall behavior of the bitrate control with ROI-based encoding.

In addition to ROI-based coding, an optimization technique is proposed for real-time face detection in a video stream by reusing motion vector information computed during motion estimation stage of video encoding. This work proposes motion vector based variable interval face detection technique instead of detecting the face in every frame of the video. The motion vectors of the face region are analyzed to perform face detection only when there is movement in the face region. Experimental results show considerable speedup for marginal inaccuracies in detecting the face.

Acknowledgement

I would like to express my sincere gratitude to everyone who supported me during the journey of my thesis.

I would like to thank my supervisors Dr.-Ing. Eugen Wige, Citrix, Germany and Muhammad Zafar Iqbal, Chair of Media Technology, TUM for guiding me continuously, steering me in the right direction with valuable advices, reviewing and guiding me in the process of writing this thesis. I would like to thank Prof. Dr.-Ing. Eckehard Steinbach for being very supportive throughout my Master's study, giving me the opportunity to carry out my Master Thesis under the Chair of Media Technology, TUM and taking the time to evaluate my work. I would also like to thank Alexander Gehlert, Citrix and the entire Video Processing team at Citrix, Germany for providing me the opportunity to work on this thesis at Citrix, Germany.

Finally, I would like to express my gratitude to my family and friends for providing continuous support and encouraging me throughout the years of my Master's study. This accomplishment would not have been possible without them.

Contents

Chapter 1

Introduction

In recent years, there is an increasing demand for high-quality video conferencing solutions. Due to the availability of high-speed Internet, video conferencing has proved to be an efficient alternative to face-to-face meetings. Video telephony has grown into a multi-billion dollar industry and has a huge commercial significance. To address this growing need there has been a constant improvement in low-delay video coding techniques in addition to better techniques to ensure low-delay transmission reliability at the network level. The tremendous increase in smartphone usage has led to an increase in video telephony over cellular networks whose bandwidth is highly constrained. Therefore, it is very important to develop methods of delivering high-quality video with less bandwidth requirement.

The most commonly used video coding standards like H.264/Advanced Video Coding (AVC) have been designed to exploit the spatial and temporal redundancies in the input video stream to achieve high data compression. The techniques of spatial and temporal prediction form the core principle of these video coding standards [TWL03]. However, after encoding the video the perceptual redundancies still remain since human attention does not focus on the whole scene but only a small region of fixation called region-of-interest (ROI) [MX14]. Therefore, reducing the perceptual redundancy gives a new dimension towards achieving lower bit-rate with acceptable perceptual quality. This work proposes ROI-based bitrate control schemes for low-delay video encoding to exploit the perceptual redundancies.

In this work, the salient region of the frame which is the face of the participant in a video conference is identified. Since the attention of the viewer is mostly focused on the face of the other participants during a video conference call, improving the quality of the face region (ROI) can improve the overall perceptual quality. In this work, ideal capture conditions are assumed and the result of face tracking is used directly as supplementary information for the H.264/AVC encoder's bitrate control. This work explores the methods of ROI-based encoding to exploit the available bandwidth to encode regions that are of high importance to perception with higher quality. Face region in the input stream is

allocated an above-average bit-count to yield a better visual quality than the background regions. It is the aim of this work to develop and extensively evaluate the strategy of uneven bit-allocation and also to identify its limitations.

In this work, OpenCV implementation of face detection based on Viola-Jones AdaBoost algorithm [VJ01] is used to detect the face and mark it as ROI. An optimization technique using motion vector-based variable interval face detection is proposed for real-time face detection in a video stream. This work proposes two different approaches of ROI-based encoding to improve the quality of the face region. The first approach is to use a negative QP offset for the macroblocks belonging to ROI. The QP offsets reduce the QP of the ROI macroblocks resulting in a higher ROI quality. This approach is used to assess the effect of increasing the magnitude of the quality difference between ROI and non-ROI on the perceived visual quality. The second approach proposes a ROI-based bit-allocation scheme which allocates a higher proportion of bits to ROI considering the importance of the face region to the perceptual quality. This approach differs from other ROI-based bit-allocation schemes by considering the content properties to compute the optimal amount of bits for ROI and non-ROI. The spatial and temporal complexities of ROI and non-ROI are used to split the available bandwidth between ROI and non-ROI parts. The behavior of ROI-based bitrate control is compared with that of conventional bitrate control to make sure that ROI-based encoding does not alter the behavior of bitrate control in any undesirable manner.

The remainder of this thesis is organized as follows. Chapter 2 gives an overview of the hybrid video coding used in H.264, functionality of the bitrate control module and the concept of ROI-based encoding. In Chapter 3, a literature review is presented which discusses related works in the field of ROI-based encoding. An insight into the limitations of earlier works is also presented in this chapter. A detailed overview of the low-delay bitrate control module [SML02] used in this work is presented in Chapter 4. Chapter 5 deals with explanation of the setup used in this work along with the assessment techniques to evaluate the ROI-based encoding approaches. The procedure for marking ROI in a frame using face detection and the proposed optimization technique for real-time face detection in a video stream is discussed in Chapter 6. The proposed approaches for ROI-based encoding along with experimental results is presented in Chapter 7. The conclusion for this thesis work along with a note on future directions is given in Chapter 8.

Chapter 2

Background

A brief overview of principles of hybrid video coding used in H.264 is presented in this chapter. The ROI-based encoding approaches discussed in this work are implemented in the form of intelligent bitrate control schemes. Therefore, an overview of the bitrate control module is presented to provide the reader with an understanding of the functionality of bitrate control in a video encoder. Finally, the concept of ROI-based encoding is introduced.

2.1 Hybrid Video Coding

H.264/AVC is one of the most commonly used video coding standard. Figure 2.1 depicts the underlying principle of block-based hybrid video coding used in H.264 [TWL03]. The encoding scheme aims to exploit the spatial and temporal redundancies that exist in a video.

In this coding scheme, the input picture is represented in block-shaped units (16x16 pixels) of associated luma and chroma samples called macroblocks (MBs). The basic source-coding algorithm is a hybrid of inter-picture prediction to exploit temporal statistical redundancies and transform coding of the prediction residual to exploit spatial statistical dependencies. The two types of prediction used are:

Intra Prediction: There exists a high similarity among the neighboring blocks in a video frame. In intra prediction, a block is predicted from its neighboring pixels of already coded and reconstructed blocks. H.264 offers nine intra prediction modes (one DC prediction mode and eight directional prediction modes) [TWL03].

Inter Prediction: When the framerate is sufficiently high, there is a great amount of similarity between the neighboring frames. It is highly efficient to code the difference between such similar frames than the frames themselves. In inter prediction, block-

based motion estimation (ME) is used to predict the motion of macroblocks relative to the previously encoded frames.

The first frame in a video is encoded using intra prediction and transform coding. The transform coefficients are quantized to achieve high compression ratio. The frames which are encoded without any dependencies on the neighboring frames are called key/intra frames, and act as reference frames to encode subsequent frames.

Once a reference frame is available, inter prediction can be used to remove the temporal redundancies. During motion estimation, it is usually not possible to find an exact match for the current macroblock. Therefore, the residual error is estimated for the prediction from motion estimation. This process is called motion compensation. The residual error is coded using transform coding followed by quantization to achieve higher compression ratio. Due to quantization, it is not possible to recover the exact transform coefficients at decoder end without any loss of information. This introduces distortion in the decoded frame.

These frames which are predicted from other reference frames are called inter frames. The inter frames which use only past frames as reference are called P-frames. In addition to past frames, future frames can also be used as a reference. Such frames are called bi-directional frames or B-frames. Both B-frames and P-frames can be used as reference frames for encoding subsequent frames.

The quantization step used to quantize transform coefficients is specified using the quantization parameter (QP). In H.264, QP range of 1 to 51 is allowed which is translated to quantization steps. The magnitude of distortion introduced by quantization depends on the quantization parameter. A lower QP implies low quantization step resulting in a high-quality output with less compression. The compression ratio increases with increase in QP at the cost of decreased output quality. The output data-rate of the encoder is controlled by computing suitable QP, this is the task of the bitrate control module as described in the following section. A detailed overview of individual steps involved in H.264 coding can be found in [TWL03].

2.2 Bitrate Control

The bitrate control module is responsible for controlling the bit-consumption of the encoder to guarantee smooth playback. Bitrate control is not specific to a video coding standard and hence operates independent of any chosen video coding standard. There are various flavors of bitrate control like Constant Bitrate (CBR) and Variable Bitrate (VBR). In this work, CBR type of bitrate control is considered since it is most commonly used in video conferencing and other real-time streaming applications.

Figure 2.2 illustrates the functionality of the bitrate control module. The main purpose

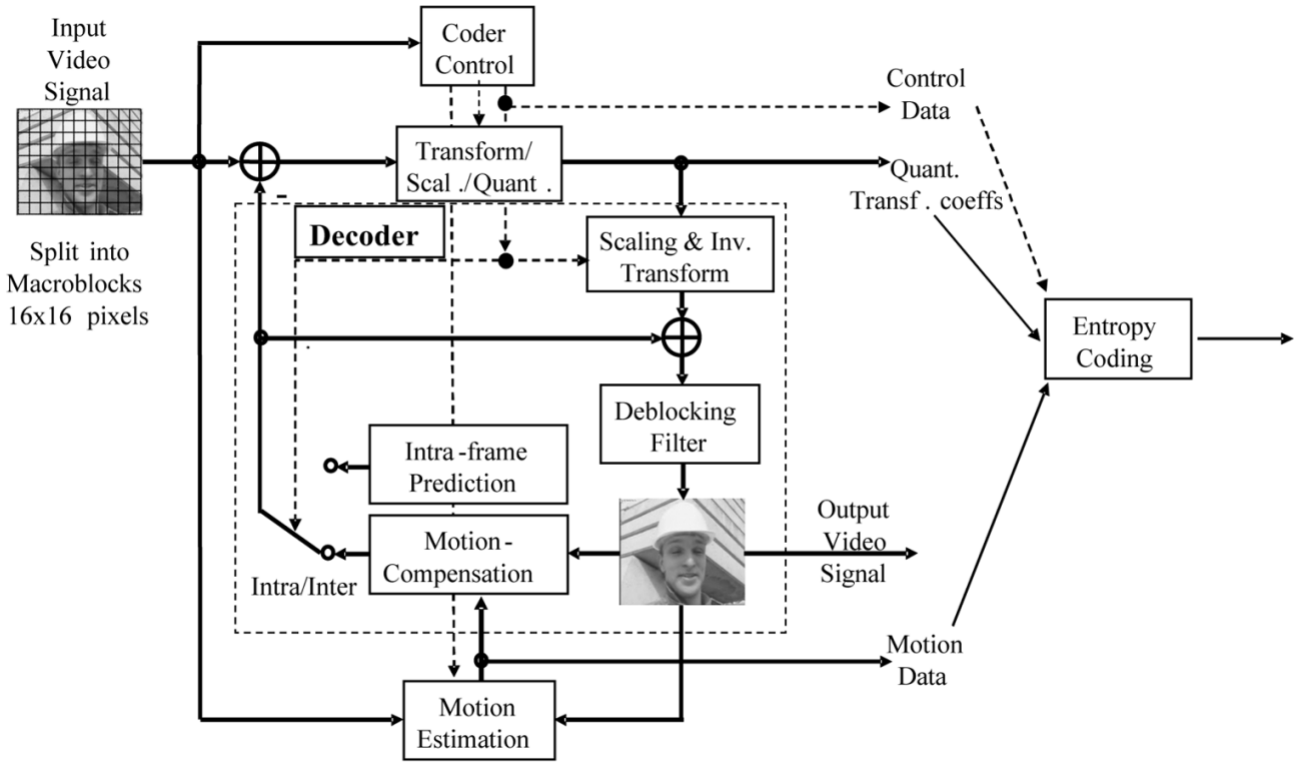


Figure 2.1: Basic coding structure for block-based hybrid video coding [TWL03].

of the bitrate control module is to ensure smooth playback of the encoded video under given bandwidth and delay constraints. It estimates the video bitrate based on the available network bandwidth, ensures that the coded bitstream can be transmitted within the specified delay and makes full use of the limited bandwidth [ZWW11]. It achieves this by controlling the QP used during encoding. The QP is computed considering the input bitrate, framerate, input complexity (spatial and temporal activity) and acceptable delay of the system. The module also receives regular feedback from the encoder to make a better QP adaptation. The feedback from the encoder gives information about the complexity of the input video and helps the bitrate control to compute the optimum QP for a given bit-budget.

The functionality of the bitrate control module can be illustrated with the help of the leaky-bucket model [ZWW11]. The output data-rate of a video encoder varies depending on the input complexity of the video (motion in the frame). It also depends on the picture type of the encoded frame. The key/I-frames consume a lot of bits compared to the inter pictures (P-frames and B-frames). In a video streaming scenario considering a constant bitrate channel, the throughput is maximum when the data-rate is constant and equal to the available bandwidth. Therefore, the output data-rate of the encoder is smoothened using a theoretical buffer called Video Buffer Verifier (VBV). The VBV is a virtual buffer modeled by the bitrate control module to ensure that the video stream can be correctly

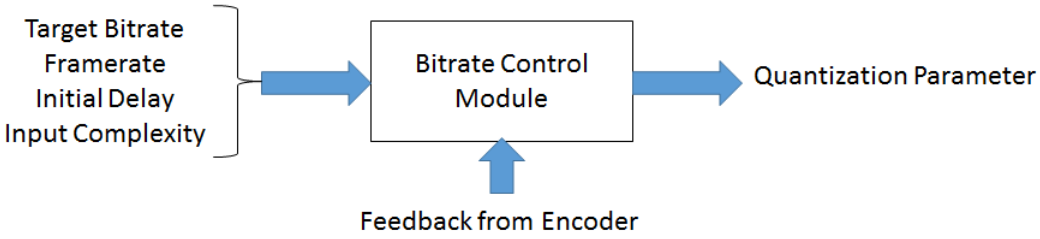


Figure 2.2: Bitrate Control Module Functionality

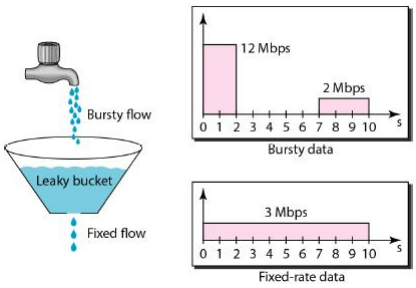


Figure 2.3: Leaky Bucket Model [Lea]

buffered and played back at the decoder end. This is equivalent to the leaky-bucket model as shown in Figure 2.3 [Lea], where the output of the encoder with a variable rate (bursty output) is stored in a buffer (leaky bucket) which is draining at a constant rate. Any underflow or overflow of this buffer causes a glitch in the video streaming. To ensure that there is no VBV overflow or underflow, the encoder's quantization parameter is adapted on a macroblock level so that the maximum allowed bit-count for an encoded frame is not exceeded. A detailed description of the low-delay bitrate control module used in this work is given in Chapter 4.

2.3 ROI-based Coding

In conventional video coding, all regions of a frame are considered equally important to the viewer. It is assumed that all regions contribute equally to the perceptual quality. However, the study of Human Visual System (HVS) shows that human eyes can only focus on one area in a frame at any given time which is called region of interest. For example, it has been found out in [Wan95] that humans normally perceive clearly a small region of 2°-5°of the visual angle. This corresponds to a very small region in the video frame.

In video coding, the compression gain from spatial and temporal prediction is reaching the saturation level. Further compression from these techniques demand exponential growth in computational capabilities. The next generation video coding standards like High Efficiency Video Coding (HEVC)/H.265 claims to offer two fold increase in compression ratio

over H.264. This increase in compression comes with multi-fold increase in computation complexity during encoding [DGH13]. Therefore, perceptual coding can provide an alternative solution towards low-bitrate video coding. The technique of encoding important regions (ROI) at a higher quality at the cost of degradation of quality in non-ROI parts is called ROI-based coding. The loss of quality in non-ROI is not perceived by the viewer leading to increased perceptual quality at the same bitrate.

The ROI-based coding is not a common practice in video coding because it is very hard to automatically detect the important regions in generic contents that contribute the most to the perceptual quality. There are many ways of detecting ROI, most of which are application specific. The most common approach is the usage of difference image based moving object detector. In these systems, any moving object is considered as ROI. A typical use-case for such a system is video surveillance. In addition to difference image based motion detection, global motion estimation is used in ROI detection [HM11] in applications like aerial surveillance.

In generic video content, ROI changes constantly depending on the context. For instance, in a movie, the ROI can depend on the context of the scene. Developing generic techniques for detection of ROI in such videos is very difficult. There have been attempts to use eye tracker to record the foveation points of a human observer on the receiver which was used to apply foveation filter in video coding of the sender. An advancement over such an approach is proposed in [SL03] which optimizes rate control to maximize foveal visual quality metric. These generic ROI detectors are very hard to implement due to the uncommon availability of eye tracking mechanism at the receiver.

The ROI in the video conferencing scenario is going to be the face region predominantly. Due to recent improvements in face detection algorithms, it is possible to detect the face with good accuracy. The study in [MB13] shows that boosting the quality of the face regions can improve the overall perceived quality of the video. This work aims to study the possible ways of improving the perceptual quality of the video by detecting the face region and coding it with higher quality than the rest of the frame.

In conventional video coding, the bitrate control allocates bits at every macroblock and adjusts the QP accordingly. In a simple approach, every macroblock is considered equally important to perceptual quality hence the available bits are distributed evenly across all the macroblocks within a frame. Since low-latency and load efficiency are of high importance, it is not advisable to do multi-pass encoding for optimal bitrate allocation. Therefore, over-allocation in one macroblock has to be compensated by under-allocation (using a higher QP) for the neighboring macroblocks, regardless of the image content. However, a more intelligent allocation strategy should take the image content into account. In this work, parts of the image with higher importance (ROI) is given a higher share of the overall bit-count resulting in a higher visual quality. The additional bits spent in coding ROI is compensated by allocating a lower proportion of the bit-count to the background regions (non-ROI).

The knowledge of region of interest in input video can be used for many other purposes in addition to its use in video coding to improve the perceptual quality. For instance, ROI information can be used in developing techniques for smart thumbnail displays in group video conferencing solutions. In a group video conference, an active person is detected based on the source of the voice and is displayed on the main screen and other participants are displayed on the smaller windows with down-scaling of the entire video. If the face coordinates of the participant is transmitted along with the bit-stream as meta-data, a cropped version of the video can be displayed to show only the face in the thumbnail display. This can improve the overall user-experience.

Chapter 3

Related Work

The concept of ROI-based encoding has been around for a while. In this chapter, some of the state-of-the-art techniques are discussed. The different approaches of using ROI information to improve the perceptual quality can be broadly classified into the following categories based on the stage of processing using ROI information.

- *Preprocessing* - Example: blurring.
- *Video Coding* - Example: during the rate-distortion optimization (RDO), bitrate control.

The relevant techniques proposed under these categories are discussed in the following sections.

3.1 Preprocessing

The input video stream can be directly altered based on the ROI information. Many preprocessing approaches are used to directly reduce unimportant information by applying a non-uniform distortion filter in the scene. For instance, the image is divided into foreground (ROI) and background (non-ROI) and the non-ROI parts are blurred to save the bits during encoding [AC05]. The work in [HM16] is an example for codec independent ROI encoding. It can work with any codec and arbitrary ROI detector. In this approach, input video stream is modified only to contain relevant information. The non-ROI pixels are replaced such that these regions can be very efficiently compressed by the encoder. The non-ROI macroblocks are either replaced by corresponding blocks from the previous frame or black blocks. The non-ROI regions are reconstructed with post-processing assuming a zero motion vector. This approach is used in scenarios where non-ROI regions are mostly discarded completely at the receiver end.

3.2 Video Coding

In video encoding, multiple approaches exist to preferentially code the ROI macroblocks with a higher quality at the cost of degrading non-ROI parts. The ROI information can be directly used in RDO during video encoding to alter the quality of ROI. One such approach is presented in [FL16] in which non-ROI macroblocks are encoded using only AMC-based modes (Active MB Concealment). The non-ROI macroblocks are encoded with only motion vector but no residual information. This creates a bias to choose larger distortion for non-ROI blocks to save bits during Lagrangian optimization. The bits saved during encoding of non-ROI macroblocks is used to encode ROI blocks with a higher quality

The resources available during encoding like computational power is preferentially allocated to ROI in [YLS08b]. The H.264 coding standard offers many methods to enhance its compression performance such as variable block size ME, quarter-sample-accurate ME, and multiple reference frames ME [TWL03]. The complexity of H.264 encoder is significantly increased due to the employment of these new methods. The computation need for non-ROI is reduced significantly at the cost of compression efficiency by adaptively adjusting the coding parameters as follows:

- Choose only a subset of macroblock partition and prediction modes offered by H.264 standard for non-ROI during rate-distortion optimization to find the best mode. This is similar to the usage of only AMC-based prediction modes for non-ROI proposed in [FL16].
- The number of reference frames is reduced to use only immediate neighboring frames for non-ROI macroblocks. The ROI MBs are allowed to reference from multiple frames.
- The accuracy of subpixel-accurate ME is set to quarter-pixel for ROI and half-pixel for non-ROI MBs.
- The ME search range is reduced significantly for non-ROI.

These modifications in coding parameters result in a higher quality for ROI even when the available bit-budget is uniformly distributed across ROI and non-ROI parts.

A more common way of ROI-based encoding is by altering the rate control module to allocate above average bits to the ROI macroblocks. In addition to the preferential allocation of computation power to ROI, ROI-based rate control is proposed in [YLS08b]. The modifications to the rate control scheme for low-delay video communication of H.264 [YLS08a] has been proposed to allocate more bits (smaller QP) to the ROI macroblocks. This is done by assigning weights to every macroblock based on its importance to the human visual system. A linear R-Q model is proposed to optimize the QP calculation to provide ROI-based rate control at the MB level.

The work in [GLW12] proposes a bit-allocation and rate control scheme for enhancing

the regional perceptual quality using the structural similarity index (SSIM) as the quality metric for distortion-quantization modeling. Statistical analysis is adopted to obtain the relation between SSIM of reconstructed MBs and corresponding QP (from 20 to 51 in this paper) after standard video coding. A target SSIM is set for the ROI and corresponding QP is determined with the help of the SSIM-QP model. The proposed algorithm has the following steps.

- Target bits for each basic unit is estimated and allocated based on the bit-budget of ROI and non-ROI parts.
- A preliminary QP for each BU is computed based on R-Q model.
- The predicted QP is used for rate-distortion optimization mode decision of the encoding flow to obtain the best prediction macroblock.
- The QP used for MBs in ROI for final encoding is altered to achieve the target SSIM quality based on the SSIM-QP model.

An ROI-based encoding scheme specific to video conferencing is proposed in [LT05] for H.263. This work proposes an algorithm to track the face using motion-vector information. Once the ROI is detected, a modified bit-allocation scheme is used for ROI-based encoding. The QP is predicted from the rate modeling of ROI and non-ROI blocks. The work described in [MX14] goes a step further in face detection based ROI encoding schemes by enhancing finer facial feature to improve the perceptual quality in high-resolution HEVC encoding. In this approach, different weights are assigned to the background, face, eyes, mouth and nose regions which are in-turn used to alter the quality by ROI-based adaptive CTU (Coding Tree Unit) partition structure for HEVC.

3.3 Inference

The preprocessing based ROI encoding approaches described in Section 3.1 [AC05, HM16] offers codec-independent ways to enhance the quality of ROI. However, these approaches are not suitable for video conferencing since they can cause a very high degree of degradation in the background regions (non-ROI) due to preprocessing. An excessively degraded background can also reduce the perceptual quality. It also needs an additional stage of preprocessing which adds to the complexity of the system making it hard to process in real-time.

The proposed bitrate control schemes for ROI-encoding [LT05, YLS08a] assign arbitrarily large weights to the ROI macroblocks compared to the non-ROI macroblocks during bit-allocation. These weights do not take into account the characteristics of the content of the input video stream. These works propose effective methods to create a quality difference between ROI and non-ROI. However, they do not throw sufficient light on determining the optimal quality difference to achieve the best perceptual quality. The approach of

targeting an SSIM value for the ROI macroblock [GLW12] offers a simple way to guarantee the minimum quality for the ROI macroblocks. However, it is difficult to come up with a target SSIM value for different contents and bitrates that yields the best perceptual quality.

Most of the previous work concerned with the modification of rate control to achieve better quality in ROI macroblocks deals with altering the bit-allocation module. Some of the commonly used bitrate control schemes [SML02] do not perform bit-allocation at the macroblock level. The modifications to the rate control proposed in these works cannot be adopted in such a bitrate control module.

In this thesis work, some of the shortcomings of the previous works are addressed. This work proposes ways of achieving ROIbased encoding for bitrate control modules that do not perform an explicit bit-allocation at the macroblock level. This work proposes multiple approaches for ROI-based bitrate control which vary in terms of the ease of implementation. The characteristics of the content in both ROI and non-ROI parts are taken into account to vary the weights used during bit-allocation. This results in a superior ROI-based bitrate control scheme which yields better perceptual quality across a wide range of input video streams.

Chapter 4

Low-delay Bitrate Control

This section gives an overview of the low-delay bitrate control module used in this work. The need for extremely low end-to-end delay in video telephony puts the following additional constraints on video coding which results in compromise of video quality.

No B-Frames: During the low-delay video encoding, tools like the bi-directional prediction (B-frames) are disabled. The usage of B-frames needs buffering of at least one frame. This adds on to the overall latency of the system which is highly undesirable in video conferencing.

Reduced buffer size: The tolerable delay in video encoding is a direct measure of VBV size. When the size of VBV is very low (due to low delay), there is less room to accommodate the variation in the bitrate of the encoder. This implies that there can be minimum variation in the frame size across different frames irrespective of the content.

Increased dropped frames: Any wrong prediction of QP by the bitrate control module can have a bad impact since there is no additional time available to re-encode the content with a corrected QP. For instance, the over-consumption of bits by a frame which can lead to dropped frames cannot be corrected by re-encoding the frame with a higher QP.

The bitrate control module used in this work is a modified version of [SML02]. The bitrate control performs frame level bit-allocation based on fullness of the VBV, followed by adapting QP at the macroblock level based on the structural complexity of the macroblock. The functionality of the bitrate control module can be divided into two stages:

- Bit Allocation
- QP Prediction

  These two stages are described in detail in the following sections.

4.1 Bit Allocation

The low-delay encoding mode does not favor the usage of key-frames at regular intervals and B-frames. Therefore, in the steady state, only P-frames are used in encoding video conferencing content. This makes frame level bit-allocation simpler since there is no need to consider relative complexities between different types of frames during frame level bit-allocation. The key-frame at the beginning is handled using special cases.

As depicted in Figure 2.2, one of the inputs for the bitrate control module is a delay/latency parameter (L). This is defined as the maximum permissible delay allowed between the encoder and the decoder assuming zero transmission delay. In other words, the delay parameter is the maximum allowed time for any encoded frame to be transmitted completely through a constant bandwidth channel of pre-defined bitrate. In this work, delay parameter (L) is configured as,

$$L_0 = 165\,ms \text{ and } L = \frac{1.5 \times 1000}{framerate}. \tag{4.1}$$

Initially a delay of $L_0 = 165\,ms$ is allowed for the key-frame. This allows, allocation of higher than average bits for the key-frame. However, the delay parameter (L) for the P frames in steady state is only 1.5 times the frame sampling delay. For instance, if the input video is sampled at 30 fps, then the time interval between two consecutive frames is 33 ms (frame sampling delay), the permissible delay (L) for frames in steady state is approximately 49 ms. The usage of different delay values for the first key-frame and steady state P-frames is handled by changing the delay value gradually. The large key-frame at the beginning results in huge delay (165 ms), this delay is gradually reduced by using less than average bit-count for the subsequent few frames (half of average bit-count per frame). Once the over-consumption of the first key-frame is compensated, the steady state delay of $49\,ms$ is maintained for the rest of the sequence.

It should be noted that the initial delay in the system can only be reduced by displaying the initial few P-frames at shorter intervals than the time interval in which they were captured. This results in a momentary increase in the playback speed. In practical implementations, usage of an above average bit-count for the I-frame results in a few dropped frames subsequently even if the I-frame over-consumes marginally. All the above artifacts are considered as an acceptable trade-off to achieve good initial spatial quality by allocating a huge amount of bits to the first key-frame.

The bit-allocation module uses VBV fullness and delay parameter (L) to compute the bits allocated for the current frame to be encoded. The VBV fullness (d_0^n) before encoding the nth frame is calculated based on the size of the previously encoded $(n-1)th$ frame in bits ($FrameSize_{n-1}$) as follows,

$$\begin{aligned} d_0^n &= d_0^{n-1} + (FrameSize_{n-1} - AvgBitsPerFrame), \\ d_0^n &= max(d_0^n, 0), \end{aligned} \tag{4.2}$$

where,

$$AvgBitsPerFrame = \frac{bitrate}{framerate}.$$

The allocated bits for the nth frame is the maximum amount of bits that can be transferred along with the residual bits in the VBV in the duration L ($49\,ms$ in the above example). The maximum acceptable delay in ms (L) is translated to bits (L_{bits}) as,

$$L_{bits} = \frac{L \times bitrate}{1000}.$$

Therefore, the amount of allocated bits for the current frame (B_{alloc}) is given by,

$$B_{alloc} = L_{bits} - d_0^n. \tag{4.3}$$

In practice, rate control QP predictions are not very accurate to exactly consume the bits that were allocated to the frame (B_{alloc}). If a frame consumes more bits than B_{alloc}, it violates the delay conditions. The encoded frame will be unable to reach the decoder in time with the available bitrate. Hence, the frame is not added to the bitstream. The frames which are encoded but are not part of the output of the encoder are called *dropped frames*. Such dropped frames must be avoided since they cause jerky playback. A small room for the inaccuracy of the QP prediction is considered at the end of the bit-allocation stage to avoid dropped frames. In practice, the amount of target bits used for QP prediction is slightly smaller than B_{alloc} to avoid dropped frames in the case of marginal over-consumption of bits.

4.2 QP Prediction

Due to low VBV size, the bitrate control needs to have a very quick reaction to any deviation in the bitrate to avoid dropped frames. The bitrate control algorithm computes the QP for every macroblock. The two factors considered while computing QP for a macroblock are:

- Macroblock Complexity
- Bitrate Deviation

The macroblock complexity is used for adapting the QP according to the structural complexity of a macroblock. The deviation in bitrate at the macroblock level is computed based on the feedback from the encoder to achieve target bitrate with higher precision. Each of these factors are discussed in the following sections.

4.2.1 Macroblock Complexity

The structural complexity of the macroblock is used to compute the delta QP (dq) which is used to adapt QP at the macroblock level. The activity of the macroblock is a measure of the complexity of the macroblock and hence indicates the amount of bits required to encode the macroblock. After motion compensation with the selected coding mode and motion vectors, the activity of mth macroblock (act_m) with original pixel value $s_m(i,j)$ and predicted pixel value $c_m(i,j)$ is calculated as,

$$act_m = \sum_{i,j} \mid s_m(i,j) - c_m(i,j) \mid, \quad i,j = 1,2,....16. \tag{4.4}$$

The relative complexity of the macroblock with respect to the entire frame complexity is used in QP adaptation. The ratio of activity of the current macroblock and the average activity of the entire frame is used to calculate the delta QP as,

$$dq = \begin{cases} -floor(\frac{avg\_act}{act_m} - 1), & 0 < \frac{act_m}{avg\_act} <= 1/2. \\ 0, & 1/2 < \frac{act_m}{avg\_act} <= 2. \\ floor(\frac{act_m}{avg\_act}) - 1, & \frac{act_m}{avg\_act} >= 2. \end{cases} \tag{4.5}$$

Where, avg_act is the average activity across all the macroblocks of a frame. As depicted in the above equation, a positive dq is used when the current macroblock is relatively complex compared to the average frame complexity. This indicates that for relatively complex macroblocks within a frame, a higher QP is used. The purpose of QP adaptation is to account for spatial and temporal masking effect of HVS. Such activity based QP adaptation results in an uneven quality within a frame. The peak signal to noise ratio (PSNR) of the simple or static region is higher than that of the regions with high motion (foreground). In practice, the average frame activity of the entire frame is unavailable until the last macroblock of the frame has been encoded. Therefore, previous frame average activity is used as current frame activity since the two adjacent frames in a video are likely to remain similar.

The activity metric used in eq. (4.5) is a measure of the complexity of the macroblock, hence it can be replaced by similar metrics depicting the complexity of the block. Other metrics like SATD (Sum of Absolute Difference in Transform Domain) and cost of the macroblock (J) can be used instead of the activity. In this work, the cost of the macroblock computed during rate-distortion optimization (eq. (4.6)) is used as the complexity metric.

$$J = D + \lambda R, \tag{4.6}$$

where, the distortion D represents the residual error after prediction which is measured as the sum of absolute differences (SAD) between the original block and the reconstructed block, and is weighted against the number of bits R associated with the motion information using the Lagrange multiplier λ. The least cost of all the evaluated modes is considered

as the complexity of the block. The cost of the macroblock factors in both the amount of residual information to be encoded after motion compensation and bits used for signaling the mode and the motion vector. This makes it more accurate in terms of reflecting the complexity of the block compared to the activity computed in eq. (4.5).

4.2.2 Bitrate Deviation

The delta QP computed in eq. (4.5) is added to the QP calculated based on the deviation in the bitrate reflected by instantaneous VBV fullness. The buffer fullness corresponds to fullness of the VBV discussed in the context of leaky bucket model in Section 2.2. Any deviation in the bitrate will be reflected in the occupancy of the buffer. For example, a higher level of the buffer indicates over-consumption of bits. The VBV buffer occupancy (d_0^n) is calculated only after an entire frame is encoded. In order to account for the deviation in bitrate at the macroblock level, a global deviation factor is computed at the macroblock level. The global deviation is computed based on the deviation in frame level bit-consumption of past frames and the size of the macroblocks encoded in the current frame. The global deviation factor ($D_m^{n'}$) when encoding the mth macroblock of nth frame is given by,

$$D_m^{n'} = d_0^{n'} + CurFrameBitCount - \frac{B_{alloc} \times m}{M}. \tag{4.7}$$

Where, M is the total number of macroblocks in a frame, B_{alloc} is the bits allocated to the frame by bit-allocation module (eq. (4.3)) and hence remains a constant for the given frame. CurFrameBitCount is the bit-consumption of the current frame until the last encoded macroblock. The term $d_0^{n'}$ in eq. (4.7) is the accumulated frame level bit-deviation which is computed similar to the VBV fullness (d_0^n). Since the VBV fullness (d_0^n) is computed only after fully encoding the frame, the factor $d_0^{n'}$ also remains constant for the given frame. The two terms $d_0^{n'}$ and d_0^n differ with initialization values at the beginning of the encoding [SML02]. The frame level deviation factor ($d_0^{n'}$) is additionally subjected to clipping as shown in eq. (4.9) after encoding every frame.

The global deviation factor ($D_m^{n'}$) accounts for the deviation in bitrate of the encoded video until the last encoded macroblock. The global deviation factor is used to calculate the QP for the mth macroblock (Q_m),

$$Q_m = \frac{D_m^{n'} \times 31}{r} + dq, \tag{4.8}$$

where,

$$r = i \times bitrate/framerate.$$

The factor r, is called the reaction factor. This factor indicates the number of frames over which the deviation in bitrate is to be compensated. The bitrate control module in this work uses $i = 1$.

The working of bitrate control can be understood by analyzing the deviation factor $D_m^{n'}$ in eq. (4.8). At the beginning of the frame ($m = 0$),

$$D_m^{n'} = d_0^{n'}.$$

The allocated QP (Q_m) solely depends on $d_0^{n'}$ if the deviation in macroblock level bit-consumption and activity based delta QP (dq) are ignored. The initial value of $d_0^{n'}$ is chosen heuristically at the start of encoding based on the most commonly used configuration. Therefore, the QP calculated for the first frame is not content dependent. Once the first frame is encoded, the bit-consumption usually differs from the allocated bits by a large extent. For instance, if the content is more complex than average, the initial QP allocation will result in a large bit consumption, increasing the value of frame level bit deviation ($d_0^{n'}$) which results in a large global deviation ($D_m^{n'}$). This will result in larger QP value for the next frame to be encoded according to eq. (4.8). Therefore, the value of $d_0^{n'}$ oscillates for the first few frames. In steady state, it takes optimum value to keep the deviation low eventually helping in achieving the target bitrate. The frame level bit-deviation $d_0^{n'}$ is clipped between pre-computed maximum and minimum value to limit the QP to a suitable range.

$$d_0^{n'} = clip(1000, \frac{40 * r}{31}). \tag{4.9}$$

The QP output by the rate control (Q_m) is clipped between the valid range of QP allowed in H.264 encoding. In addition to these limits, the QP computed in eq. (4.8) is subjected to swing restrictions. Since the QP is modulated based on the activity of the macroblock, an upper limit of maximum QP (QP_{max}),

$$QP_{max} = QP_{avg} + 5, \tag{4.10}$$

is set to make sure the high activity regions are not excessively penalized with higher quantization. Here, QP_{avg} corresponds to the average QP of all the blocks in the previously encoded frame.

Chapter 5

Study Setup

This section describes the setup used in this work. It describes the configuration of the encoder used to evaluate different algorithms. It also describes the metrics and other aspects used to evaluate the proposed ROI-based bitrate control schemes and to compare its performance with the state-of-the-art bitrate control algorithm.

5.1 Sample Video Sequences

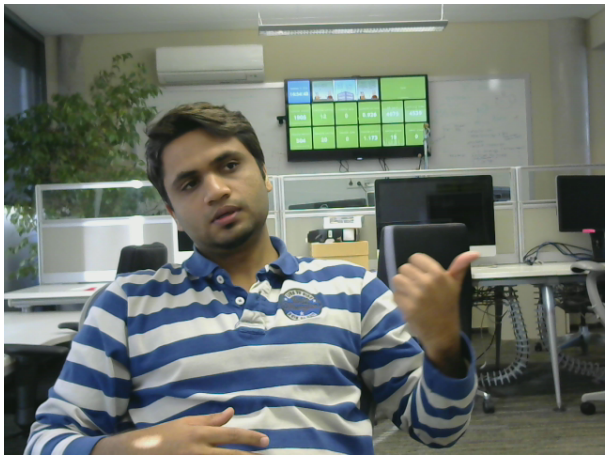
This section describes the sample video sequences chosen for evaluating the algorithms used for ROI-based encoding. A typical video conferencing scenario is considered in this work. The list of inputs and their characteristics are tabulated in Table 5.1.

Name	**Resolution**	**Framerate (Frames/second)**	**Total Frames**
Paul	640×480	30	1000
Chet	640×480	30	490
Johnny [Joh]	1280×720	30	300

Table 5.1: Sample video sequences and their properties.

The three chosen sample video sequences have different spatial and temporal complexities. The sample input videos are chosen to cover most of the typical video conferencing environments. The sample video sequence *Johnny* [Joh] is a commonly used content for evaluation of video coding algorithms. The sample video sequences *Paul* and *Chet* are chosen from Citrix database. The snapshots of the chosen sample video sequence is shown in Figure 5.1. Table 5.2 lists the relative spatial and temporal complexities across different contents. The table also lists the relative area of the face (ROI) within a frame throughout the given sequence. The relative ROI area (A_{roi}) is defined as,

$$A_{roi} = \frac{M_{roi}}{M},$$



(a) *Chet*, frame number 106

(b) *Paul*, frame number 119

(c) *Johnny*, frame number 35

Figure 5.1: Snapshot of sample video sequences (uncompressed).

where, M_{roi} is the number of ROI macroblocks and M is the total number of macroblocks in a given frame. The area of face region in *Chet* changes significantly within the sequence, other input sequences have almost constant A_{roi} across all the frames.

Name	Relative Complexity		A_{roi} (approx.)
	Spatial	Temporal	
Paul	Medium	Low	0.05
Chet	Medium	High	0.04 - 0.4
Johnny	Low	Low	0.10

Table 5.2: The relative spatial and temporal complexity comparison for the sample video sequences in Table 5.1.

5.2 Encoder Configuration

This work uses the Citrix H.264 video codec for studying different ROI-based encoding approaches. The encoder is configured in low-delay mode suitable for video conferencing and other real-time applications. The bitrate control described in Chapter 4 is used to control the output data-rate of the encoder. The encoder is configured to use IPPP mode with key/I-frame used only at the beginning of the sequence followed by only uni-directional P-frames. Due to the low-delay requirements there is no provision to re-encode the frame in case of buffer overflow. The frames are dropped entirely in case of a buffer overflow to maintain a constant low-delay.

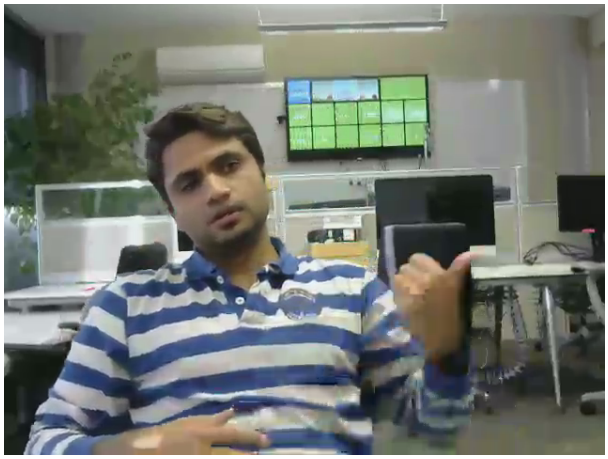
Name	Bitrate (Kbps)	VBV Size (Bits)
Paul	250	12500
Chet	250	12500
Johnny	350	17500

Table 5.3: Encoder configuration

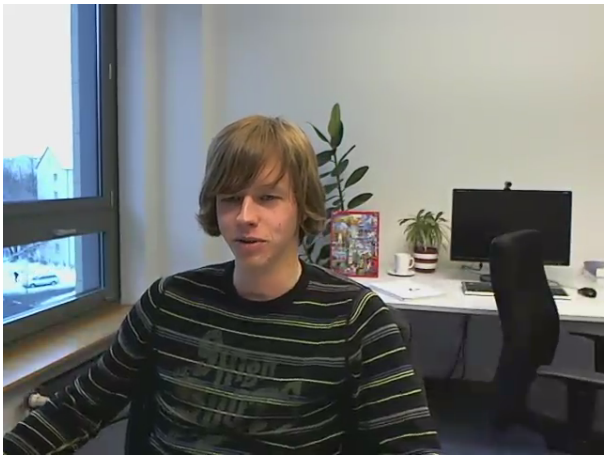
The configuration used for encoding each of the sample input sequences listed in the above section is shown in Table 5.3. The bitrates for different input sequences are configured considering the resolution and the complexity of the content. The chosen bitrates are relatively low yielding low visual quality with conventional encoding. This makes it easy to assess the gain in the perceptual quality with ROI-based encoding. Figure 5.2 shows the results of conventional encoding for the configuration shown in Table 5.3. Due to the usage of low bitrate, blocky artifacts can be noticed in the face regions.

5.3 Evaluation Criteria

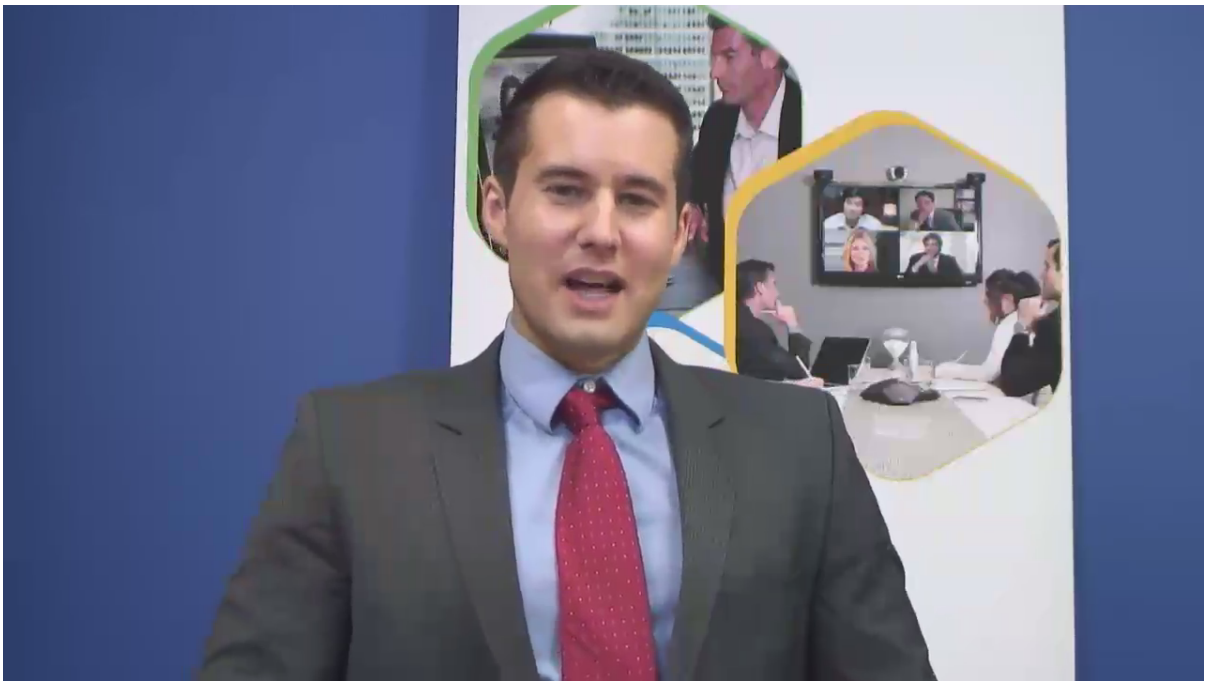
One of the crucial aspects in this study is the metric used to evaluate various approaches in order to choose the best approach. The goal of this study is to improve the quality of the ROI macroblocks at the cost of degrading the non-ROI macroblocks. Since the whole



(a) *Chet*, 250 Kbps



(b) *Paul*, 250 Kbps



(c) *Johnny*, 350 Kbps

Figure 5.2: Result of conventional encoding with corresponding bitrates.

approach is to measure the gain in perceptual quality, using frame level PSNR alone as a metric could be misleading.

In this study, the difference in average PSNR of frames and average ROI PSNR is used as one of the metrics to evaluate different approaches. The expectation is to see an improvement in ROI PSNR, with the degradation in the PSNR of non-ROI. The shift in quality of ROI and non-ROI parts should be achieved keeping the bitrate unchanged. Therefore, the second aspect is to measure the deviation in bitrate behavior with ROI-based encod-

ing compared to conventional encoding without using any ROI information. Third aspect involves analysis of PSNR and QP variation within a frame. The QP and PSNR distributions within a frame are studied to ensure that there is visible improvement in the quality of ROI without badly degrading the quality of non-ROI.

To measure all these behaviors the following metrics are considered. These are discussed in detail in the following sections.

- *Quality metrics* - PSNR of ROI and non-ROI parts.
- *PSNR and QP maps* - The map of variation of QP and PSNR within a frame.
- *Delay plot* - Plots time delay for every frame.

5.3.1 Quality Metrics

The increased bit-allocation to ROI macroblocks in ROI-based encoding results in an increase in the PSNR of ROI. This also results in a drop in the non-ROI PSNR. Finding the desirable magnitude of improvement in the PSNR of ROI along with an acceptable drop in the PSNR of the non-ROI is tricky. The idea here is to find the right balance between quality improvement in the ROI and the degradation of non-ROI as to achieve the maximum perceptual quality. The PSNR computed specific to the regions within a frame gives insight into the magnitude of transfer of bits from ROI to non-ROI during ROI-based encoding. This measure will also indicate the aggressiveness of an algorithm which is measured in terms of the magnitude of objective quality difference forced between ROI and non-ROI parts.

Table 5.4 shows the PSNR values for conventional encoding of the sample sequences listed in Section 5.1. It is clear that for most of the sample inputs, the PSNR of ROI is much lower compared to that of the overall frame PSNR. This is not desirable since the regions that matter the most to the perceptual quality have a lower PSNR on an average.

Content	PSNR Avg (dB)	PSNR ROI (dB)	PSNR non-ROI (dB)
Paul, 250 kbps	39.15	37.72	39.22
Chet, 250 kbps	31.35	32.91	31.24
Johnny, 350 kbps	37.93	36.08	38.13

Table 5.4: PSNR values for conventional encoding

The PSNR is calculated using the weighted sum of PSNR of individual components per picture ($PSNR_Y$, $PSNR_U$ and $PSNR_V$) [JOGJSW12].

$$PSNR_{YUV} = (6 \times PSNR_Y + PSNR_U + PSNR_V)/8, \tag{5.1}$$

where, individual components are computed as,

$$PSNR = 10 \times log_{10}((2^B - 1)^2/MSE), \tag{5.2}$$

where $B = 8$ is the number of bits per sample (bit-depth) of the video and MSE is the mean squared error.

The change in PSNR of ROI and non-ROI parts is measured as an average of PSNR of the entire frame and average of PSNR of ROI of all the frames in the sequence. The PSNR for the entire video sequence can be computed in two ways.

- *Average of frame PSNR* - This is the average of PSNR of all the frames in the video sequence.
- *Average MSE-based PSNR* - The PSNR of average MSE of all the frames in the sequence. This is computed by accumulating MSE over entire sequence and then computing PSNR.

The average of PSNR metric is preferred over average MSE based PSNR since the latter metric was found to be heavily influenced by the outliers. For instance, when the sequence has very few extremely low-quality frames but has good quality on an average, the average MSE based PSNR was found to be very low due to the presence of very few extremely low-quality frames.

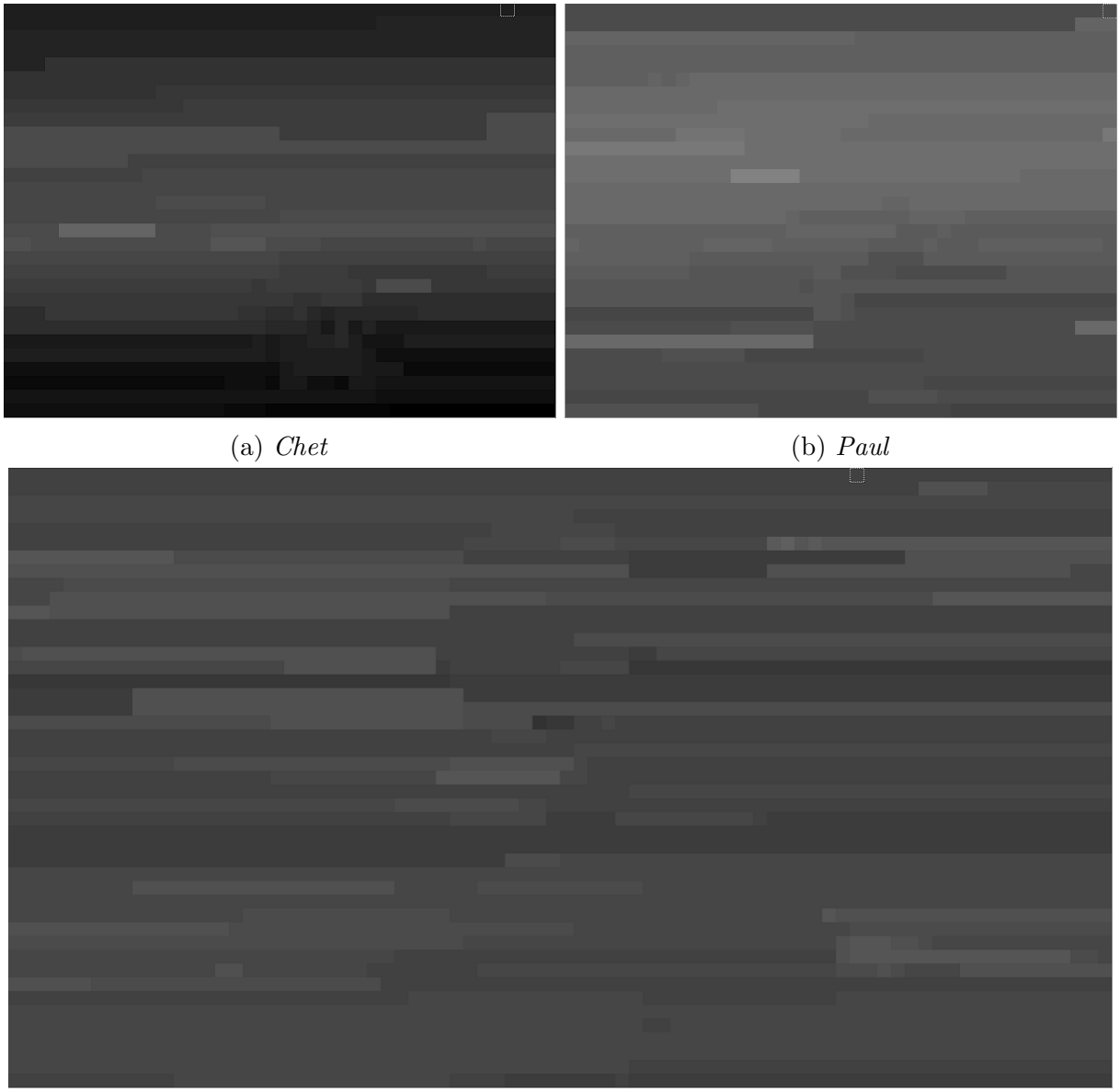
5.3.2 PSNR and QP Maps

The study of PSNR and QP distribution within a frame is important to understand the effect of movement of bits from ROI to non-ROI parts. The PSNR and QP values are extracted at the macroblock level. It is then stored in raster scan order which can be used to display as an image to compare the structure with that of the video frame. These values are illustrated in a gray scale image.

In a QP scale map, the darker regions in a frame indicate higher quantization. Even though the quantization parameter used for encoding a block is closely related to the PSNR of the block, it is not the only determinative factor. The PSNR can also vary depending on the content. Generally, the lower frequency regions have better PSNR even when encoded with a higher QP. The static regions of the frame also tend to have better PSNR even when higher QP is used because of less new information to be encoded. The PSNR map helps in visualizing the effect of movement of bits from non-ROI to ROI parts.

Figure 5.3 shows the quantization parameters used during conventional encoding for the frames in Figure 5.2. A darker shade of gray in this map indicates usage of a higher quantization parameter compared to the regions with a brighter shade of gray. The QP range of 1-51 is mapped to a gray scale value between 0 - 255 in the quantization maps. It can be noticed that since no information about the region of interest is used while encoding the frame, the pattern of quantization appears almost random. The shape of the original content is not recognizable from the quantization map.

Figure 5.4 shows the PSNR distribution for the frames in Figure 5.2 which are encoded



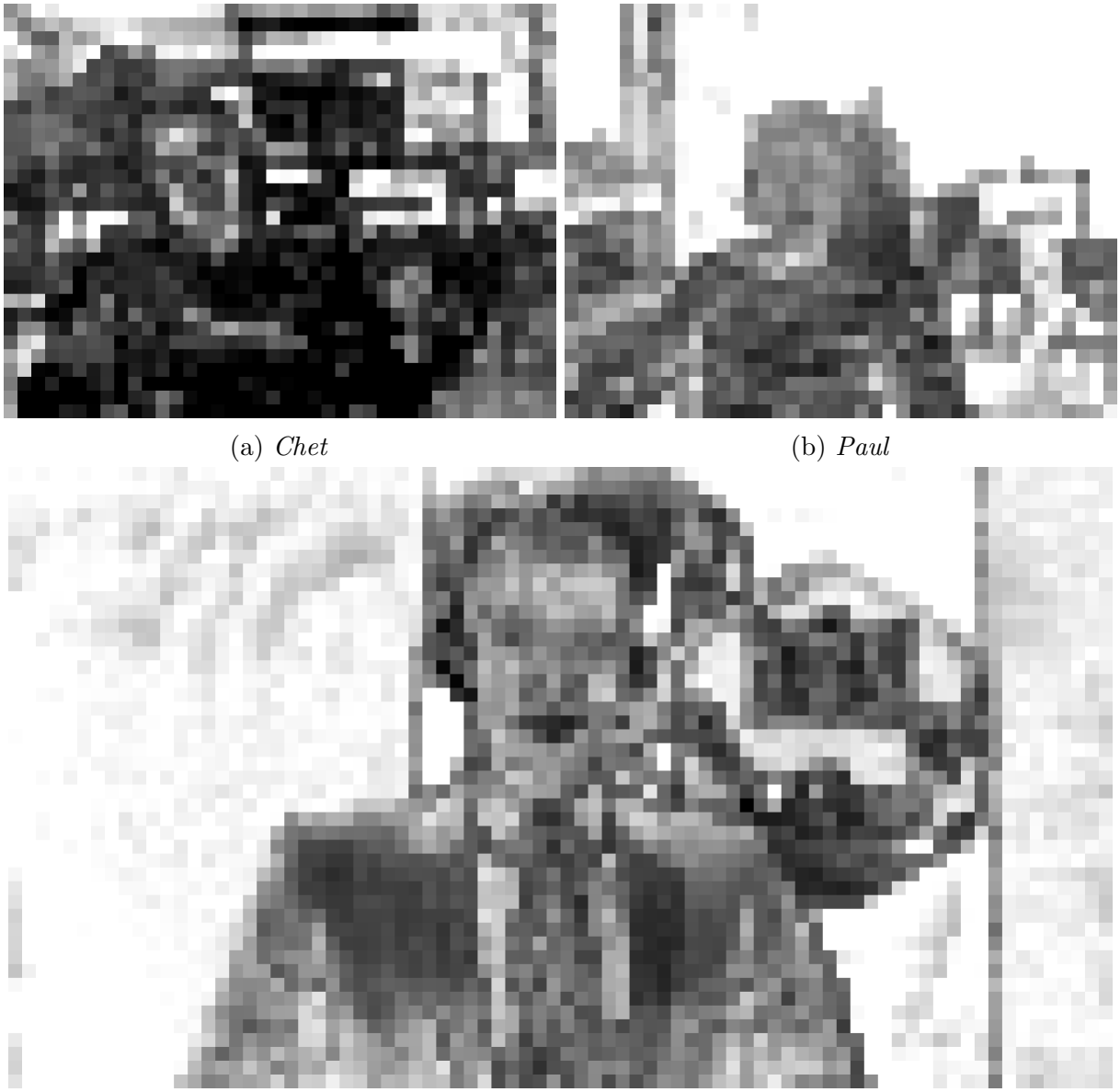
(a) *Chet* (b) *Paul*

(c) *Johnny*

Figure 5.3: Result of conventional encoding - Quantization maps (QP range of 1 - 51).

using conventional encoding. Similar to the quantization map, a brighter shade of gray represents the regions with higher PSNR, the darker regions indicate lower PSNR and worse quality. A PSNR range between 25 dB - 45 dB is linearly mapped to a gray scale value of 0 - 255 in the PSNR maps.

It is evident that the structure of the original content is preserved in the PSNR map. In most of the contents (*Johnny* and *Paul*), the background regions have better PSNR, the foreground has worse quality and the difference in quality is quite huge. The difference in



(a) *Chet* (b) *Paul*

(c) *Johnny*

Figure 5.4: Result of conventional encoding - PSNR maps (PSNR range of 25 dB - 45 dB).

the quality is due to the fact that the background in a video conferencing scenario is mostly static and hence gets encoded better with every frame. On the other hand, the foreground has motion and new data to be encoded with every frame, hence it cannot achieve the same quality as the background. Since the focus of attention during video telephony is foreground or the face region, improving the face region must help in improving the overall perceptual quality. The PSNR maps help to visualize the effect of such preferential encoding. The goal of ROI-based encoding is to boost the quality of the foreground (face region) to the

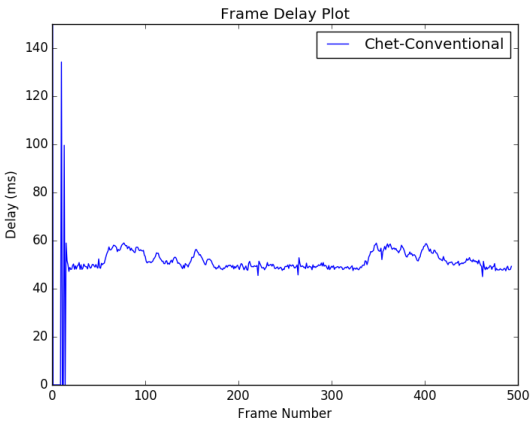
same level as the background or even better.

5.3.3 Delay Plot

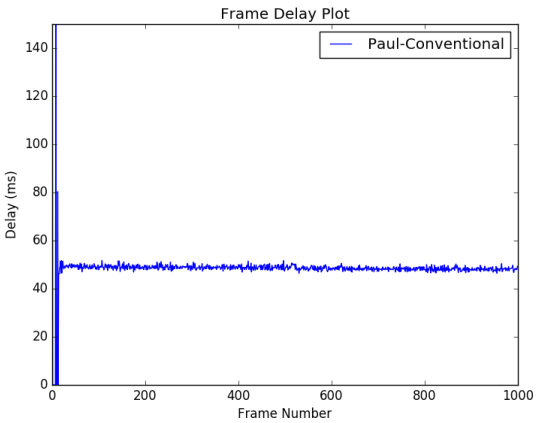
The fundamental idea in this work is to efficiently use the bits within a frame to encode ROI with better quality. The algorithms used to achieve this should not alter the behavior of the encoder in terms of frame level bit-consumption. As mentioned in the previous sections, in the case of VBV overflow, the encoder drops the frame in order to maintain strict VBV compliance. The dropped frames result in jerky playback and hence should be avoided. The ROI-based bitrate control scheme should not contribute to an increase in the number of dropped frames due to changes in the bit-consumption pattern.

In constant bitrate control, the size of an encoded frame decides the buffering delay of each frame. Therefore, any change in bit-consumption pattern at the frame level gets reflected in the buffering delay of the corresponding frames. The delay due to buffering of each frame in the sequence is plotted to analyze the bit-consumption behavior. Figure 5.5 is the delay plot of the sample video sequences encoded with the configuration shown in Table 5.3. Every point in the delay plot specifies the time taken by the corresponding frame (marked on the x-axis) to reach the decoder assuming zero transmission delay. It can be noticed that the delay is almost constant for *Johnny* and *Paul*. However, there is a slight variation for *Chet* due to high temporal complexity.

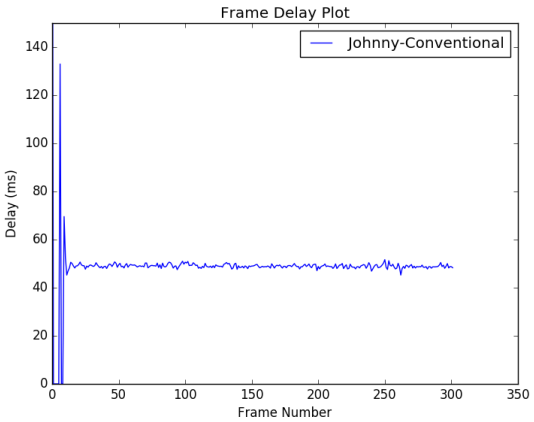
The curve in delay plots appears mostly smooth except for few sudden drops (zero values). These zero-valued points indicate dropped frames. Since these frames are not included in the final bitstream and hence not transmitted, the delay is indicated as zero. Ideally, the ROI-based encoding approaches should not alter the shape of this plot. The ROI-based bitrate control should re-distribute the bits within a frame with minimum error carried to the next frame. It is also not desirable to have any increase in the number of dropped frames.



(a) *Chet*



(b) *Paul*



(c) *Johnny*

Figure 5.5: Result of conventional encoding - Delay plots.

Chapter 6

Face Detection and Tracking

In video conferencing, face is the most important region in the video stream to the viewer. In this work, various approaches of improving the perceptual quality by increasing quality of the face region are presented. The prerequisite for these ROI-based encoding techniques is precise co-ordinates of the face (ROI) in any given video frame. This section describes the setup used to detect the face prior to encoding.

The face detection module is independent of the encoder and ROI-based encoding schemes discussed in this work. The face detection module uses input video stream to the encoder and marks the ROI at the macroblock level. Figure 6.1 shows the face map generated for the frame shown in Figure 5.1b. Each byte value in the face-map generated by the face detection module represents a macroblock scanned in raster scan order. The region in white is considered as ROI. This information is used by the bitrate control module of the encoder for ROI-based encoding.

Since every ROI-based encoding approach discussed in this work involves improving the quality of ROI at the cost of degrading the quality of non-ROI, it is very important to have high reliability with the face detection. Any false detection will lead to degradation of the actual ROI compared to conventional encoding. Therefore, the damage caused by a false detection is higher than the loss due to not detecting any face.

6.1 Viola-Jones Face Detection

Many open-source solutions like OpenCV offers a ready to use solution that can be integrated with the codec library. In this work, OpenCV implementation of Viola-Jones AdaBoost face detection algorithm [VJ01] is used. It has a large set of trained classifiers considering many types of faces and viewing angles. A standard frontal face Haar-classifier is used to detect faces in all the sample video sequences.

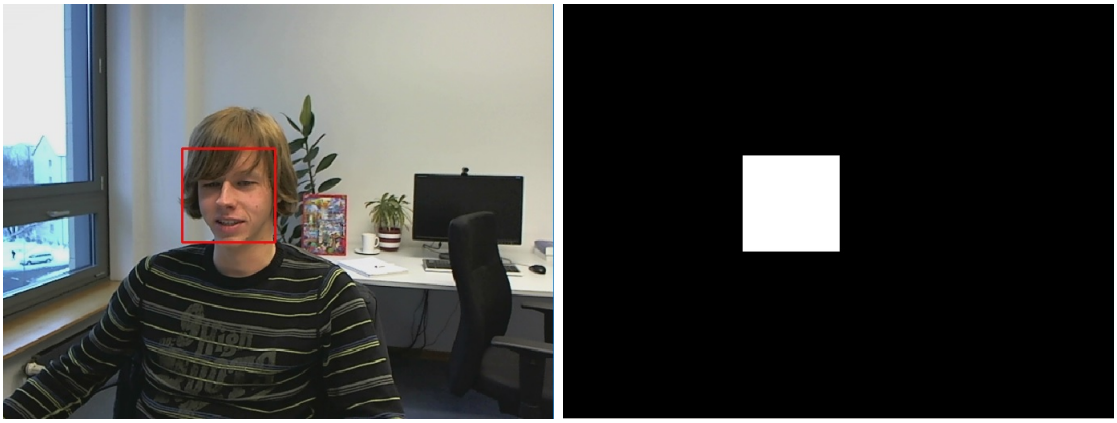


Figure 6.1: Face detection output for *Paul* and the corresponding face map. The white and black regions represent ROI and non-ROI respectively.

6.1.1 Performance Benchmark

In this work, face is detected independently in every frame of the video stream. This is used to achieve high accuracy face detection to evaluate ROI-based encoding techniques. This setup is not the most optimal solution to be used in a real-time system. Face detection is a computation intensive task and performing this task on every frame of the video stream demands huge computation power. Table 6.1 gives the measured performance of OpenCV based face detection on *Intel(R) Xeon(R) CPU E3-1225 V2 @ 3.20 Ghz*. The numbers mentioned in Table 6.1 also account for input raw file format conversion in addition to face detection. It is clear that the face detection module complexity increases almost linearly with increase in resolution. For a resolution of 1280×720, the processing is not real-time (assuming 30 fps video stream) with the specified computation resource. The following section proposes an optimization scheme to detect the face in real-time.

Name	Resolution	Frames/second
Paul	640×480	44.26
Vidyo4 [Joh]	1280×720	18.89

Table 6.1: OpenCV face detection performance benchmark on *Intel(R) Xeon(R) CPU E3-1225 V2 @ 3.20 Ghz*.

6.2 Optimization Using Motion Vectors

As mentioned above, it is not an optimal solution to invoke face detection module for every frame of the input video stream. Usually, face tracking techniques are used to track the face once it is detected. Some of the conventional approaches for tracking are mean-shift

and continuous adaptive mean shift (CAMSHIFT) [Bra98]. However, these techniques also demand a considerable amount of computation resource and requires additional effort to implement face tracking in addition to face detection.

6.2.1 Regular Interval Face Detection

An alternative approach is to detect the face at regular intervals in every *nth* frame instead of every frame of the video. Due to the high framerate of the input video stream, the movement of the face region is limited between the consecutive frames. The intermediate frames consider the previously detected face region as the face. In other words, the approach is to use the temporally subsampled video stream with sampling interval n as input to the face detection module. However, this approach of temporally subsampled face detection leads to inaccuracy which increases with the sampling interval between two frames in which the face is detected. The inaccuracy also increases with rapid movements in the face region.

Figure 6.2 shows a snapshot of the sample input content *Vidyo4* [Joh] with 30 frames per second with face detection at every frame and at regular intervals (temporal subsampling). The green bounding box represents the output with face detection on every frame ($n = 1$) and the red bounding box represents face detection performed on every 15th frame ($n = 15$). It is clear that there is a large inaccuracy with temporally subsampled face detection. For sampling interval $n = 15$, majority of the face region lies outside the bounding box. This can degrade the performance of ROI-based encoding since the region most important to the perceptual quality is considered as non-ROI. The error due to temporal subsampling is measured as the euclidean distance (pixels) between the top-left corner pixels of the two bounding boxes. The error is always computed with respect to the green bounding box ($n = 1$). The error with different sampling intervals for face detection is tabulated in Table 6.2.

Sampling interval for face detection (n)	**Average error (pixels)**	**Maximum error (pixels)**
5	8.06	57.55
10	16.89	112.21
15	23.27	137.29

Table 6.2: Inaccuracy due to face detection at regular intervals (n) for the sample content *Vidyo4*. The error is computed with respect to the output with face detection performed on every frame ($n = 1$).

During video conference, face region is mostly static except for movement of lips and eye regions. The face detection at regular intervals is not an optimal solution due to the following reasons.

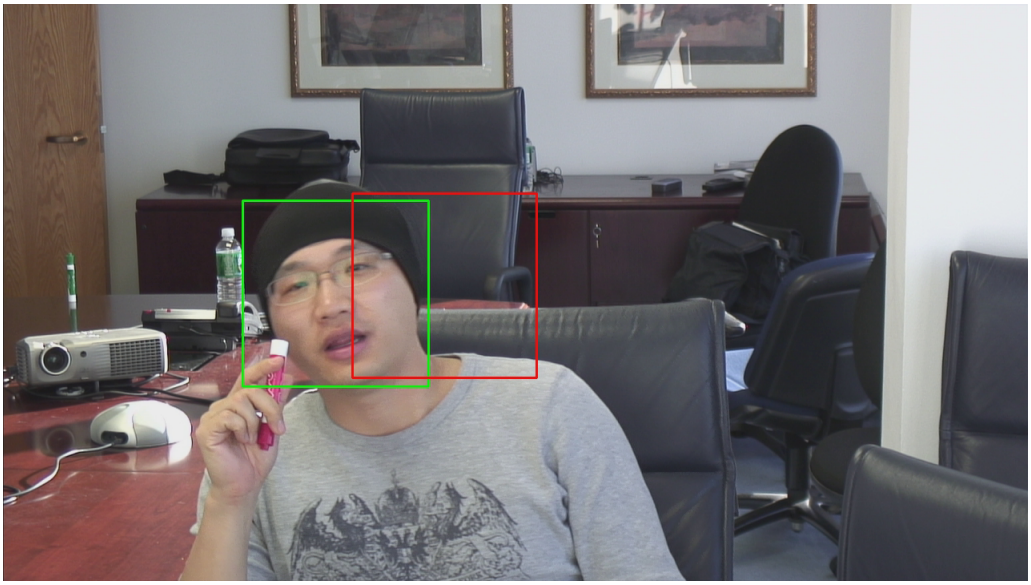


Figure 6.2: Snapshot of *Vidyo4* with face detection at regular intervals (temporal sub-sampling). Green and red bounding boxes represent the output of face detection performed on every frame and every 15th frame respectively.

- The face detection module is invoked even when the face region is static. This is a wastage of precious computation resource.
- The error during sharp movements in face region is very high (Table 6.2).

In order to address the above disadvantages, this work proposes an alternative approach in which motion vector information computed during motion estimation stage of video encoding is reused to detect the movement of face to invoke face detection instead of detecting the face in every *nth* frame.

6.2.2 Face Movement and Motion Vector

Any movement in the face region is reflected in the motion vectors of macroblocks in the corresponding regions. The magnitude of these motion vectors is directly proportional to the magnitude of movement in the face region. Figure 6.3 shows the predictability of sharp movements in the face region with average magnitude of motion vectors of the face region. The series in blue represents the difference between top-left corner pixels of the bounding boxes of detected face in a given frame and its previous frame. The face detection is performed on every frame of the video stream ($n = 1$). The series in red represents the average magnitude of motion vectors in the face region of any given frame predicted from its previous frame. It is clear that all sharp movements in the face region have correspondingly high magnitude motion vectors. This shows that sharp movements in the face region are easily predictable from the motion vectors of the corresponding regions.

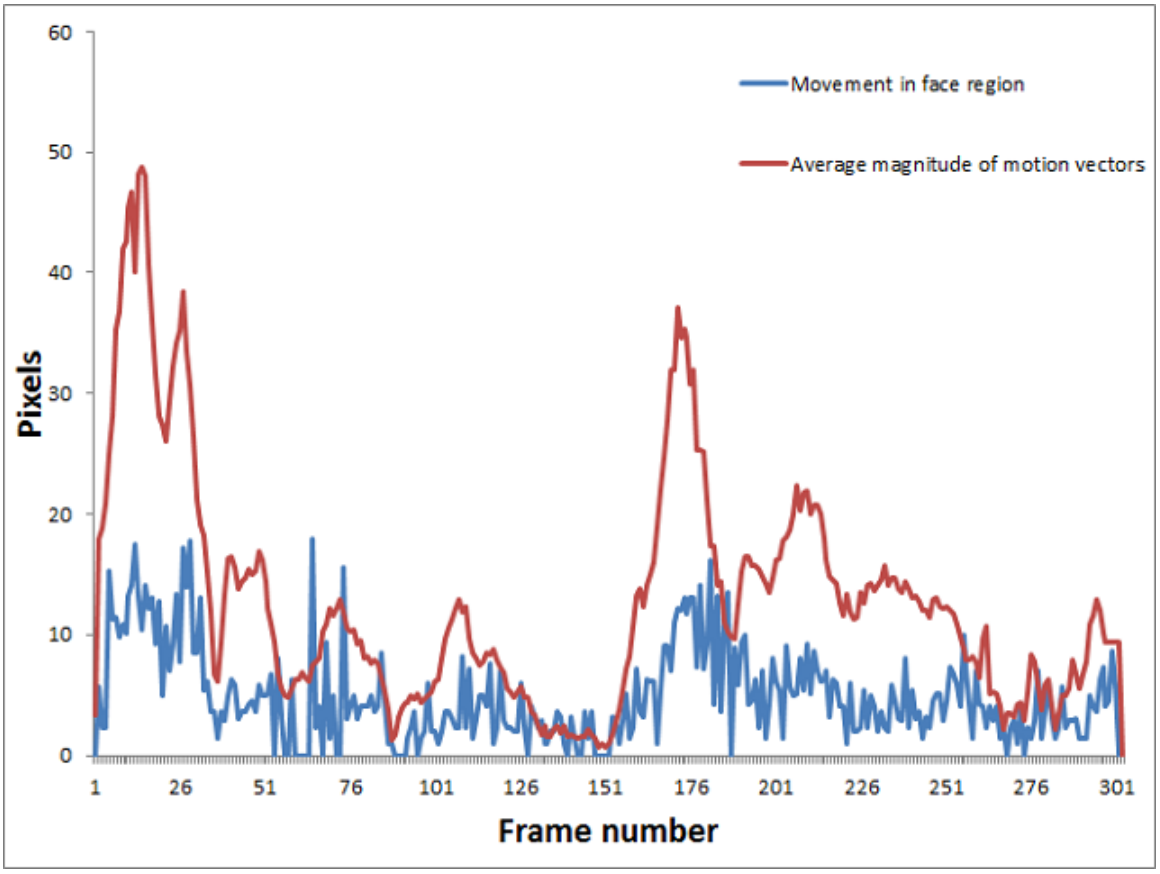


Figure 6.3: Comparison of the movement of the face region between consecutive frames (blue) and the corresponding average magnitude of motion vectors in the face region (red).

6.2.3 Motion Vector Based Variable Interval Face Detection

The predictability of sharp movements of the face region using motion vectors can be used to find the optimal interval to perform face detection. The idea is to perform face detection when sharp movement in the face region is detected using motion vectors. This does not alter the face detection or tracking algorithm used, but it invokes these modules only when the movement in the face region is detected. The algorithm used to perform variable interval face detection based on motion vector is as follows.

1. Perform face detection on the first frame.
2. Consider the previously detected face region as face in the subsequent frames until face detection is performed again.
3. For every frame, compute the average magnitude of motion vectors of all the macroblocks belonging to the face region. If this value is greater than a predetermined threshold (T_H), perform face detection on the current frame.

4. It is possible that face region can slowly drift with the average motion vector less than T_H on every frame. To correct the accumulated error due to such gradual movement of the face region, face detection is performed after a large number of frames (N) even if no motion is detected. Since the movement of the face region is very slow in this case, a large sampling interval N can be used such that $N >> n$.

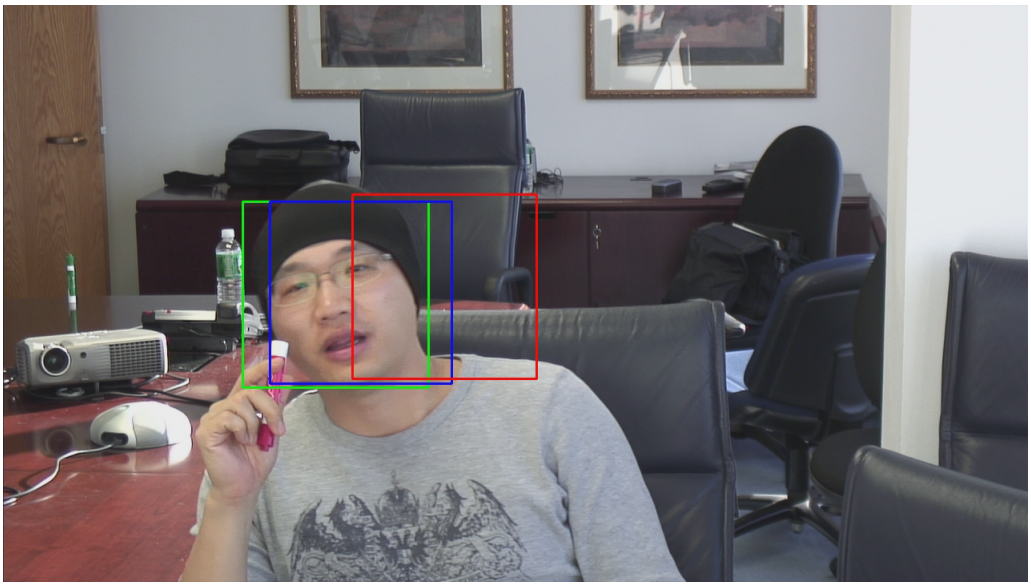


Figure 6.4: Face detection at regular intervals as shown in Figure 6.2 along with motion vector based variable interval face detection (blue bounding box).

The technique described above will avoid the wastage of computation resource by avoiding face detection or tracking when the face region is highly static. This will also reduce the error due to face detection with temporal subsampling since face detection is performed immediately after considerable movement of the face region is detected. Figure 6.4 shows the same frame shown in Figure 6.2 with motion vector based variable interval face detection (blue bounding box). It is clear that the inaccuracy in face detection is much lower compared to the approach where face is detected in every 15th frame ($n = 15$).

The comparison of inaccuracies with regular interval face detection with $n = 15$ and motion vector based variable interval face detection is given in Table 6.3. Table 6.4 gives average processing time required for face detection in 30 frames of input (1 second). It is clear that face can be detected in a video stream in real-time with the proposed techniques. The results show a reduction in average error and peak error by 25% and 56.3% respectively. The total number of face detection performed in a sequence of 300 frames for both the approach is almost the same. This implies that the computation resource required for face detection with these approaches remain almost the same. The chosen sample content has very high motion in the face region which results in higher number of face detection performed based on the motion vector information. For a typical video conferencing stream, the number of face detection is expected to be much lower compared to regular interval

Face detection interval	Total number of face detection	Average error (pixels)	Maximum error (pixels)
Every frame, $n = 1$ (reference)	300	0	0
Regular interval, $n = 15$	21	23.27	137.29
Motion vector based variable interval	23	17.45	59.93

Table 6.3: Performance of regular interval and motion vector based variable interval face detection for the sample content *Vidyo4* with 300 frames.

Face detection interval	Average processing time for face detection (seconds)
Every frame, $n = 1$ (reference)	1.588
Regular interval, $n = 15$	0.105
Motion vector based variable interval	0.115

Table 6.4: Average processing time required for face detection for 30 frames (1 second) of sample content *Vidyo4* (resolution 1280×720) measured on *Intel(R) Xeon(R) CPU E3-1225 V2 @ 3.20 Ghz*. The table shows that real-time performance can be achieved using proposed techniques.

face detection. However, the error due to temporal subsampling is greatly reduced with motion vector based variable interval face detection. The advantages and disadvantages of using motion vector based variable interval face detection instead of regular interval face detection are summarized below.

Advantages

- The average error due to not performing face detection on every frame is greatly reduced.
- The ROI-based encoding techniques are highly sensitive to maximum error in face detection. This is reduced by a factor greater than two since sudden motion always triggers face detection.
- For a typical video conferencing stream, the total number of face detection performed will be lower compared to regular interval face detection. This reduces the overall computation requirement making it easier to achieve real-time performance.
- This approach is based on the reuse of motion vectors computed during video encod-

ing. Therefore, it has very low computation complexity.

Disadvantages

- *Unpredictable load* - In motion vector based variable interval face detection, the number of face detection performed for a given sequence depends on the magnitude of motion in the face region. This varies across different contents. Therefore, the processing load due to this approach is not predictable.
- *Higher peak load* - Motion vector based variable interval face detection can trigger too many face detection in a small interval due to very high motion in the face region.

Chapter 7

ROI-based Bitrate Control

As discussed in the previous sections there are many ways in which ROI information can be used to improve the perceptual quality of the video. This section describes the following two major approaches of ROI-based bitrate control to enhance the quality of the ROI macroblocks.

- ROI QP Offset
- ROI-based Bit-Allocation

These approaches are designed considering the bitrate control module described in Chapter 4. However, the underlying principles of ROI-based encoding to create an optimal quality difference between ROI and non-ROI are applicable to any generic bitrate control modules. The principles presented in this work can be used to modify other standard low-delay bitrate control algorithms to produce equivalent results. The different approaches presented here vary in terms of the ease of implementation, complexity and output quality. This offers flexibility to choose a suitable approach according to the specific requirements. The following subsections describe each of the ROI-based bitrate control approaches.

7.1 ROI QP Offset

A simple way of creating a bias in the quality between ROI and non-ROI is by using a QP offset for the macroblocks belonging to the ROI. A negative QP offset (dq_{roi}) is added to the QP allocated by the bitrate control module (Q_m) for ROI macroblocks. The QP for a macroblock assigned by the bitrate control is modified before using it for final encoding as,

$$Q_m^{roi} = Q_m - dq_{roi},$$
$$Q_m^{nroi} = Q_m,$$

where, Q_m^{roi} and Q_m^{nroi} are the QP used for encoding a macroblock belonging to ROI and non-ROI parts respectively. The QP offset is used outside the bitrate control module, hence this approach requires minimum or no modifications to the bitrate control module.

The effects of usage of external QP offset for ROI macroblocks on encoding of non-ROI macroblocks are:

- A lower QP for ROI results in ROI macroblocks consuming more bits.
- The rate control module obtains feedback from the encoder regarding bit-consumption at the macroblock level (Chapter 4). This feedback signals over-consumption of bits by ROI macroblocks resulting in a higher deviation factor ($D_m^{n'}$).
- The bitrate control reacts to the usage of reduced QP for ROI macroblocks by increasing the QP of non-ROI blocks to compensate for the additional bits used by ROI macroblocks.
- This results in the frame level bit-consumption very similar to conventional encoding without using any ROI-based QP offsets. Therefore, the effect of ROI-based QP offset is mostly neutralized within the frame.
- Any error in bit-consumption at the frame level is compensated in the subsequent frames.

The feedback from the encoder to the bitrate control module will ensure that the target bitrate is successfully achieved. The bit-allocation based on buffer level (Section 4.1) will ensure that a constant delay is maintained even after using the QP offset. This approach offers the simplest way of implementing quality bias between ROI and non-ROI parts without breaking the core functionality of the bitrate control module.

7.1.1 Conventional and ROI-based Encoding Comparison

The images in Figure 7.1 shows the comparison between conventional encoding and ROI-based encoding using QP offset and their corresponding attributes like PSNR map and quantization map. A QP offset (dq_{roi}) of -4 was used during ROI-based encoding. The image in Figure 7.1b looks better in the face regions compared to Figure 7.1a due to the usage of a negative QP offset for ROI. This improves the overall perceptual quality of the video. Figures 7.1e and 7.1f shows the effect of QP offset on the final QP used for encoding the macroblock. The macroblocks in the face region of 7.1f have lower QP which is seen as macroblocks with a lighter shade of gray compared to other macroblocks in the frame.

The PSNR maps in Figures 7.1c and 7.1d shows that the PSNR of the face region is improved considerably with ROI-based encoding using QP offset. There is no major difference in PSNR of the background region (non-ROI). The face appears sharper due to the additional boost in quality from reduced QP. The overall bitrate of both the compared

(a) (b)

(c) (d)

(e) (f)

Figure 7.1: Comparison of conventional encoding and ROI-based encoding with QP offset of -4 for ROI. The figures $(\mathrm{a}, \mathrm{c}, \mathrm{e})$ and $(\mathrm{b}, \mathrm{d}, \mathrm{f})$ correspond to conventional encoding and QP offset based ROI encoding approaches respectively. (a, b) are snapshots from the encoded output of *Chet*. (c, d) are PSNR maps (PSNR range of 25 dB - 45 dB) and (e, f) are quantization maps (QP range of 1 - 51) corresponding to the frames in (a) and (b).

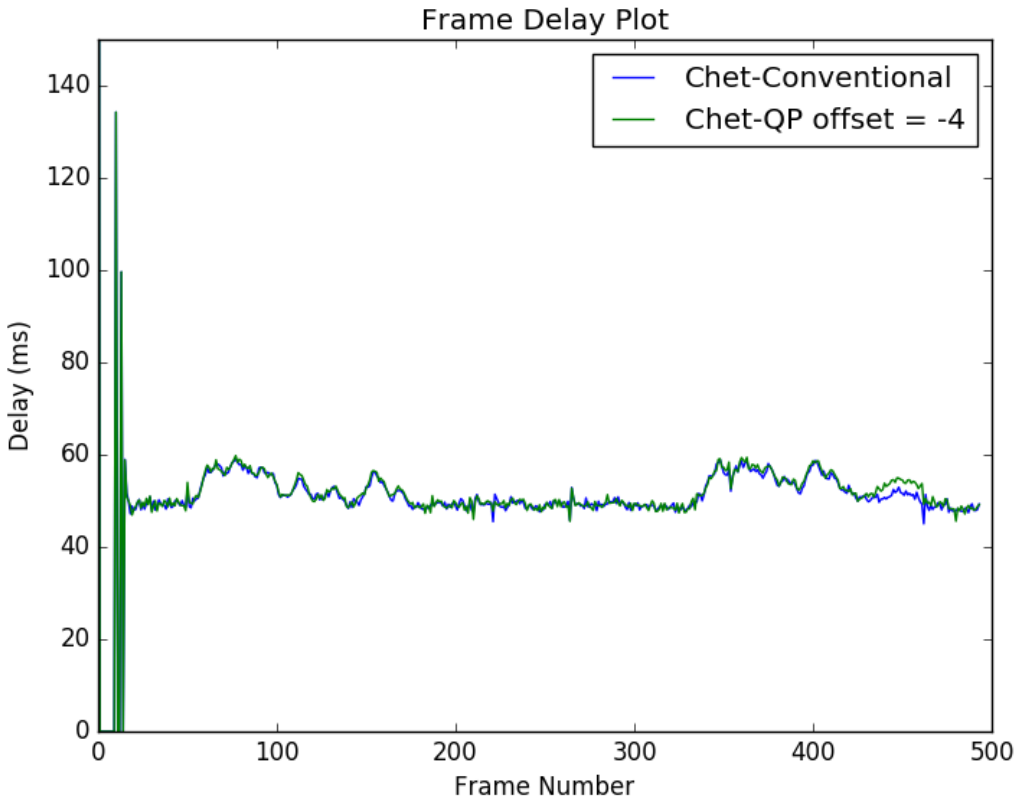


Figure 7.2: Delay plot for conventional encoding and ROI-based encoding with QP offset of -4 for ROI (Purple - Conventional encoding, Green - QP offset based ROI encoding).

bitstreams remained almost equal. The difference in the perceptual quality is only due to the movement of bits from non-ROI to ROI macroblocks.

Figure 7.2 shows the delay plot described in Section 5.3.3 for both conventional and ROI-based encoding using QP offset. The visibility of a single color predominantly (due to overlapping) shows that there is no significant change in the bit-consumption at the frame level. This implies that the additional bits consumed by the ROI macroblocks are compensated in non-ROI blocks within the given frame. There is very less difference in bit-consumption at the frame level that is carried over to the next frame. There is no significant change in the number of dropped frames which are represented as zero points in the delay plot. The number of dropped frames with conventional and ROI-based encoding remains the same. The dropped frames also appear at the same time interval in both the output bitstreams.

The PSNR of ROI-based encoding with QP offset is tabulated in Table 7.1. There is a considerable improvement in the PSNR of ROI compared to conventional encoding (Table 5.4). There is a corresponding drop in the PSNR of non-ROI which is also reflected in the reduced overall frame PSNR. The drop is not significant leading to an overall increase

QP Offset	Content	PSNR (dB)		
		Frame	**ROI**	**non-ROI**
−4	*Chet*, 250 kbps	31.22	34.27	31.04
−8	*Chet*, 250 kbps	30.87	35.51	30.64
−12	*Chet*, 250 kbps	30.39	36.54	30.13

Table 7.1: PSNR Comparison for QP offset based ROI encoding using different QP offsets.

in perceptual quality.

The boost in ROI PSNR is dependent on the magnitude of QP offset used for the ROI macroblocks. A QP offset of −4 might not be an optimum QP offset for all the contents. Therefore, it is necessary to consider the following aspects during QP offset computation to achieve optimal PSNR for ROI and non-ROI parts resulting in an improved perceptual quality.

- *Tuning QP offset* - Understanding the effect of using different QP offsets.
- *Area of ROI* - The relative area of ROI and non-ROI parts in a frame.
- *Bi-direction QP offset* - A positive QP offset is used for non-ROI blocks along with the negative QP offset for ROI.

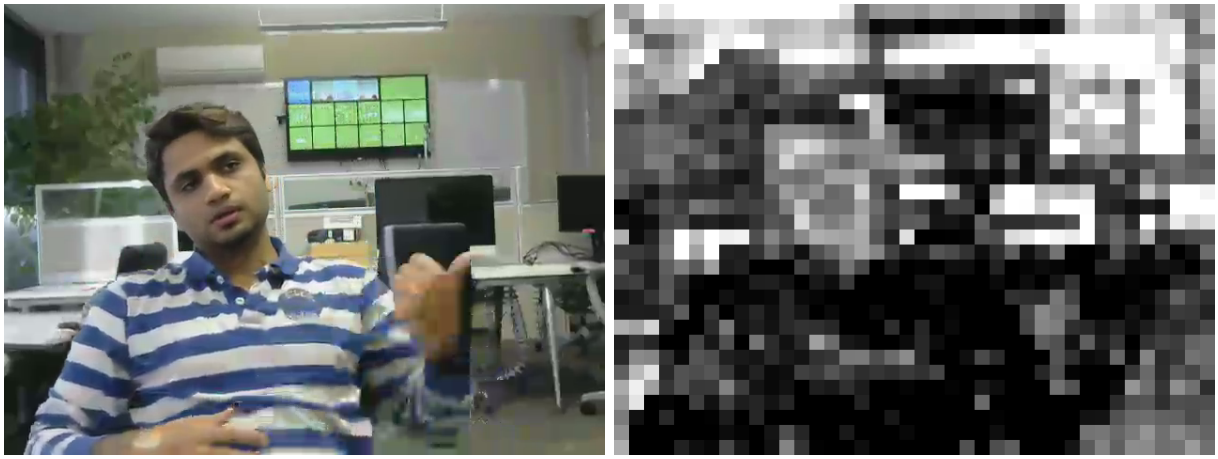
The following sections discuss each of these aspects in detail to form a generic QP offset computation approach applicable to most of the contents.

7.1.2 Tuning QP Offset

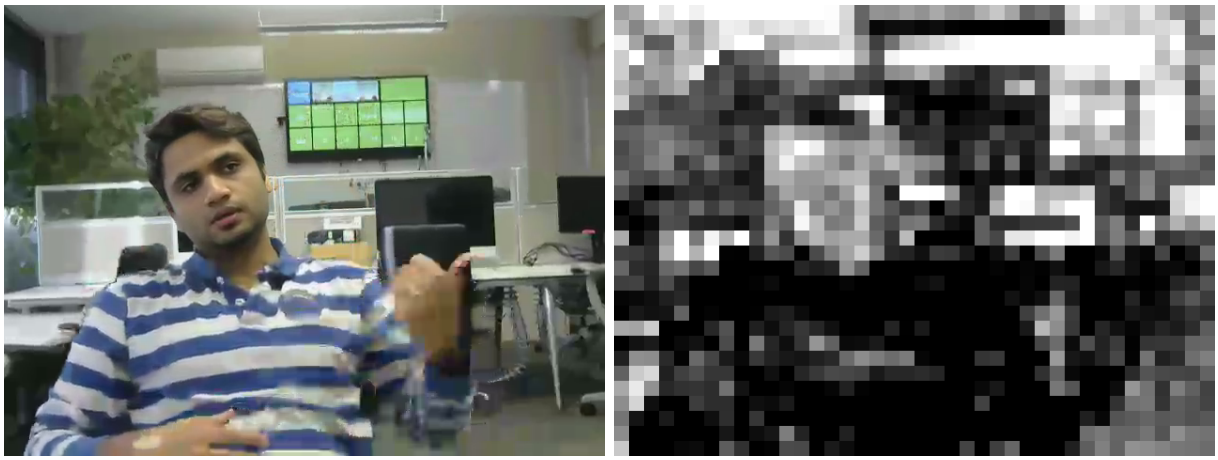
As mentioned earlier, the ROI-based encoding approaches discussed in this work only aim to re-distribute the bits within a frame based on ROI. The magnitude of re-distribution should be carefully chosen to avoid degradation of the background to an extent that artifacts become noticeable to the viewer even though those regions are not of primary importance to the viewer. An ideal redistribution of bits will make sure that there is a maximum transfer of bits from non-ROI part to ROI part without creating any visible artifacts in the non-ROI parts of the image.

In this work, various offsets are used to study the effect of magnitude of QP offset on the perceptual quality. The results of ROI-based encoding with QP offset of −4 shows favorable results with increased perceptual quality. This section examines the impact of increasing the magnitude of QP offset further.

The snapshot of same sample input encoded with QP offset of −8 shown in Figure 7.3a has a sharper face region compared to QP offset of −4 shown in Figure 7.1b. This is expected since the face region is coded with a lower QP due to a larger negative QP offset. The blockiness in the non-ROI, specifically around the arm region has increased. The increase in



(a) QP offset = −8



(b) QP offset = −12

Figure 7.3: Snapshots of *Chet* encoded with QP offset for ROI (Left) and the corresponding PSNR maps with a range of 25 dB - 45 dB (Right) for different QP offsets.

sharpness in the face region masks this blockiness to some extent. However, with further increase in the magnitude of QP offset the blockiness in the background becomes more prominent. The image in Figure 7.3b is encoded with a QP offset of −12. At this point, the increase in sharpness in the face region is masked by extreme blockiness in non-ROI, specifically around the arm region. The corresponding PSNR plots show extreme low PSNR in the non-ROI macroblocks with increasing QP offsets. Therefore, a very high QP offset can have an adverse effect on the perceptual quality.

QP Swing Restriction

It is possible to preserve the quality of the non-ROI macroblocks even with high QP offsets by using QP swing restriction discussed in Section 4.2.2. The maximum value of

ROI QP Offset	Encoding Mode	
	QP Swing Restriction = on	**QP Swing Restriction = off**
0	16	12
−4	19	12
−8	31	12
−12	45	14

Table 7.2: Number of dropped frames with different QP offsets for ROI for the sample sequence *Chet* with 490 frames.

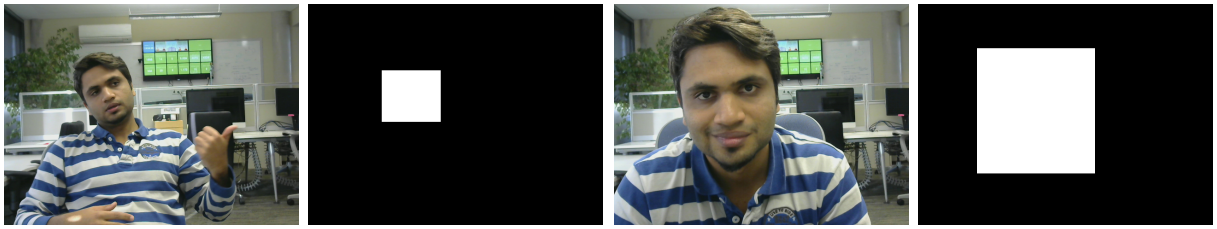
macroblock QP (QP_{max}) given by eq. (4.10), avoids the excessive degradation of the non-ROI macroblocks due to the usage of large QP offset. The QP of the non-ROI macroblock is not allowed to go very high despite over-consumption of bits by ROI blocks. The side-effects of increased QP offset shows up in the form of an increase in the number of dropped frames. Table 7.2 shows the number of dropped frames in the encoded video with 490 input frames. There is a drastic increase in the number of dropped frames with QP offset of −8 and −12. This increase in dropped frames reduces the smoothness of the playback which is annoying to the viewer.

When the QP swing restriction is turned off, the number of dropped frames with increased QP offset reduces drastically (Table 7.2). There is no considerable increase in the number of dropped frames compared to the output of conventional encoding. However, the quality of non-ROI blocks drops significantly without QP swing restrictions as shown in Figure 7.3. The extreme blockiness in the non-ROI decreases the perceptual quality. Therefore, using large QP offsets can lead to less perceptual quality either due to increased dropped frames or excessive blockiness in non-ROI depending on the configuration of QP swing restriction. Therefore, the QP offset for the ROI should be tuned not only considering the degradation of the background quality but also by assessing any other side-effects like an increase in the number of dropped frames.

7.1.3 Area of Region of Interest

This section discusses the importance of considering the relative area of ROI with respect to the non-ROI parts (A_{roi}) in computing the QP offsets. The variation in QP offsets discussed in the above section corresponds to a frame with relatively smaller ROI area compared to the whole frame. In this frame, 64 macroblocks out of a total of 1200 macroblocks belong to the face region (ROI). Therefore, ROI is less than 5 percent of the entire video frame ($A_{roi} < 0.05$). For a small percentage of ROI blocks, it is possible to use a large QP offset (very low ROI QP) since there are a large number of non-ROI macroblocks to compensate for the over-consumption of bits by ROI macroblocks.

However, in a video conferencing scenario, based on the focal length of the camera and



(a) (b)

Figure 7.4: Snapshots and the corresponding face maps of the sample content *Chet* (uncompressed), (a) frame number 106 with $A_{roi} = 0.067$ and (b) frame number 446 with $A_{roi} = 0.35$.

distance of the participant from the camera, area of the face in the video frame can change significantly. The number of ROI macroblocks can also change within a given input video sequence. Figure 7.4 shows the variation in the size of ROI within a video sequence. This demands continuous adaptation of the QP offset to avoid visual artifacts due to the usage of wrong QP offset. For instance, when the area of ROI greater than the area of non-ROI ($A_{roi} > 0.5$), usage of a higher QP offset will lead to huge over-consumption of bits by ROI due to an increased number of ROI macroblocks causing severe degradation in the quality of non-ROI macroblocks. In order to avoid severe degradation of the background, the magnitude of the QP offset should be inversely proportional to the ratio of the number of ROI blocks to non-ROI blocks.

The algorithm implemented to adapt QP offset uses the relative area of ROI. The QP offset is calculated using the linear relationship described in eq. (7.1). This is a heuristic approximation which was found to yield the best perceptual quality across many different video contents. The scaled QP offset is then clipped to a value of -6 to avoid large QP offsets which can lead to side-effects discussed in the previous section.

$$\begin{aligned} dq_{roi'} &= -round\Big(\frac{M}{M_{roi} \times 3}\Big), \\ dq_{roi} &= clip(dq_{roi'}, -1, -6), \end{aligned} \tag{7.1}$$

where, dq_{roi} is the offset used for ROI blocks, M is the total number of macroblocks in the frame and M_{roi} is the total number of macroblocks marked as ROI. The negative sign in the equation implies that the calculated offset is negative, which results in QP lower than non-ROI blocks. It is also evident from eq. (7.1) that a minimum QP offset ($dq_{roi} = -1$) is used when ROI area is more than two-third of the whole frame. The magnitude of QP offset increases linearly with subsequent decrease in the area of ROI.

7.1.4 Bi-direction QP Offset

The QP offset computed so far is only applied to the ROI macroblocks. The bitrate control module is responsible for compensating the additional bits used in encoding the ROI blocks by increasing the QP of non-ROI blocks.

As explained in Section 4.2.2, the bitrate control uses feedback from the encoder to constantly react to any deviation in the bitrate at the macroblock level. This feedback loop is not aware of the QP offset which is applied externally to the QP computed by the bitrate control (Q_m). Due to this, the bitrate control reacts to over-consumption of bits by increasing the QP of non-ROI blocks only after encoding the ROI macroblocks. This effect is clearly seen in the quantization map of ROI-based encoding in Figure 7.1f.

The macroblocks encoded immediately after ROI have larger QP (depicted as a darker shade of gray). Therefore, this approach does not increase the QP of non-ROI blocks uniformly. The non-ROI blocks encoded before ROI blocks have no increase in QP since reaction by the rate control to increased deviation in bitrate is not triggered until the ROI macroblocks are encoded. The non-ROI macroblocks encoded after ROI blocks tend to have lower quality than the non-ROI blocks encoded before the ROI blocks. This non-uniform loss of quality in non-ROI blocks is not desirable for good perceptual quality.

The non-uniform increase in non-ROI QP also depends on the position of the ROI within the video frame. For instance, consider a content where ROI part in the frame falls in the bottom right corner of the frame. Assuming that macroblocks are encoded in raster-scan order, the over-consumption of bits by the ROI part cannot be compensated within that frame. The error is carried over to the next frame. This alters the frame level bit-consumption behavior and can lead to increased number of dropped frames in extreme cases.

A bi-directional QP offset is used to avoid the behavior described above. The non-ROI blocks are assigned with a positive QP offset, which can compensate the over-consumption in ROI blocks. The non-ROI blocks are encoded with a higher QP even before ROI blocks are encoded. Since the non-ROI blocks from the start of the frame are encoded with a higher QP, there will be a surplus of bits available which can be used in encoding ROI blocks. In an ideal scenario, the negative and positive QP offsets must negate each other's effect resulting in frame level bits being unchanged had the frame been encoded without any offset.

The study in [MB13] suggests that for the frame level bit-count to be constant, the average QP of the frame must remain unchanged before and after adding the offsets. This is an observation made after multiple experiments. The QP offset for non-ROI macroblocks used to compensate the negative QP offset used by ROI is given by,

$$dq_{nroi} = \frac{M_{roi} \times dq_{roi}}{M - M_{roi}}, \tag{7.2}$$

where, dq_{nroi} is a positive QP offset used for non-ROI blocks when negative QP offset of dq_{roi} is used for ROI blocks (eq. (7.1)).

Inference

The result of QP offset based ROI encoding is discussed in Section 7.3.1. It should be noted that the term 'QP offset based ROI encoding' or 'bi-direction QP offset approach' refers to the QP offset scheme with all the improvements discussed in the previous Sections 7.1.2 to 7.1.4.

In this approach, the QP offset is applied outside the bitrate control. Therefore, there is less possibility of other factors like buffer fullness and global deviation ($D_m^{n'}$) overriding the QP offset. Such guaranteed QP offsets will ensure a boost in the ROI quality at all circumstances. However, this approach has many side effects due to the forced QP offset. The main disadvantage of this approach is not considering the characteristics of the input video content to decide the QP offset. Any given input with a given ratio of the number of ROI blocks to non-ROI blocks will have the same QP offset irrespective of the content. The QP offset computation is empirical which will work for most of the contents. However, it is not guaranteed to give optimal results for all contents.

The next section introduces an alternative approach for ROI-based encoding using ROI-based bit-allocation scheme. Some of the disadvantages of the QP offset based approach are addressed in the second approach.

7.2 ROI-based Bit-Allocation

This section introduces an approach with modifications to the bit-allocation module for an improved ROI-based encoding. The main drawback of the QP offset based approach is that the input video content characteristics are not considered for determining the QP offset. The difference in the complexity of ROI and non-ROI macroblocks determines the magnitude of quality difference required for good perceptual quality. This magnitude of optimal quality difference varies across different contents depending on the variation in complexity within a frame. For instance, consider a sample input video with complex background with some temporal complexity (non-ROI) but a simple static foreground (ROI). The usage of heuristic QP offsets which assumes the background to be mostly static will result in annoying blocky artifacts in the background. This might decrease the perceptual quality to a level below that of conventional encoding. Therefore, it is important to consider the relative complexities of ROI and non-ROI parts to compute the desired quality difference between these regions.

In conventional encoding, the bit-consumption is not uniform across all the macroblocks even though all regions of a frame are considered equally important to the viewer. This

is due to the variation in complexities across different macroblocks. Figure 7.5 shows the relative macroblock size in bits for a conventionally encoded frame. This map is generated by mapping the range of macroblock sizes of a given frame to a gray-scale image (0-255). Similar to quantization and PSNR maps, a brighter shade of gray represents larger bit-consumption. It is clear that the background regions consume the least amount of bits and the face region consumes an above average amount of bits. A bit-allocation strategy for ROI-based encoding which allocates higher than average amount of bits to ROI based on arbitrary weights [LT05, YLS08b] might not allocate any additional bits compared to the conventional encoding. For ROI-based encoding, the bit-allocation strategy should take into account the relative complexities of ROI and non-ROI and allocate bits with a bias over the complexity based bit-allocation. The fundamental idea behind this approach is to allocate bits to ROI and non-ROI proportional to their complexities with an additional bias for ROI.

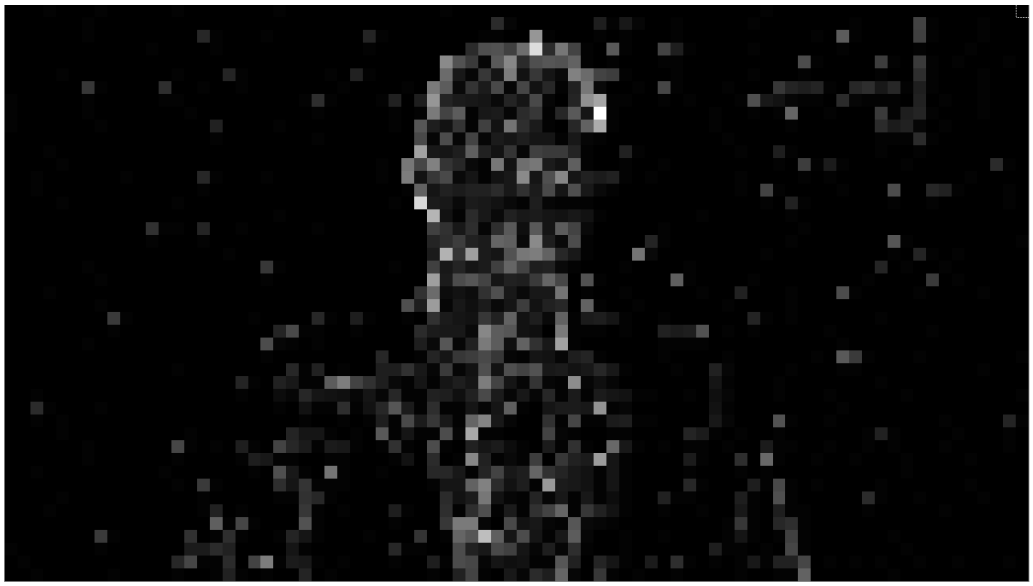


Figure 7.5: Result of conventional encoding - Macroblock level bit-consumption map (range minimum MB size - maximum MB size) for *Johnny* (frame number 35).

In this approach, the bits allocated at frame level (B_{alloc}) is split between ROI and non-ROI parts based on the relative complexities of these regions. The cost of a macroblock computed during rate-distortion optimization (eq. (4.6)) is accumulated for ROI and non-ROI regions to get the relative complexity (C_r),

$$C_r = \frac{\sum_{i=0}^{M_{roi}} J_i}{\sum_{i=0}^{M-M_{roi}} J_i},$$

where, J_i is the RDO cost of the macroblock i. The cost of a region determines the bits consumed by the corresponding region. The relative bit consumption of ROI and non-ROI

is predicted assuming that,

$$C_r \propto B_R, \tag{7.3}$$

where,

$$B_R = \frac{B_{roi}}{B_{nroi}}.$$

Here B_{roi} and B_{nroi} are the accumulated bit-consumption of M_{roi} macroblocks belonging to ROI and M_{nroi} non-ROI macroblocks respectively. The above equations hold true for any two regions in a given frame given that these two regions are encoded with equal importance (conventional encoding). Therefore, B_{roi} and B_{nroi} corresponds to the bit-consumption of two regions if the frame was encoded conventionally without using any ROI information. The idea is to predict the bit-consumption in conventional encoding and allocate bits for ROI and non-ROI regions according to the estimate but with an additional bias for the ROI part.

The ROI-based bit-allocation involves the following major steps.

- *Relative Bit-consumption Prediction* - Predict the relative bit-consumption of ROI and non-ROI (B_R) in conventional encoding.
- *ROI and non-ROI Bit Allocation* - Split the allocated frame level bits (B_{alloc}) between ROI and non-ROI parts according to B_R with an additional bias for ROI parts.
- *Region-based bitrate control* - Independent bitrate control for ROI and non-ROI regions.

The following sections discuss each of the above steps in detail.

7.2.1 Relative Bit-consumption Prediction

The first step in ROI-based bit-allocation is to determine the relative bit-consumption of ROI and non-ROI parts (B_R) if the frame was encoded without any bias for ROI (conventional encoding). There are two possible ways of achieving this:

1. Encode a frame using conventional encoding without any bias for ROI, measure the bit-consumption for ROI and non-ROI. The frame can be re-encoded with a bias in bit-allocation for ROI.
2. Estimate the complexities of ROI and non-ROI parts and use a cost-bits model to predict relative bit-consumption.

The first approach is very inefficient and not suitable for real-time encoding scenario. Therefore, a cost-bits model is used to predict B_R.

In this work, the relationship between the complexity of a macroblock and its bit-consumption during conventional encoding is studied. The RDO cost given by eq. (4.6) is

chosen as the complexity metric for prediction of B_R. The computation of a new complexity metric for predicting bit-consumption is not desirable due to increased complexity. The RDO cost which is computed by the encoder during rate-distortion optimization stage is reused to predict B_R.

Effect of RDO Cost on Bit-consumption

The cost of a macroblock affects its bit-consumption in two opposite ways listed below.

- The cost of a macroblock reflects the complexity of the macroblock. A macroblock with a higher cost consumes more bits (eq. (7.3)).
- The RDO cost is also used in the bitrate control for delta QP prediction as discussed in Section 4.2. The delta QP computation (eq. (4.5)) allocates a higher QP to the macroblock with a higher cost for spatial and temporal masking considering HVS. This will marginally reduce the bit-consumption of the macroblock.

These conflicting influence of cost on bit-consumption makes its prediction more complex. In this work, the effect of delta QP on bit-consumption is ignored for simplicity. The cost-bits model is developed assuming constant QP across all the macroblocks in a given frame.

Cost-Bits Model

A model to relate cost of a macroblock and its bit-consumption is developed using data from multiple video conferencing sequences. Figure 7.6 shows the plot of cost vs bits at the macroblock level. Every point on this plot corresponds to bit-consumption of a macroblock and its corresponding cost computed during the RDO stage. The plot corresponds to all the macroblocks in the encoded video sequences with 50 frames each from four different video conferencing samples. The sample input sequences used for evaluation (listed in Section 5.1) are not included for developing the model for a fair evaluation. The video sequences used for developing the model and the ones used for evaluating the model are different. The encoding for this plot was done in constant QP mode with $QP = 35$ for all the macroblocks. The data of the first key frame is not included in this plot since the first key frame is encoded without using any ROI information. It can be noticed that there is a linear relationship between predicted macroblock cost and the corresponding bit-consumption.

The linear relationship between cost of a macroblock (J) and its corresponding bit-consumption (b) at a constant QP is given by,

$$J = 18.2 \times b + 693. \tag{7.4}$$

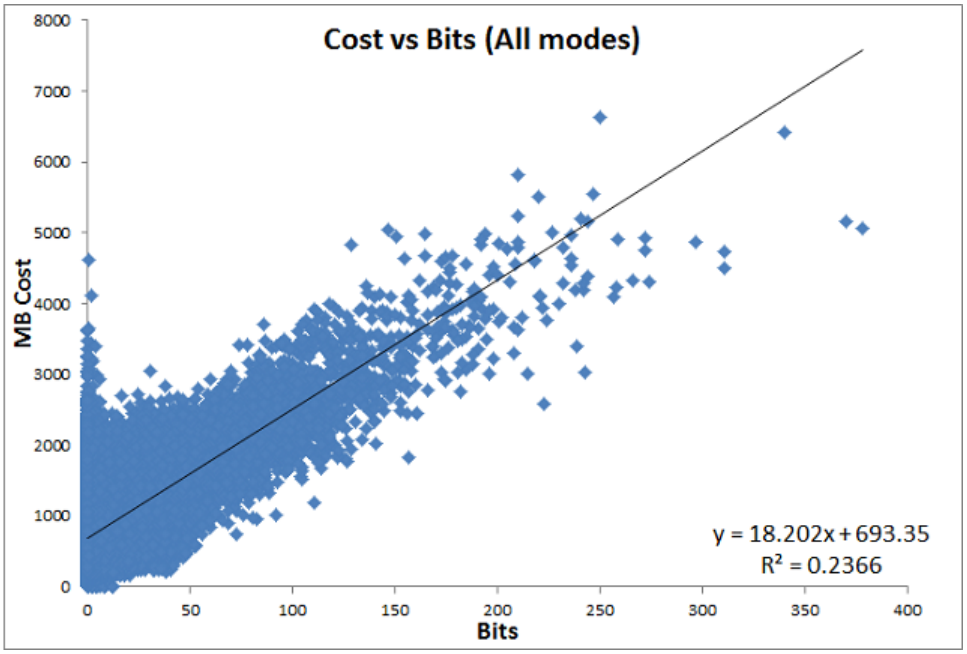


Figure 7.6: Cost vs bits plot for all the macroblocks of multiple video sequences encoded with constant QP ($QP = 35$).

The above equation is obtained by performing linear regression on the entire data-set shown in Figure 7.6. It has a corresponding R^2 value of 0.2366. The R^2 value (also displayed on the plots) is a measure of correlation between actual data and the predicted data using the linear relationship. A value of $R^2 = 1$ indicates perfect correlation. The R^2 value in this case is low to form an efficient prediction model. The correlation between macroblock cost and its bit-consumption is increased further by considering the prediction mode of the macroblock.

Cost-Bits Model - Prediction Modes

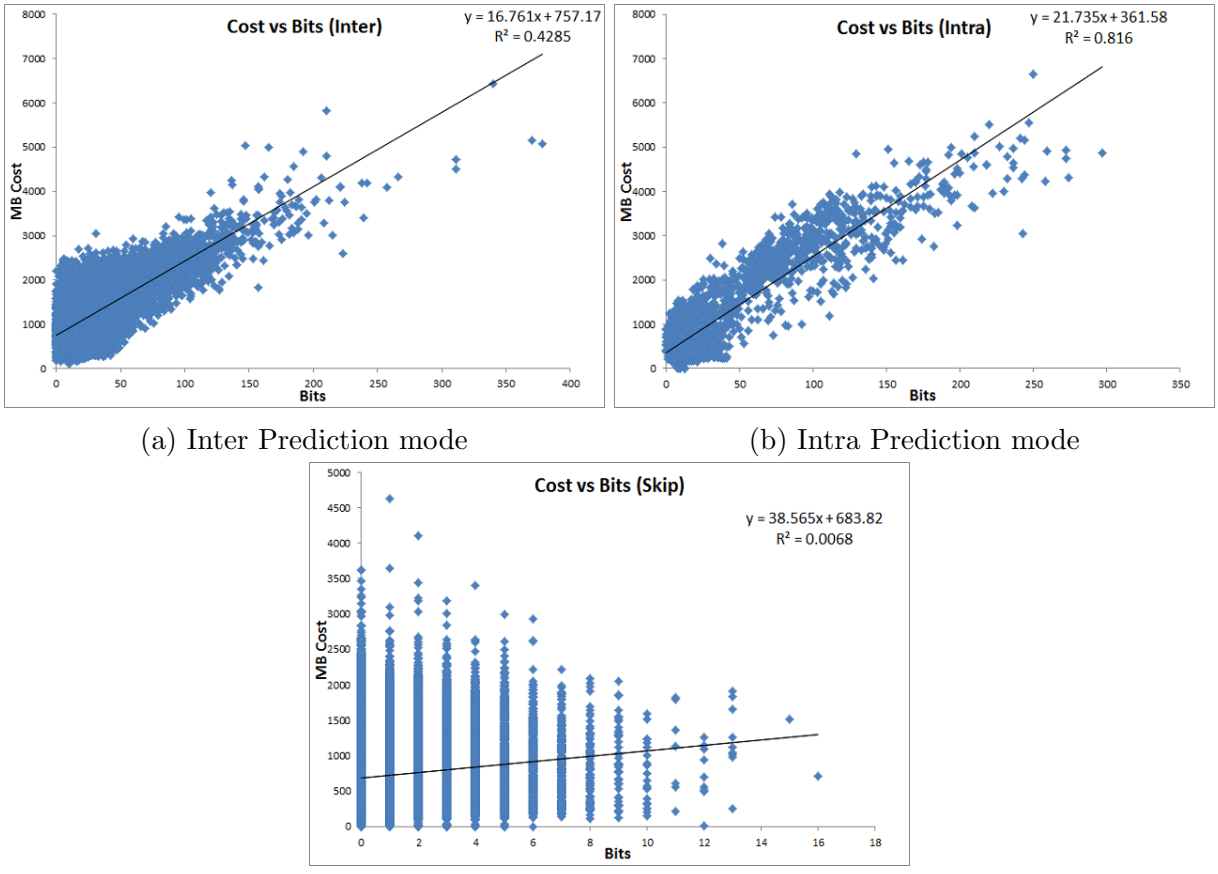
The data used in Figure 7.6 includes data for macroblocks of all types of prediction modes like skip, intra and inter prediction. It was observed that the predictability of bit-consumption using RDO cost was heavily dependent on the prediction mode used for encoding the macroblock.

The analysis of the macroblock cost and its corresponding bit-consumption specific to a prediction mode is shown in Figure 7.7. The plots in Figure 7.7 are generated using the same data as shown in Figure 7.6. Different plots are generated for skip, inter and intra prediction modes. It can be noticed that the correlation between the macroblock cost and its bit-consumption is maximum for intra prediction ($R^2 = 0.8757$) followed by inter prediction ($R^2 = 0.2995$). However, the bit-consumption pattern was found to be almost independent of the cost for skip-mode macroblocks ($R^2 = 0.0412$). This is because

the skip-mode decision happens in a later stage of encoding when there are no transform coefficients generated for a macroblock encoded with intra or inter prediction. The cost associated with the skip macroblock corresponds to the prediction mode chosen before the macroblock was decided to be encoded as skip macroblock.

Prediction Mode	Occurrence Probability	R^2 value for Cost-bits model
Intra	0.23	0.816
Inter	0.28	0.428
Skip	0.69	0.006

Table 7.3: The probability of prediction modes in constant QP ($QP = 35$) encoded sequences and its corresponding R^2 value.



(a) Inter Prediction mode

(b) Intra Prediction mode

(c) Skip mode

Figure 7.7: Cost vs bits plots for all the macroblocks of multiple video sequences encoded with constant QP ($QP = 35$) specific to the mode of encoding. (a) inter prediction, (b) intra prediction, (c) skip mode.

The data in Table 7.3 shows the probability of choosing a given prediction mode. It is clear that skip mode is the most commonly used prediction mode for the encoder configuration

discussed in Section 5.2. This is due to the low target bitrate and mostly static background in video conferencing sample contents. Therefore, it is very important to have good bits vs cost prediction model for skip macroblocks to compute B_R with a reasonable accuracy.

Fortunately, the extremely low correlation between macroblock cost and its bit-consumption for skip mode can be easily handled by assuming a constant number of bits for skip mode and ignoring its cost. The average amount of bits consumed by the skip macroblocks over the entire data-set was found to be 0.458 bits. This is extremely low bit-consumption compared to the intra and inter mode macroblocks. Therefore, a constant bit-estimate (average bits) is used in eq. (7.3) to compute the relative bit-consumption of ROI and non-ROI based on the number of skip macroblocks in each of these regions. The R^2 value of the prediction model is improved considerably by this assumption for the skip macroblocks.

The steps involved in computing the relative bit-consumption of ROI and non-ROI (B_R) are summarized below.

- Identify the ROI and non-ROI macroblocks in the input frame.
- Get the number of intra, inter and skip macroblocks in the ROI and non-ROI parts separately. This can be done using data from the previous frame.
- Compute the relative complexities of the ROI and non-ROI parts by accumulating costs of macroblocks in these regions.
- The relative complexity (C_R) is translated to relative bit-consumption (B_R) between ROI and non-ROI using prediction models shown in Figure 7.7. A previously trained cost-bits model is used for intra and inter blocks. A pre-computed value is used for the skip macroblocks to account for the bits consumed by the skip macroblocks.

The procedure involved in using (B_R) to allocate bits to ROI and non-ROI parts with a bias for ROI parts to perform ROI-based encoding is discussed in the subsequent sections.

7.2.2 ROI and non-ROI Bit-allocation

This section gives an overview of the process involved in splitting the allocated frame level bits (B_{alloc}) as computed in eq. (4.3) between ROI (B^{roi}_{alloc}) and non-ROI (B^{nroi}_{alloc}) parts. This is done using the relative bit-consumption factor (B_R) computed in the previous section. The idea behind ROI-based bit-allocation is to estimate relative bit-consumption between ROI and non-ROI in conventional encoding, perform bit-allocation according to the estimate with an additional bias for ROI. The predicted relative bit-consumption between ROI and non-ROI in conventional encoding is given by B_R computed in the previous sections. For ROI-based encoding, a bias factor k is introduced to create bias in bit-allocation. The bias factor signifies the importance of ROI over non-ROI. The bits allocated for ROI region

is given by,

$$B^{roi}_{alloc} = k \times B_R \times B^{nroi}_{alloc} \tag{7.5}$$

and

$$B^{nroi}_{alloc} = B_{alloc} - B^{roi}_{alloc}.$$

Here k is the ROI bias factor. In conventional encoding where all the parts of the frame are considered equally important, $k = 1$. For ROI-based encoding, a value $k > 1$ is used to bias bit-allocation to allocate more bits to ROI considering its importance to the perceptual quality. The bias factor (k) gives the flexibility of tuning the bias for ROI based on the importance of ROI.

As mentioned earlier, the ROI-based bit-allocation is independent of the procedure involved in determining the relative bit-consumption factor (B_R). The approach presented in Section 7.2.1 needs computation of the relative complexities of ROI and non-ROI. A simpler approach is to assume that the frame is uniformly complex and splitting the allocated frame level bits (B_{alloc}) uniformly between ROI and non-ROI with a bias factor specified in eq. (7.5). Therefore, if the content information is not considered for simplicity, the bits allocated for ROI and non-ROI is computed using eq. (7.5) with,

$$B_R = \frac{M_{roi}}{M - M_{roi}}.$$

The bias factor k determines the additional bits allocated to the ROI macroblocks. The main advantage of encoding with ROI-based bit-allocation compared to ROI-encoding based on QP offset is that it gives more control to guarantee minimum quality for the background macroblocks (non-ROI). For instance, the value of the ROI bias factor (k) can be bounded in such a way that non-ROI macroblocks are allocated at least half the amount of bits allocated during conventional encoding. This ensures that the quality of non-ROI does not deteriorate to an extent that blockiness in the background reduces the perceptual quality.

In this work, the value of k is chosen to first allocate non-ROI half of the bits it would have consumed in conventional encoding. The excess is allocated to ROI parts. The maximum excess bits allocated to ROI is three times its bit-consumption in conventional encoding. The excess is allocated back to the non-ROI parts. Therefore, the value of k changes depending on the complexities of ROI and non-ROI.

7.2.3 Region-based Bitrate Control

In the previous sections, the procedure for splitting the allocated frame level bits (B_{alloc}) between ROI and non-ROI is described. The input to this stage is bits allocated for ROI (B^{roi}_{alloc}) and non-ROI (B^{nroi}_{alloc}) parts. The target is to achieve bit-consumption according to the split between ROI and non-ROI parts with minimal error.

The bitrate control module described in Chapter 4 performs only frame level bit-allocation. There is no macroblock level bit-allocation. The task of the conventional bitrate control module is to reduce the error between allocated frame bits B_{alloc} and the frame level bit-consumption without worrying about the bit-distribution within a frame.

In ROI-based bit-allocation, the bit-allocation is not limited to frame level. The input frame is divided into two regions based on the ROI information and target bits are specified for each of these regions separately. The task of the bitrate control module is to meet bitrate for the different regions independently (ROI and non-ROI) with minimal error. This section describes the modifications to the bitrate control module to accomplish this task.

The bitrate control module studied in Chapter 4 computes the deviation factor ($D_m^{n'}$) based on the accumulated error in frame level bit-consumption after encoding every frame as given by eq. (4.2). In steady-state, the deviation factor ($D_m^{n'}$) takes a value such that the error in frame level bit-consumption is minimum.

The following steps summarize the approach to implement region-based bitrate control using the existing bitrate control described in Chapter 4.

- Compute the deviation factor for ROI ($Droi_m^{n'}$) and non-ROI ($Dnroi_m^{n'}$) independently. The deviation factors are updated using the feedback data from ROI and non-ROI parts independently. This can be visualized as having two instances of rate control running for ROI and non-ROI treating ROI and non-ROI as separate frames.
- A single instance of macroblock level bitrate error is maintained (eq. (4.2)). This ties the two instances of bitrate control together, so that frame level bit-error is minimized. The error in allocation and consumption of bits in ROI and non-ROI is compensated throughout the frame resulting in less overall frame level error.
- The VBV compliance works with frame level bit-allocation and therefore remains unchanged. There will be no increase in the number of dropped frames as long as the frame level deviation in bit-consumption is kept under check.

7.3 Experimental Results

This section describes the experimental results of the ROI-based encoding approaches discussed previously.

7.3.1 ROI QP Offset - Results

The result of improved QP offset based ROI encoding is shown in Figures 7.8 to 7.10. The improvements include consideration of relative ROI area to compute the QP offset and bi-directional QP offset discussed in the previous sections. For the sample input *Chet*,

Methodology	Content	PSNR (dB)		
		Frame	**ROI**	**non-ROI**
Conventional Encoding	*Paul*, 250 kbps	39.15	37.72	39.22
	Johnny, 350 kbps	37.93	36.08	38.13
	Chet, 250 kbps	31.35	32.91	31.24
Improved QP Offset based ROI encoding	*Paul*, 250 kbps	38.29	40.10	38.24
	Johnny, 350 kbps	37.50	37.37	37.52
	Chet, 250 kbps	31.08	34.89	30.89
ROI-based Bit-allocation	*Paul*, 250 kbps	38.22	40.95	38.15
	Johnny, 350 kbps	36.94	38.57	36.85
	Chet, 250 kbps	30.55	36.17	30.30

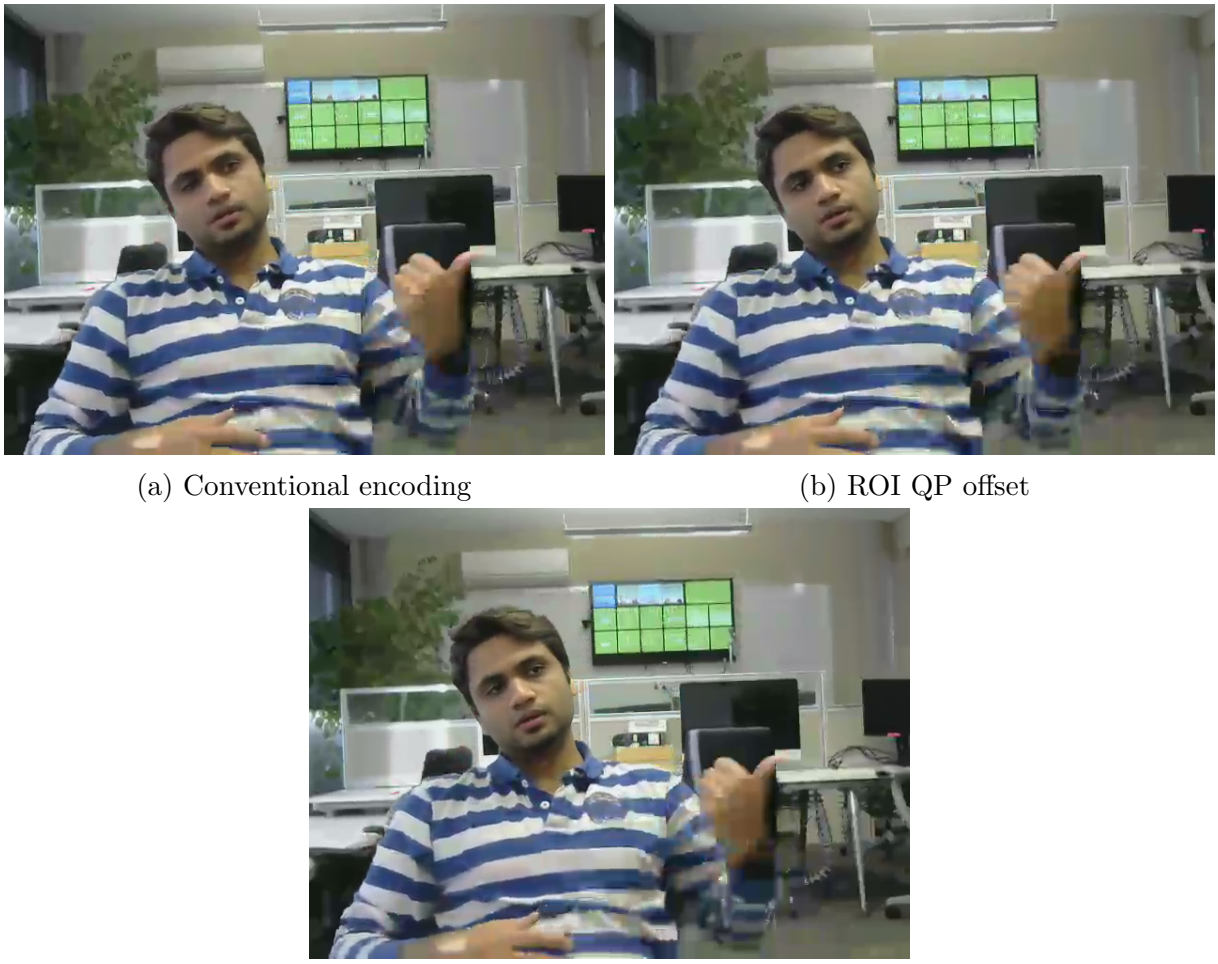
Table 7.4: PSNR comparison between conventional encoding and different approaches of ROI-based encoding.

there is a clear improvement in the sharpness of the face region with improved QP offset based ROI encoding (Figure 7.8b) over conventional encoding (Figure 7.8a). The minor drop in the PSNR of non-ROI is masked by the improvements in the face region which improves the overall perceptual quality. Similar trend is observed for other sample input sequences as well.

PSNR Value: The PSNR values for all the sample video sequences with ROI-based encoding using QP offset is tabulated in Table 7.4. As expected, there is a huge improvement in the PSNR of ROI and a corresponding drop in non-ROI PSNR. The average improvement in PSNR of ROI is 1.88 dB with an average drop in non-ROI PSNR of 0.646 dB across all the chosen sample contents. The magnitude of the drop in non-ROI PSNR is minor compared to the improvement in ROI PSNR. The reason for this observation is that in all the three chosen sample input sequences, the area of ROI is small compared to the area of non-ROI ($A_{roi} < 0.5$). Due to low A_{roi}, the bits saved from quality degradation of non-ROI is used to encode a smaller number of ROI macroblocks with a higher quality leading to asymmetric change in PSNR values.

PSNR and Quantization Maps: The comparison of PSNR and quantization maps between conventional encoding and improved QP offset based ROI encoding is shown in Figure 7.11 and Figure 7.12 respectively. In a simpler approach with uni-directional QP offset, one of the disadvantages discussed is the non-uniform increase in ROI QP. The quantization map shown in Figure 7.12b has more uniform increase in QP of non-ROI compared to the approach without bi-direction QP offset shown in Figure 7.1f. The regions below ROI do not reach extreme high QP since a positive QP offset is added to non-ROI QP from the start of the frame. This also improves the overall perceptual quality.

Delay Plots: The delay plots for improved QP offset approach is shown in Figure 7.13.



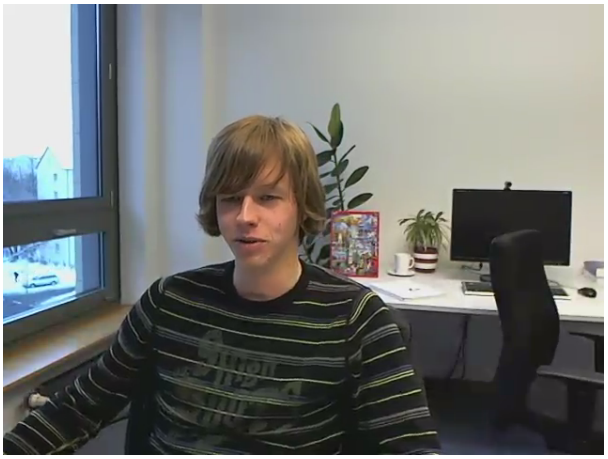
(a) Conventional encoding (b) ROI QP offset

(c) ROI-based bit-allocation

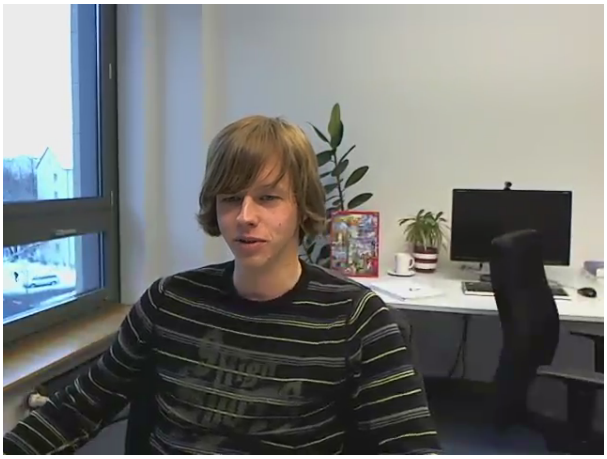
Figure 7.8: Comparison of *Chet* encoded at 250 Kbps with conventional encoding and different ROI-based encoding approaches. The increase in the sharpness of the face region is clearly noticeable in ROI-based encoding approaches.

For each of the sample video sequences, the delay curve follows very closely the curve for conventional encoding. This shows that the QP offset based approach does not modify the bit-consumption behavior at the frame level. The absence of any additional zero-points indicates that there is no increase in the number of dropped frames. This is a highly desirable behavior since it indicates that there will be no glitch in the playback of the video due to ROI-based encoding.

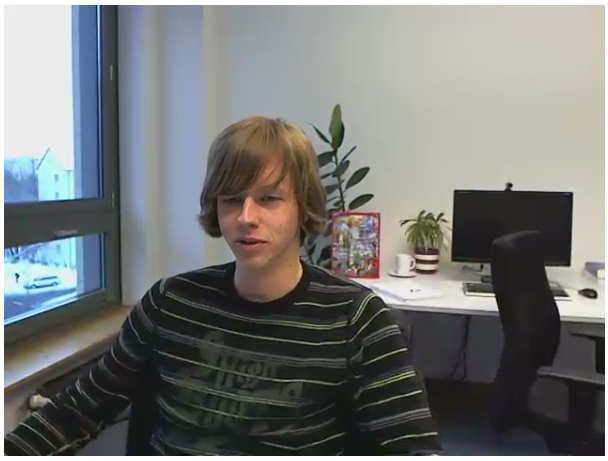
The advantages and disadvantages of this approach can be summarized as follows:



(a) Conventional encoding



(b) ROI QP offset



(c) ROI-based bit-allocation

Figure 7.9: Comparison of *Paul* encoded at 250 Kbps with conventional encoding and different ROI-based encoding approaches. The increase in the sharpness of the face region is noticeable in ROI-based encoding approaches.

Advantages

- Easy to implement with minimal changes required to the encoder.
- As shown in the delay plots of Figure 7.13, there is hardly any change in the frame level behavior of the bitrate control module due to ROI-based encoding.
- QP offset is forced irrespective of the buffer conditions, therefore guarantees quality difference between ROI and non-ROI.

Disadvantages

- The QP offsets chosen does not take into account any of the input content characteristics. Same QP offset is used for all the contents for a given ratio of ROI to non-ROI area.
- Since buffer conditions are not considered at the macroblock level, there is an increased possibility of dropped frames.
- The deviation control mechanism described in Section 4.2.2 is not in sync with the QP offsets. The deviation control mechanism will increase the QP even for ROI blocks when there is an increased deviation due to over-consumption of bits by ROI macroblocks.

7.3.2 ROI-based Bit-allocation - Results

The ROI-based bit-allocation scheme for ROI encoding considers the input content characteristics to perform bit-allocation for ROI blocks. Therefore, this approach is applicable to generic contents and hence can be expected to be more robust across multiple contents. The result of ROI encoding using ROI-based bit-allocation is shown in Figures 7.8 to 7.10. For the sample input *Chet*, there is a clear improvement in the sharpness of the face region with ROI-based bit-allocation (Figure 7.8c) over conventional encoding (Figure 7.8a) and even ROI encoding using QP offsets (Figure 7.8b). The degradation of non-ROI region is not noticeable which improves the overall perceptual quality by a large extent. Similar trend is observed for other sample input sequences as well.

The overall difference in the image quality is more noticeable in the decreasing order for the sample input sequences *Chet*, *Johnny* and *Paul*. This observation can be explained by considering the overall PSNR values. The PSNR values increases in the same order with *Chet* and *Paul* having minimum and maximum PSNR values respectively. The reason for this difference in PSNR between *Chet* and *Paul* is the difference in temporal complexity. Even though the resolution and operating bitrates are the same for both, *Chet* has very high temporal complexity resulting in very low PSNR value. This observation shows that the gain in perceptual quality is more significant at lower overall PSNR.

PSNR Value: The PSNR values for all the sample video sequences with ROI-based bit-allocation is tabulated in Table 7.4. Similar to observations made with QP offset based ROI encoding, there is a huge improvement in PSNR of the ROI and a small drop in non-ROI PSNR. It can be noticed that PSNR of ROI is higher with ROI-based bit-allocation compared to QP offsets based ROI-encoding. The average improvement in PSNR of ROI is 2.9 dB with an average drop in non-ROI PSNR of 0.9 dB across all the chosen sample contents. This shows that ROI-based bit-allocation scheme results in more aggressive movement of bits from non-ROI to ROI parts. This is possible since the input content characteristics are considered in the algorithm which

allows it to exploit the static background to allocate maximum amount of bits to complex ROI parts. This is not possible in ROI encoding with QP offset because the approach needs to be conservative due to the lack of information about the content characteristics. The adverse effects of using large magnitude QP offset is discussed previously in Section 7.1.2.

PSNR and Quantization Maps: The comparison of PSNR and Quantization maps with conventional encoding and improved QP offset based ROI-encoding is shown in Figure 7.11 and Figure 7.12 respectively. The usage of a lower quantization parameter in the face regions compared to conventional encoding is clearly visible. The quantizations maps corresponding to ROI-based bit-allocation shows more prominent distinction between ROI and non-ROI parts compared to QP offset based approach. This is achieved by allocating bits to ROI and non-ROI parts independently. Unlike QP offset ROI encoding, the deviation control mechanism does not conflict with the higher bits allocated to ROI. This leads to clear distinction between ROI and non-ROI QP. The increase in non-ROI QP is also more uniform with this scheme as shown in Figures 7.12c, 7.12f and 7.12i as compared to QP offset based approach shown in Figures 7.12b, 7.12e and 7.12h.

The PSNR maps shown in Figure 7.11 depicts better quality in the face regions compared to conventional and QP offset based ROI encoding. This is due to the aggressive movement of bits from non-ROI parts to ROI parts.

Delay Plots: The delay plots for ROI-based bit-allocation are shown in Figure 7.13. For each of the sample video sequences, the delay curve follows the curve for conventional encoding. However, this is one aspect in which ROI-based bit-allocation scheme performs inferior compared to QP offset based ROI encoding. There is a considerable deviation in the delay curve for *Chet* and *Paul* with ROI-based bit-allocation (figs. 7.13b and 7.13d) as compared to QP offset based ROI encoding (figs. 7.13a and 7.13c). There is also an increase in the number of dropped frames which can be seen as an additional zero-point in Figure 7.13d. However, the delay curves for sample content *Johnny* for both approaches of ROI-based encoding shown in Figures 7.13e and 7.13f have very low deviation from delay curve for conventional encoding. This is due to the comparatively low temporal complexity.

The reason for increased deviation in delay curve and an increase in the number of dropped frame is the usage of region-based bitrate control. Since bits are allocated independently to ROI and non-ROI, the bitrate control attempts to meet these region-specific target bitrate leading to deviation in bitrate in two regions. This increased deviation leads to overall increase in frame level bit-allocation and bit-consumption error. In extreme cases, this increased error leads to dropped frames.

The ROI-based bit-allocation scheme offers an efficient way of improving perceptual quality in video conferencing scenarios. This approach performs better in most aspects compared to QP offsets based ROI encoding. The advantages and disadvantages of this approach

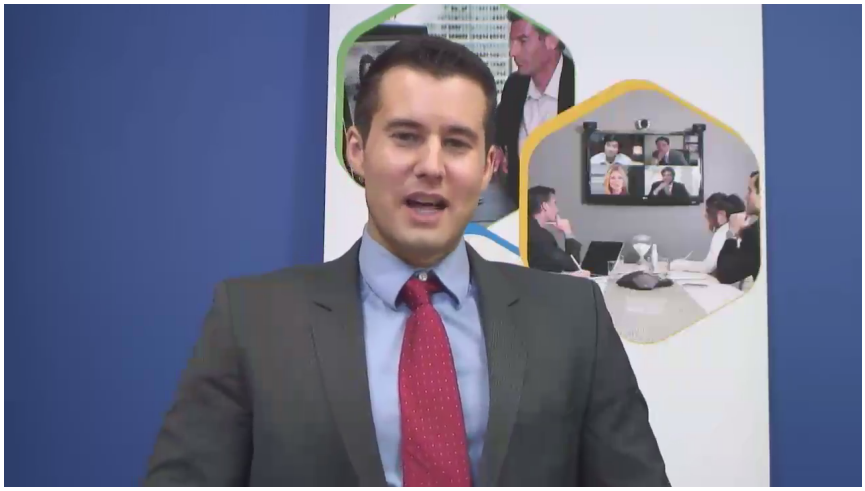
compared to QP offset based ROI encoding can be summarized as follows.

Advantages

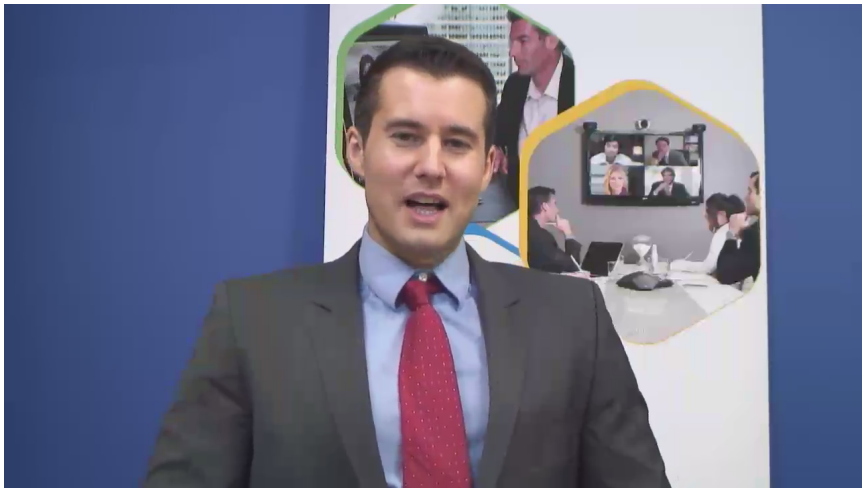
- The consideration of input content characteristics for ROI-based encoding makes this approach work well for all types of contents.
- This approach can guarantee minimum quality for the background region by having sufficient bits allocated to the non-ROI to avoid extreme blockiness.
- The variable importance of ROI can be factored easily by altering the bias factor (k) to vary the magnitude of bit movement from non-ROI to ROI.

Disadvantages

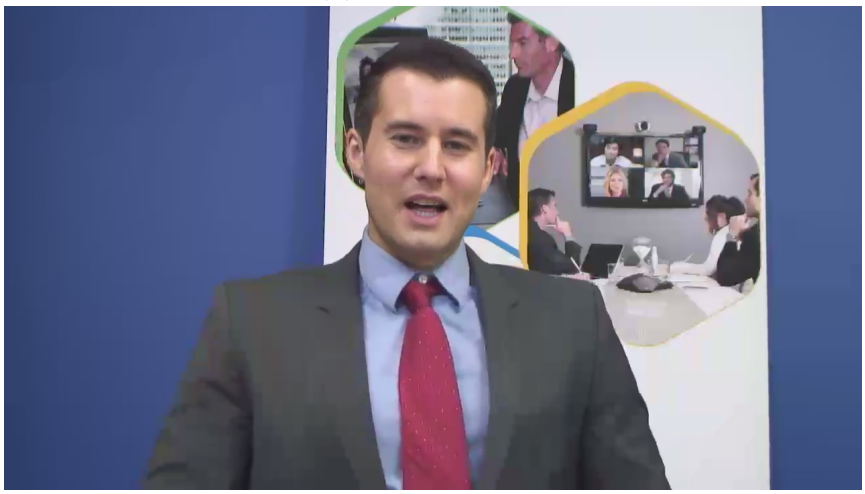
- The delay plots in Figure 7.13 show that there is a higher deviation in bit-consumption behavior with ROI-based bit-allocation scheme compared to ROI-encoding using QP offsets. This might lead to an increased number of dropped frames in extreme conditions.
- This approach requires major modifications to the bit-allocation module and hence more complex to implement.
- The determination of relative complexity factor (C_R) needs trained models to predict bit-consumption of ROI and non-ROI parts during conventional encoding without using ROI information. This prediction model has bad accuracy for macroblocks encoded with inter-prediction.
- The cost-bits model developed to estimate the relative bit-consumption factor (B_R) depends on the approach used to compute RDO cost. This varies across different video encoders making it necessary to develop different models for different encoders.



(a) Conventional encoding



(b) ROI QP offset



(c) ROI-based bit-allocation

Figure 7.10: Comparison of *Johnny* encoded at 350 Kbps with conventional encoding and different ROI-based encoding approaches. The increase in the sharpness of the face region is noticeable in ROI-based encoding approaches.

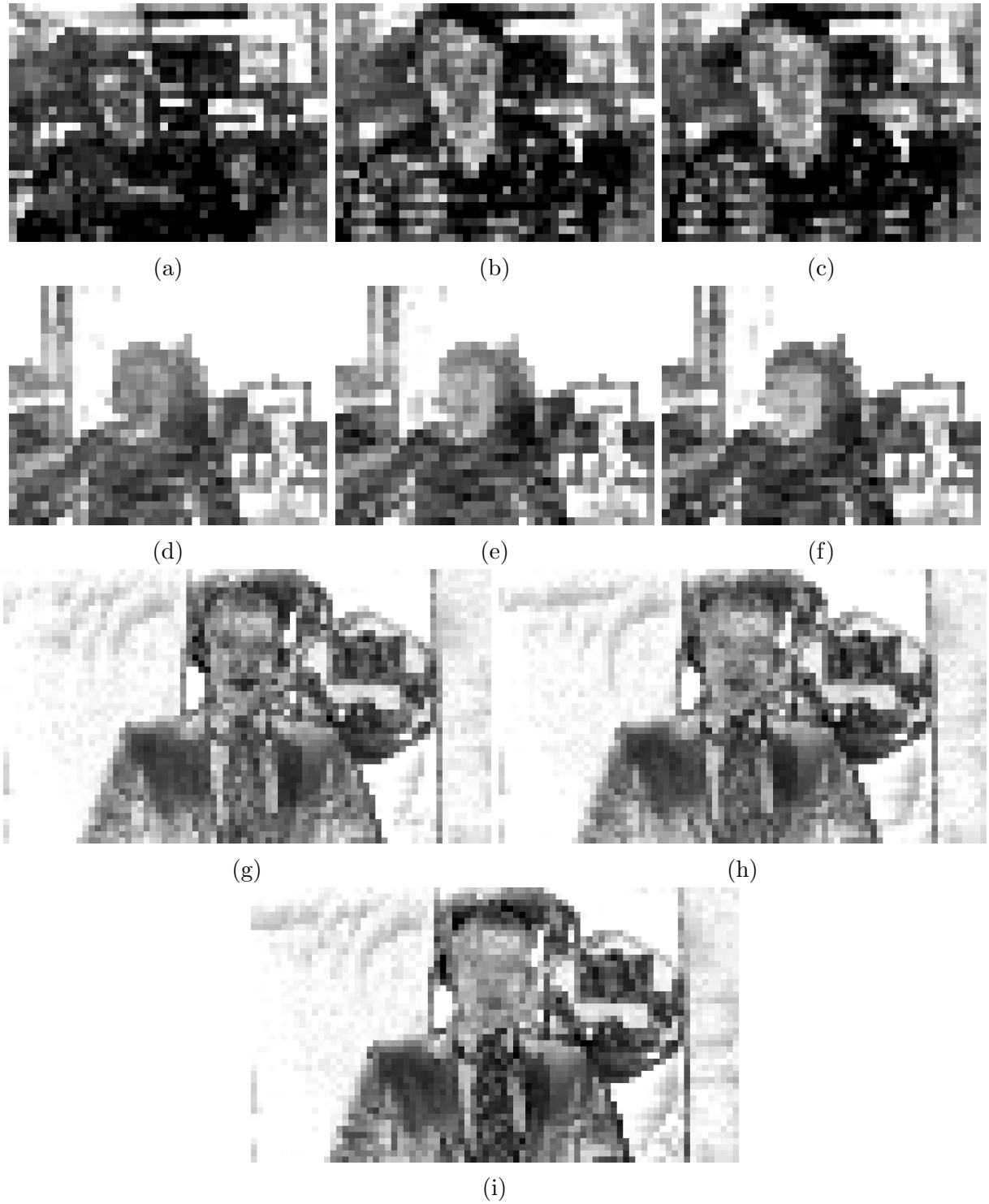


Figure 7.11: Comparison of PSNR maps (PSNR range of 25 dB - 45 dB) of sample contents (a, b, c) *Chet*, (d, e, f) *Paul* and (g, h, i) *Johnny* for conventional and ROI-based encoding approaches. (a,d,g) Conventional encoding, (b,e,h) ROI QP Offset, (c,f,i) ROI-based bit-allocation.

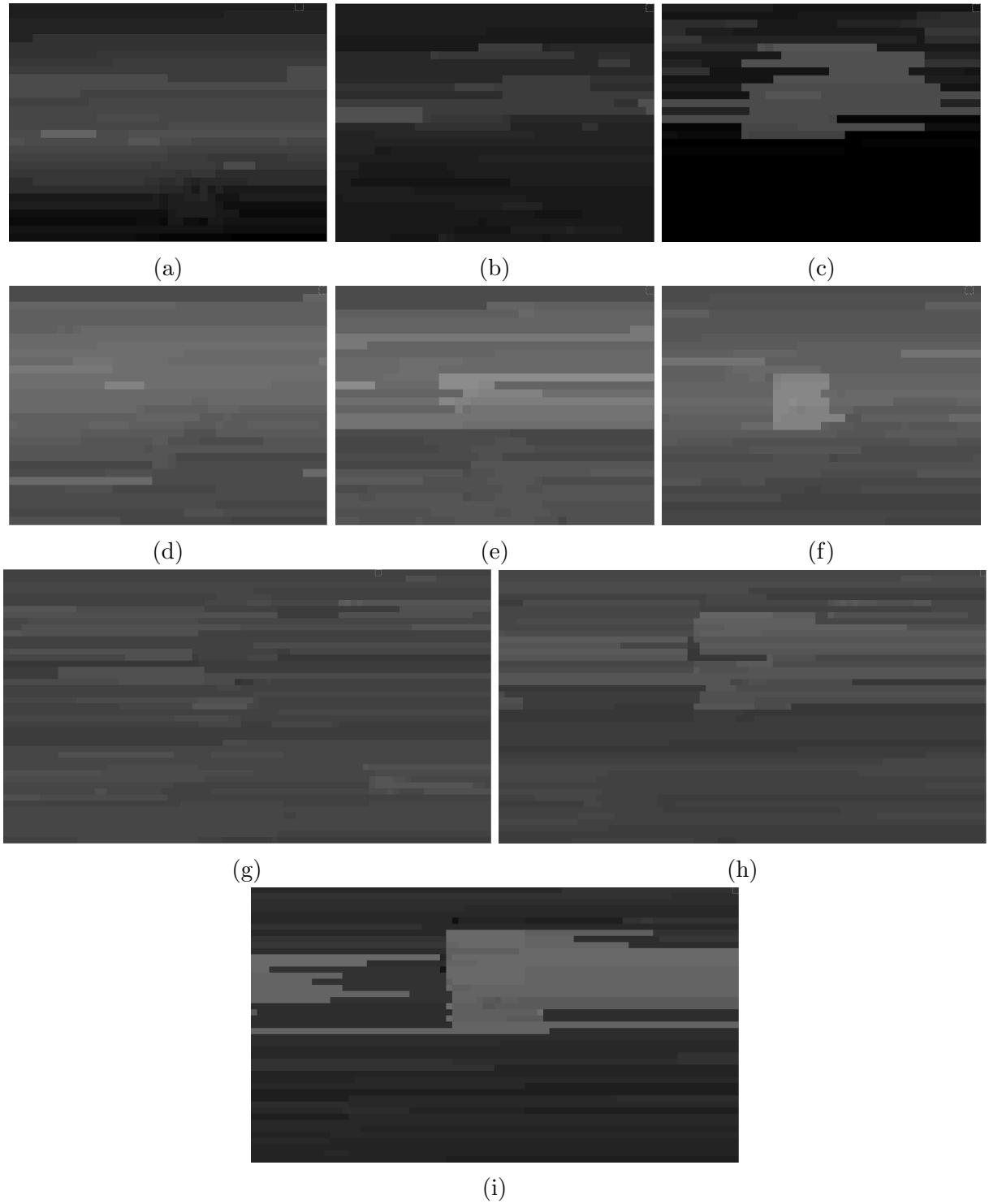
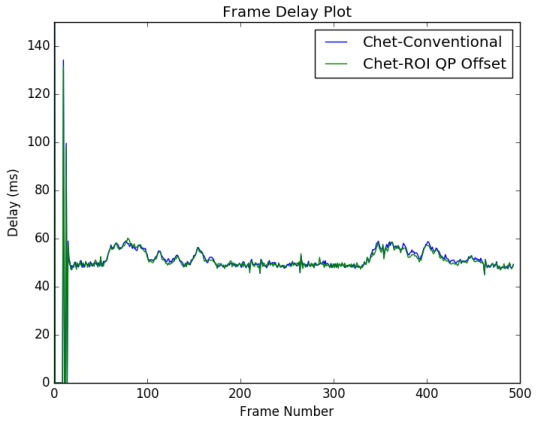
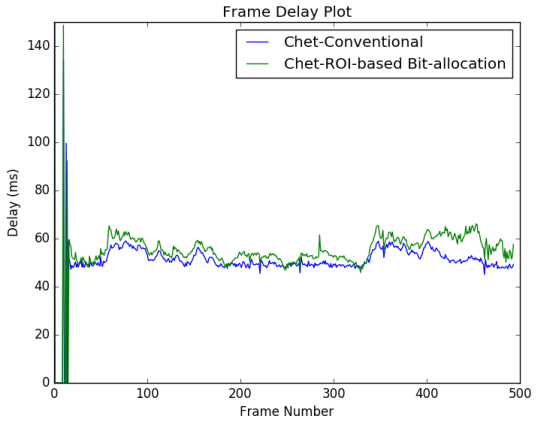


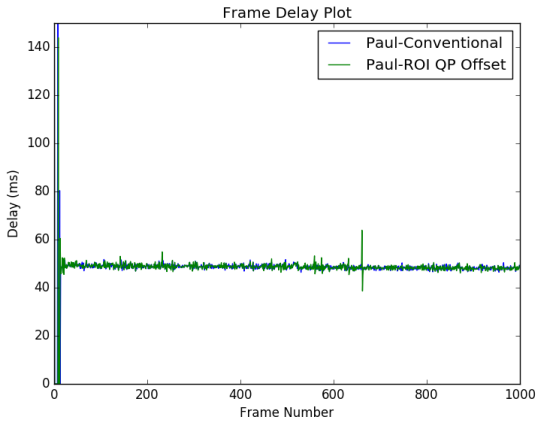
Figure 7.12: Comparison of quantization maps (QP range of 1 - 51) of sample contents (a, b, c) *Chet*, (d, e, f) *Paul* and (g, h, i) *Johnny* for conventional and ROI-based encoding approaches. (a,d,g) Conventional encoding, (b,e,h) ROI QP Offset, (c,f,i) ROI-based bit-allocation.



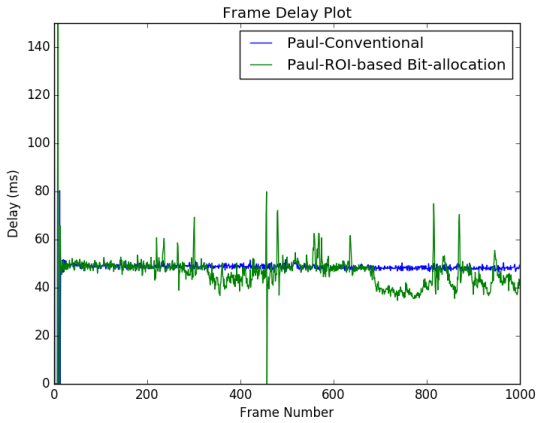
(a) *Chet* - ROI QP Offset



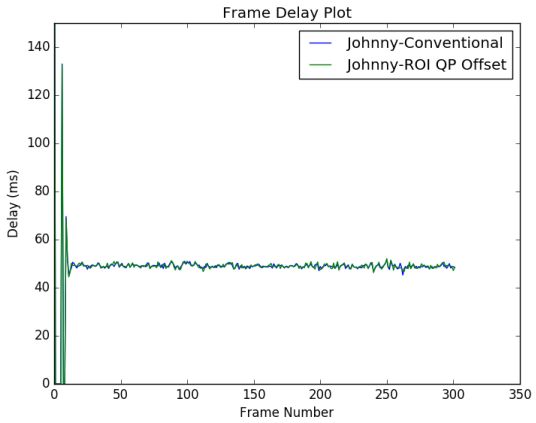
(b) *Chet* - ROI-based bit-allocation



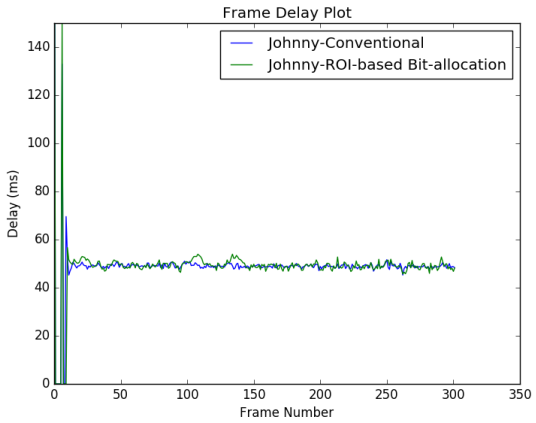
(c) *Paul* - ROI QP Offset



(d) *Paul* - ROI-based bit-allocation



(e) *Johnny* - ROI QP Offset



(f) *Johnny* - ROI-based bit-allocation

Figure 7.13: Comparison of Delay plots for the sample sequences for different approaches of ROI-based encoding (Green) and conventional encoding (Purple). The plots show that ROI-based encoding does not alter the delay behavior with respect to conventional encoding in most cases.

Chapter 8

Conclusion

The increasing popularity of video telephony over mobile platforms like smartphones has increased the demand for efficient low bitrate solutions. The availability of high-speed mobile Internet has not kept its pace with the demand in many parts of the world. Therefore, it is important to develop efficient low bitrate solutions that enable high-quality video telephony over slow cellular networks.

This thesis contributes to the research area by proposing an ROI-based bitrate control for low-delay encoding to increase the perceptual quality during video conferencing by preferentially encoding the face regions (ROI) with higher quality compared to the rest of the frame. As a first step, an existing low-delay bitrate control module's performance is evaluated at low bitrate using multiple sample video conferencing sequences. The H.264 encoder is used for all the evaluations in this work. This work proposes two different approaches of ROI-based encoding for achieving better perceptual quality at low bitrate. These approaches offer a trade-off between implementation complexity and output quality.

Firstly, a simple approach of using negative QP offset for the macroblocks belonging to ROI is studied. This approach requires very minimal changes to the existing low-delay bitrate control module. Further improvements like tuning QP offsets, using area of ROI for QP offset computation and usage of bi-direction QP offsets for both ROI and non-ROI are described. This approach of QP offset based ROI encoding offers considerable improvement in the overall perceptual quality compared to conventional encoding. This approach is used to analyze the effect of increasing the magnitude of the quality difference between ROI and non-ROI on the perceptual quality. It is established through experiments with different magnitude of QP offsets that improving quality in ROI at the cost of degrading quality in non-ROI does not always guarantee improvement in the perceptual quality. The usage of a very high magnitude QP offset leading to an unbounded movement of bits from ROI to non-ROI can have an adverse effect on the perceptual quality.

The second approach discussed in this work includes ROI-based bit-allocation to allocate a higher proportion of bits to ROI to improve the perceptual quality. In this approach,

explicit region based bit-allocation is performed where a target bitrate is assigned to ROI and non-ROI parts independently. In order to make this approach more robust across different classes of input video contents, the characteristics of input frame complexity are considered to compute the bits allocated for ROI to achieve an improved perceptual quality. Therefore, the magnitude of bit movement from non-ROI to ROI parts is optimal for the specific encoded content instead of using the same arbitrary weights for all contents. This is accomplished by developing a cost-bits model to predict bit-consumption using the complexity of the region (RDO cost). Finally, the output of conventional encoding is compared against the two proposed approaches for ROI-based encoding. The evaluation includes the PSNR values, PSNR and QP distribution within a frame. The gain in PSNR of ROI region is considerably higher compared to the corresponding drop in PSNR of non-ROI in both the approaches. In QP offset based ROI encoding, the average improvement in PSNR of ROI is 1.88 dB with an average drop in non-ROI PSNR of 0.646 dB. A more aggressive bit-movement is observed with ROI-based bit-allocation. In this approach, the average improvement in PSNR of ROI is 2.9 dB with an average drop in non-ROI PSNR of 0.9 dB. The delay curve for each of the encoded sequences is analyzed to make sure that ROI-based encoding does not alter the overall behavior of the encoder. It is found that the ROI-based bit-allocation offers superior perceptual quality output with well-defined movement of bits from non-ROI to ROI compared to ROI-based encoding using QP offset. However, both the proposed approaches offer better perceptual quality compared to the conventional encoding with marginal drop in overall PSNR.

The OpenCV implementation of face detection based on Viola-Jones AdaBoost algorithm is used to detect the face. An optimization technique using motion vector-based variable interval face detection is proposed for real-time face detection. Experimental results show a reduction in average error and peak error by 25% and 56.3% respectively compared to the approach of detecting the face in every 15th frame. The computation requirements remained almost similar in both the compared approaches. This enables real-time face detection in a video stream with marginal inaccuracies.

8.1 Future Work

There is a lot of scope to improve the perceptual quality at low bitrate by removing the perceptual redundancies. This work proposes two approaches to achieve this in low-delay encoding. The results obtained in this work can be improved further by developing more accurate cost-bits model which considers the changes in QP within a frame due to varying complexities across the macroblocks.

This work is based on the fundamental assumption that improving the quality of ROI at the cost of non-ROI improves the overall perceptual quality. In this work, the magnitude of the quality difference between ROI and non-ROI is decided based on the relative complexities of these regions. Therefore, this work offers a way to avoid the usage of arbitrary weights for

ROI used in most of the previous works. However, the value of ROI bias factor chosen in this work is heuristic. There is no well-defined formulation to determine the absolute magnitude of the quality difference between ROI and non-ROI to achieve the best perceptual quality. This is an open research area which needs investigation in the fields like subjective quality assessment and determination of ROI-based subjective quality metric.

List of Figures

List of Tables

Appendix A

List of Abbreviations

AVC Advanced Video Coding

CAMSHIFT Continuous Adaptive Mean Shift

CBR Constant Bitrate

HEVC High Efficiency Video Coding

HVS Human Visual System

MB Macroblock

ME Motion Estimation

PSNR Peak Signal to Noise Ratio

QP Quantization Parameter

RDO Rate Distortion Optimization

ROI Region Of Interest

SAD Sum of Absolute Difference

SATD Sum of Absolute Difference in Transform Domain

SSIM Structural Similarity Index

VBR Variable Bitrate

VBV Video Buffer Verifier

Bibliography

[AC05] T. Ebrahimi. A. Cavallaro, O. Steiger. Semantic video analysis for adaptive content delivery and automatic description. *IEEE Trans. Circuits Syst. Video Technol.*, 15(10):1200–1209, 2005.

[Bra98] G.R. Bradski. Real time face and object tracking as a component of a perceptual user interface. *Applications of Computer Vision, 1998. WACV '98. Proceedings., Fourth IEEE Workshop on*, 17(1):214–219, 1998.

[DGH13] A. Mulayoff D. Grois, D. Marpe and O. Hadar. Performance comparison of H.265/MPEG-HEVC, VP9, and H.264/MPEG-AVC encoders. *Picture Coding Symposium (PCS), 2013*, pages 394–397, 2013.

[FL16] Na Li. Fan Li. Region-of-interest based rate control algorithm for H.264/AVC video coding. *Multimed Tools Appl*, 75:4163–4186, 2016.

[GLW12] S.-Y. Chien. G.-L. Wu, Y.-J. Fu. Region-based perceptual quality regulable bit allocation and rate control for video coding applications. *VCIP*, 2012.

[HM11] Jorn Ostermann. Holger Meuel, Marco Munderloh. Low Bit Rate ROI Based Video Coding for HDTV Aerial Surveillance Video Sequences. *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition - Workshops (CVPRW)*, pages 13–20, 2011.

[HM16] Jorn Ostermann. Holger Meuel, Florian Kluger. Codec independent region of interest video coding using a joint pre- and postprocessing framework. *IEEE International Conference on Multimedia and Expo (ICME)*, 2016.

[JOGJSW12] Thiow Keng Tan J. Ohm. G. J. Sullivan, H. Schwarz and T. Wiegand. Comparison of the Coding Efficiency of Video Coding Standards-Including High Efficiency Video Coding (HEVC). *IEEE Transactions on Circuits and Systems for Video Technology*, 22(12), 2012.

[Joh] `https://media.xiph.org/video/derf/`.

[Lea] `http://www.eenadupratibha.net/pratibha/engineering/content_three_tra_layer_u6.html`.

[LT05] K.R. Rao. Lin Tong. Region of interest based H.263 compatible codec and its rate control for low bit rate video conferencing. *Intelligent Signal Processing and Communication Systems*, 2005.

[MB13] Manzur Murshed and James Brown. High Quality Region-of-Interest Coding for Video Conferencing based Remote General Practitioner Training. *The Fifth International Conference on eHealth, Telemedicine, and Social Medicine*, 2013.

[MX14] Shengxi Li. Mai Xu, Xin Deng. Region-of-Interest Based Conversational HEVC Coding with Hierarchical Perception Model of Face. *IEEE Journal of Selected Topics in Signal Processing*, 8(3), 2014.

[SL03] A. C. Bovik. S. Lee. Fast algorithms for foveated video processing. *IEEE Trans. Circuits Syst. Video Technol.*, 13(2):149–161, 2003.

[SML02] Yan Lu Siwei Ma, Wen Gao and Hanqing Lu. Proposed draft description of rate control on JVT standard. Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG (ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6) 6h Meeting, 2002.

[TWL03] Gary J. Sullivan Thomas Wiegand and Ajay Luthra. Overview of the H.264/AVC Video Coding Standard. *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY*, 13(7), 2003.

[VJ01] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features, 2001.

[Wan95] B. Wandell. Foundations of Vision. Sinauer, 1995.

[YLS08a] Z. G. Li Y. Liu and Y. C. Soh. A novel rate control scheme for low delay video communication of H.264/AVC standard. *IEEE Trans. Circuits Syst. Video Technol*, 17(1):68–78, 2008.

[YLS08b] Z. G. Li Y. Liu and Y. C. Soh. Region-of-Interest Based Resource Allocation for Conversational Video Communication of H.264/AVC. *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY*, 18(1), 2008.

[ZWW11] Kexin Zhang Zongze Wu, Shengli Xie1 and Rong Wu. Rate Control in Video Coding. `http://www.intechopen.com/books/recent-advances-onvideo-coding/rate-control-in-video-coding`, 2011.