USC-SIPI REPORT #429

Techniques for Compressed Visual Data Quality Assessment and Advanced Video Coding

by

Sudeng Hu

May 2016

Signal and Image Processing Institute
UNIVERSITY OF SOUTHERN CALIFORNIA
Viterbi School of Engineering
Department of Electrical Engineering-Systems
3740 McClintock Avenue, Suite 400
Los Angeles, CA 90089-2564 U.S.A.

TECHNIQUES FOR COMPRESSED VISUAL DATA QUALITY ASSESSMENT AND ADVANCED VIDEO CODING

by

Sudeng Hu

A Dissertation Presented to the
FACULTY OF THE USC GRADUATE SCHOOL
UNIVERSITY OF SOUTHERN CALIFORNIA
In Partial Fulfillment of the
Requirements for the Degree
DOCTOR OF PHILOSOPHY
(MING HSIEH DEPARTMENT OF ELECTRICAL ENGINEERING)

May 2016

Contents

List of Tables

List of Figures

Abstract

Object quality assessment for compressed images and videos is critical to various image and video compression systems that are essential in the delivery and storage. Although the Mean Squared Error (MSE) is computationally simple, it may not be accurate to reflect the perceptual quality of compressed signals, which is also affected dramatically by the characteristics of Human Visual System (HVS) such as masking effect. In this thesis, first, video quality metrics are developed based on machine learning approaches. Due to the complicated relationship among a large number of factors, machine learning is used to build a proper model for various features including the distortion features and video content features. Second, an image quality metric (IQM) and a video quality metric (VQM) are proposed based on perceptually weighted distortion in term of the MSE. To capture the characteristics of HVS, for images, a spatial randomness map is proposed to measure the masking effect and a preprocessing scheme is proposed to simulate the processing that occurs in the initial part of human HVS. For the VQM, the dynamic linear system is employed to model the video signal and is used to capture the temporal randomness of the videos. The visual attention is included in the proposed VQM as well, since only a limited parts of details are perceived with high sensitivity while the other parts are significantly blurred in the HVS. The performance of the proposed IQM and VQM are validated on various image and video databases with

various compression distortions. The experimental results show that the proposed IQM and VQM outperforms other benchmark quality metrics.

In addition to the quality assessment, video compression is also important in the system of video delivery and storage, especially different kinds of video content emerging in recent industries such as screen content and 3-D videos. These video formats have very different characteristics from the traditional videos. In this thesis, first, we propose a coding method that is able to code the content with sharp edges efficiently. Such method is highly valuable for the screen content coding and depth map coding of 3-D video. Second, a RD optimized bit allocation scheme is proposed for 3-D videos. In 3-D videos, there are multiple views and each view contain two types of video, i.e., texture map and depth map. The proposed bit allocation method could properly allocate bits among different views as well as between different maps. The experimental results also verify that proposed bit allocation outperform the benchmark algorithms in terms of RD efficiency.

Acknowledgments

I would like to thank all the people who have helped and inspired me during my study at the University of Southern California.

I would like to thank my advisor Prof. C.-C. Jay Kuo, for his guidance and suggestions in my research. He has always been a constant source of encouragement, and his enthusiasm in research had motivated all his advisees, including me.

Thanks also go to all the MCL group members at USC. I would like to thank them for their friendship and help in the past more than four years.

Finally, I especially would like to thank my wife, Zhiguan Wang and my parents. They are always tolerant, supportive and encouraging. Without them, I would not have finished this thesis.

Chapter 1

Introduction

1.1 Significance of Research

Due to the rapid development of various digital video and image application system, such as video conference, IPTV, image and video quality assessment becomes increasingly important as it can either evaluate the performance of these system or send feedback to them for the performance optimization. Image and video quality should be evaluated in subjective terms, since human satisfaction is the ultimate criteria to determine video quality. The subjective measurement such as the Mean Opinion Score (MOS) is often used as the ground truth. On the other hand, subjective evaluation is time-consuming and costly. It sometimes demands special facilities. Moreover, it is not suitable for real-time video quality monitoring. Hence, it is desirable to develop an objective quality assessment method that can automatically assess image and video quality without involving human in the loop.

Due to the inconvenience of subjective image and video quality assessment, a large number of objective image quality metrics (IQM) and video quality metrics (VQM) have been developed. Most of the developed IQMs and VQMs are aimed at handling a large range of distortion types and usually tested in the databases such as the LIVE database [45]. However developing an universal quality metric is quite challenge. Due to the wide application of image and video compression in delivery and storage, the compression distortion is one of major distortion among

various distortion types. Besides, IQM and VQM play a key role in image and video coding in the processes such as Rate-Distortion Optimization (RDO) [56, 50, 121]. Therefore, it is highly desired to have accurate IQMs and VQMs for image and video compression. Moreover compression distortion is quite different from other distortion types such as white noise or transmission error distortion, it has its unique characteristics. For example, the distortion is content dependent and it is usually larger in complex content than in smooth content within the same images or the same frames of video sequences. However its characteristics haven't been fully investigated and utilized to design proper quality metrics.

The peak-signal-to-noise-ratio (PSNR) and the meansquared- errors (MSE) indices are often used as quality indices in the coding community. Although there has been criticism on their suitability for ignoring the human perception factor, they do offer two attractive features: 1) computational simplicity and 2) fine granular scores. The latter is especially important since the quality of videos coded by different encoders could be quite close to each other. A coarse-scale mean opinioin score (MOS) system obtained from the traditional subject test may not be sufficient to differentiate their sutble difference. Instead, we may demand the pairwise comparison by a few gold eyes. Furthermore, PSNR and MSE still work well for some distortion such as the quantization noise. Therefore in this thesis, by analyzing the properties of the HVS and compression distortion, we modify MSE to develop proper metrics specifically for the compression distortion.

Besides the quality assessment, video compression still play an important role in the system of video storage and delivery. With the emerge of new type of video materials, such as screen content video, 3D video, the traditional video codecs are no longer able to compress these video formats effeciently. Therefore it is highly

desired to develop new coding tools that fit the characterists of these video formats and optimize the coding efficiency.

1.2 Background of Human Visual System

1.2.1 Contrast Sensitivity

The initial visual signal processing in the HVS includes two steps. In the first step, the visual signal goes through eye's optics, forming an image on the retina. Because of the diffraction and other imperfections in the eye, such processing would blur the passed image. In the second step, the image will be filtered by neural filter as it is received by photoreceptor cells on retina and then passed on to lateral geniculate nucleus (LGN) and the primary visual cortex. These processes are more like low-pass filtering and will hide parts of signal from perception. This effect in the HVS can be described as contrast sensitivity function.

Human contrast sensitivity has been explored for vision models by many vision scientists in various studies. For examples, Barten [13] cataloged measured luminance data from many different studies and derived an analytical expression for modeling CSFs. The CSF data used in his work were measured mostly with horizontally or vertically oriented sinusoidal patterns but the effect of orientation was ignored. In addition, his CSF model was restricted to photopic luminance conditions (day light vision). Daly [33] used an observer model that incorporates a CSF and detection mechanisms for studying the visual equivalence of two images. The CSF was modeled based on a Barten CSF with consideration of other effects including orientation in degrees, lens accommodation due to observation distance, and essentricity in visual degrees. Peli [11] used measured CSFs of individuals in simulating the appearance of natural images from different observation distances.

His CSF data were obtained with 1-octave Gabor patches and a detection task, where the mean luminance of all images was about 40 . To threshold luminance images by the CSF, Peli applied his CSF to the luminance images in a nonlinear fashion using a local mean contrast for each element in the image.

1.2.2 Masking Effect

Masking effect refers to human's reduced ability to detect a stimulus on a spatially or temporally complex background. The traditional way to measure the masking effect is using a divisive gain control method, which decomposes the image into multiple channels and analyzes the masking effect among the channels by divisive gain normalization [66] and [129]. However, the mechanism of gain control mostly remains unknown. Additionally, since only simple masker such as sinusoidal gratings or white noise is used in the experiments to search for optimal parameters to fit the gain control model, there is no guarantee that these models are applicable to natural images [26].

In [128] and [47], it is pointed out that masking effect highly depends on the level of randomness created by the background. Usually the regular background contains predictable content and the stimulus will become distinct from neighborhood when it is different from human's expectation of its position. While in the random background, the content is unpredictable, and thus any change on it will be less noticed. Therefore, there is higher masking in the random background than the regular background. In [128], a concept of entropy masking is proposed to measure masking effect of background using zero order entropy. However, it fails to consider the spatial relation of pixel values. In addition, a single value might not be enough to indicate randomness of the whole background, because the content

in the background may vary significantly. Furthermore, only with masking measurement is insufficient to predict the perceptual distortion, because it is unclear how the proposed masking measurement affects the perceived distortion.

1.3 Background and Related Work on Video Compression

Screen content coding has received much interest from academia and industry in recent years. The High Efficiency Video Coding (HEVC) standar has achieved significant improvement in coding efficiency as compared with the state-of-the-art H.264/AVC standard. However, HEVC has been designed mainly for natural video captured by cameras. Screen content images and video, also known as compound images, hybrid images, and mixed-raster content material, typically contains computer-generated content such as text and graphics, sometimes in combination with natural or camera-captured material.

There has been a lot of research done on the classification of screen and natural content, e.g., [46]-[28]. Since screen content video may contain artificial content generated by computers, it tends to have sharp edges on object boundaries. The strong edges will lead to discontinuities in the residual signal after intra prediction, and these discontinuities will spread the energy over a wide frequency range, thus reducing the efficiency of transform-based coders such as HEVC. To address this issue, a new intra mode called residual scalar quantization was proposed in [65], where the residual signal is directly encoded by an entropy coder without performing the DCT transform. A similar transform skip was proposed in [91], where the 2D transform can be skipped in either one or both directions. A method was proposed in [94] to quantize residual signals adaptively in either the transform

or the spatial domain. These papers report improvements in coding efficiency by skipping the transform for some blocks.

For screen content, it is our observation that directly encoding residual signals in the spatial domain may not be efficient enough. This is because, except for the edge, the remaining areas are still smooth and can be coded more effectively with a transform. In this work, we propose a new scheme, called *Edge Mode* (EM), to encode these kinds of blocks. Based on the intra prediction direction, six possible edge positions inside a block are defined, and one of them will be selected via rate-distortion (RD) optimization. To reduce the encoding complexity, the proposed scheme can be further simplified by classifying intra modes into four categories. Then, $M \times N$ 2D DCT transforms or non-orthogonal 2D transforms are performed separately in sub-blocks. Finally, the new edge mode is integrated into HEVC to result in a more powerful coding scheme.

Three-dimensional video (3DV) has gained increasing interests recently. The typical 3DV is stereo-view video which provides each eye with one video separately at the same time. The small differences between these two videos cause the illusion of depth perception for human. In addition to the stereoscopic 3D video, the emerging autostereoscopic display [116, 62, 51, 15] which emits a number of views enable autostereoscopic 3D video. Comparing with stereoscopic viewing, it involves a more general case of n-view multiview video. In this scenario, the viewpoint can be interactively changed by selecting different stereo pairs of views from n-view.

Delivering or storing n-view video requires tremendous bits that beyonds current transmission or storage capacity. Multiview Video Coding (MVC) [30] is developed to encode the multiview videos, where both the temporal redundancy within each view and inter-view redundancy among the neighbouring views are exploited [89]. Although the MVC encoder performs excellent coding efficiency, it

is still not efficient enough to store or to transmit large numbers of views. Multiview plus depth format (MVD) [93, 90] presents a promising solution for the efficient delivery of 3DV. Only a subset m of n views are coded and transmitted, along with additional supplementary information such as per-pixel depth map which provides scene geometry information. At receiver side, these m coded views provide references for generating the rest views, which are synthesized as the virtual views via Depth-Image-based-Rendering (DIBR) [60, 40]. MVD reduces the number of the views to be transmitted but it can still reconstruct all the required views at the receiver side.

Rate Control (RC) is employed in video coding to regulate the bit rate meanwhile guarantee good video quality. As for 3DV, it becomes more complicated because multiple views are involved in coding and within each view there are two kinds of video sequences (*i.e.* texture and depth map). One of challenge problems is the bit allocation between the texture and depth map. Since the quality of virtual view is affected by the quality of both the texture and depth map, the bits should be allocated to balance their quality. In [35], bits are allocated to minimize the total distortion of the texture and depth map. However since the depth map is not presented for viewing, the minimum total distortion does not guarantee the optimal quality in the virtual views. In [110] the optimal bit allocation between the texture map and depth map is exhaustively searched by a hierarchical search method. In [79], the distortion of the virtual view is modeled and the optimal bit allocation between the texture and depth map is searched based on this distortion model. In [138], a similar distortion model is derived for virtual view and to achieve optimal virtual view quality, bits are allocated between the texture and depth map based on the derived distortion model. In [80], a joint RC scheme is proposed where inter-view bit allocation are performed according to

the sequence complexity of each view, while the bits are allocated at fixed ratio between texture and depth map within each view. However in these algorithms, inter-view bit allocation is rarely considered or only simply allocated according to the sequence complexity. In 3DV, more general case involves m views coding, thus bits allocation among different views is highly desired.

In this work, the RC algorithm is proposed aiming at improving the overall quality in 3DV, where both the qualities of the coded views and the virtual views are considered. This is more reasonable, since both the virtual view and the coded view would be presented for viewing at the receiver side. On the other hand, the virtual view is synthesized by referencing nearby coded views, thus its quality depends on the coded references' quality. Different coded views are referenced by different number of virtual views. Intuitively, the coded view with more dependants should have better quality, as it would benefit more virtual views. In order to achieve the optimal R-D performance in 3DV, we first investigate the R-D characteristics of the texture and depth map. Then the quality dependency between the virtual view and the coded view is studied in the texture and depth map respectively. Based on the R-D characteristics of both the coded view and the virtual view, a bit allocation scheme is proposed for both the texture and depth map of all coded views. In this work, a simple case of multiview 3DV is discussed, where only three views are coded and two views are synthesized, but the bit allocation scheme and the conclusions derived in this work can be easily extend to n views cases.

1.4 Contributions of the Research

1.4.1 Quality Assessment

In the study of quality assessment, quality metrics are proposed for images and videos respectively. Firstly, quality metrics based machine learning are proposed for video quality assessment. Since perceptual quality is determined by a number of factors in HVS, it is difficult to determine the their relations and the parameters of model when a model is proposed to simulate the process in HVS. We use machine learning to find the proper relations. In addition, the simplify the problem, we decompose the quality assessment into multiple basic simple problem. One approach is that we classify video content into different groups according to its content complexity and build models for each group. Another approach is that we classify video according to distortion types. Therefore within each group, video has the same type of distortion, and we could simplify the problem by neglecting the effect of distortion type on quality asssessment.

Second, an IQM is proposed based MSE by incorporating the properties of HVS. The masking effect and the contrast sensitivity are two major properties of HVS that affect the perceptual quality of images and videos. Therefore in the proposed IQM and VQM, these two properties are investigated. For the proposed IQM, to find out the masking effect on perceptual image quality, we propose a method to measure the spatial randomness of the background with a spatial statistics model. Since a regular structure has strong spatial correlation among their neighborhood, which makes it easier to predict the pixel values from the neighboring pixel. Therefore, the prediction error actually reflects the randomness of background. The random background is less spatially predictable, resulting in larger prediction error. Thus the spatial prediction error is used as the measurement of

randomness, indicating how much the background could mask the noise. With this method, we have a randomness map to indicate the randomness of the structure at each pixel. For the proposed IQM, our contribution can be summarized in the following list

- We develop the spatial randomness to measure the masking effect quantitatively. A spatial statistical model is introduced to measure the regularity of the spatial structure of images.

- We propose a low pass filter to simulate the visual signal processing in the HVS. The low-pass filter is developed based on contrast sensitivity function and it removes the imperceivable error signals.

- By investigating the model of masking modulation, which mathematically analyzes how distortion is reduced with the proposed randomness measurement, an IQM is proposed based on MSE and it outperforms the benchmark IQMs according to our experimental results.

Third, an IQM is proposed based MSE by incorporating the properties of HVS. For the proposed VQM, the masking effect is investigated as well. However the video is more complicated than the image, since it has one additional dimension, the temporal dimension. The temporal activities in the video result the temporal masking that will affect the perceptual quality of the video. Therefore we propose to use the dynamic linear system to model the video signal and the temporal randomness of the video is developed based on the dynamic linear system. Moreover, due to the dynamic changes of the visual scene in video applications, it is usually impossible to observe all details within every frame. Our gaze is mainly driven to follow the most salient regions, unlike with images, where sufficiently long viewing times also allow to analyze the background regions. Thus the contrast sensitivity

on videos varies in different locations and using a constant low-pass filter over the whole frames of the video sequences is inappropriate. Therefore we developed a foveated low-pass filter to solve this problem. Our contribution on the VQM can be summarized in the following list

- We introduce a foveated low-pass filter to adaptively remove the high frequency signals according to the visual attention.
- We develop a dynamic linear model to simulate the video signal and use it to evaluate the temporal randomness and thus measure the masking effect in the video.
- By modifying MSE, a VQM is developed to simulate the masking effect. The developed VQM achieves precise perceptual quality prediction according to the experiment results.

1.4.2 Video Compression

In the study of video compression, we proposed a new tools for screen content video and introduce an optimized bit allocation scheme for 3D video coding.

First, a new coding tool called Edge Mode is proposed for HEVC intra coding, aimed at improving coding efficiency for screen content video. A set of edge modes that correspond to edge positions are identified based upon intra prediction directions. Then, a simpli- fied scheme is developed to select the best edge mode. To avoid ap- plying a transform over strong edges, directional 2D separable transforms are applied to blocks partitioned using these edge modes. Ex- perimental results show that HEVC with edge modes (HEVC/EM) can achieve up to an 17.9% reduction in bit-rate as compared to unmodified HEVC, with an average reduction of 10.4% for screen con- tent video sequences.

Second, a novel rate control scheme is proposed with optimized bits allocation for the 3D video coding. Firstly, we investigate the R-D characteristics of the texture and depth map of the coded view, as well as the quality dependency between the virtual view and the coded view. Secondly, an optimal bit allocation scheme is developed to allocate target bits for both the texture and depth maps of different views. Meanwhile a simplified model parameter estimation scheme is adopted to speed up the coding process. Finally, the experimental results on various 3D video sequences demonstrate the proposed algorithm achieves the excellent R-D efficiency and the bit rate accuracy comparing to the benchmark algorithms.

1.5 Organization of the Thesis

The rest of this thesis is organized as follows. In Chapter 2, the machine learning based quality assessment is introduced. In Chapter 3, the image quality metric for compressed images is proposed. In Chapter 4, the video quality metric for compressed video is proposed. In Chapter 5, new coding tools is proposed for screen content videos and optimized bit allocation is adopted for 3-D video coding. Finally in Chapter 6, future work is discussed.

Chapter 2

Quality Assessment Based on Distortion/Video Content Grouping

2.1 Overview of Distortion Grouping Approach

Various objective quality metrics have been proposed to predict the perceptual quality. VQM [102] is proposed using several features, which measure the information like contrast, motion, edge distortion of distorted video sequence. Then these features are linearly combined to provide a final quality score. MOVIE [109] was proposed by considering both the spatial and temporal distortion. Motion information extracted by optical flow estimation, is used to select proper filters from a set of Gabor filters. ST-MAD [120] was developed based on MAD [67] by considering the visual perception of motion artifacts. It currently achieved best performance on the LIVE database.

Although the performance of video quality assessment has been improved, it is still not accurate enough. The challenge comes from several aspects: first there are various distortion types, e.g. blurring, ringing, jitter, and they may affects the perceptual quality differently. Second, video signals are diverse in content. The video content could vary from low motion activity to high motion activity, from simple texture to complex texture, etc. These properties have different masking effect on

distortions and thus affect perceptual quality differently as well. However, most of works in the literature tried to handle all situations with an universal method without explicitly considering the effects of the video content and distortion types.

In this chapter, we first decomposed the quality assessment problem into simple cases, where only single distortion type and the video sequences distorted from the same original videos are considered. Then in each case, the perceptual quality can be simply predicted by the linear relation between the structure similarity index (SSIM) [124]. In order to decompose the problem, we classified all the distortions into local distortion and global distortion, based on the observation that the distortion that occurs in small spatial or temporal region has different impact on the perceptual quality than the distortion that occurs in the entire video sequence. A detection scheme is proposed to distinguish them automatically. Due to dependency of model parameters on video content, both temporal and spatial features are extracted to model the relation with the parameters of the linear models using a machine learning approach.

2.2 Video Quality Metric Derivation with Distortion Grouping

2.2.1 Linear relation between DMOS and SSIM

There are various types of distortions in digital images and videos, like blur, ring, compression distortion [78]. We classify all the distortions into two general types, which are global distortion and local distortion. The global distortion occurs almost in every pixel of every frame in video sequences, such as compression distortion, while the local distortion only appear in limited regions of limited number of

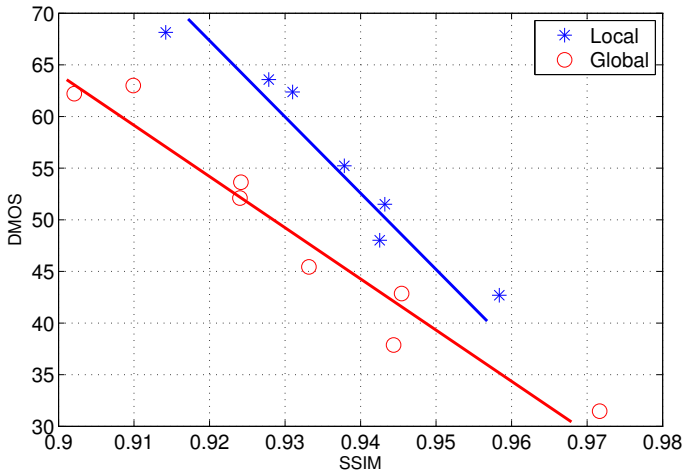


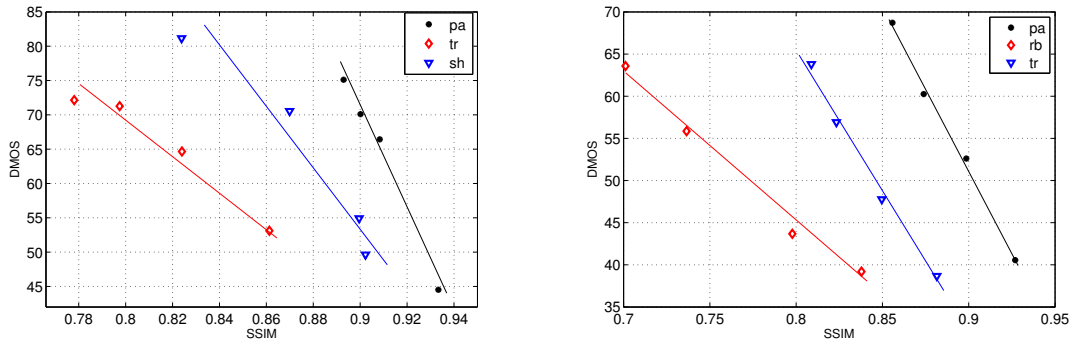
Figure 2.1: SSIM v.s. DMOS for the sequence “Rushhour" under different distortion types

frames, such as the distortions caused by transmission error, where the lost packets only corrupt corresponding blocks in a set of frames but other blocks still can be reconstructed correctly.

These two different distortion types have very different perceptual impact on the quality assessment. The global distortion covers large areas and last longer time, and usually the distortion is small and thus easy to be masked by the video content. While human eye is more tolerant to global distortion, the local distortion occurs in small region with large intensity, which is easy to attract human's attention and thus become noticeable.

On the other hand, SSIM was proposed to assessment perceptual quality by capturing the loss of image structure. The SSIM between original signal x and distorted signal y is calculated as

$$SSIM = \frac{(2\mu_x\mu_y + C1)(2\sigma_{xy} + C2)}{(\mu_x^2 + \mu_y^2 + C1)(\sigma_x^2 + \sigma_y^2 + C2)} \tag{2.1}$$



(a) Local Distortion (b) Global Distortion

Figure 2.2: Different linear relation of DMOS and SSIM for different sequences.

where μ_x, μ_y are mean; σ_x, σ_y are standard deviation and σ_{xy}^2 is cross variance.

However SSIM doesn't take into account the impact of different distortions on the perceptual quality. Fig 2.1 illustrates the relationship of SSIM and Difference Mean Opinion Score (DMOS) under different distortion types in the same sequence. The blue points correspond to the sequence distorted by H.264/AVC and MPEG-2 compression distortion and the red points are distorted by transmission error over wireless network and IP network. Here the compression distortion and transmission error can be classified as global and local distortion respectively. As shown in Fig. 2.1, SSIM is consistent with DMOS for the same type of distortion, but it is not consistent across the distortion types. The same SSIM may correspond to different DMOS depending on the distortion types. Therefore we should model the relationship between SSIM and DMOS differently for different distortion types. Meanwhile, from observation in Fig. 2.1, it can be assumed that the relation between SSIM and the actual perceptual quality, *i.e.*, DMOS, is approximated as linear for the same type of distortions.

Even under the same distortion types, the perceptual quality is affected by sequence contents as well. Different contents have different masking effect on the

distortion and consequently result in different relationships between DMOS and SSIM. Fig. 2.2 shows the DMOS-SSIM relation for different sequences under the same distortion type. In Fig. (2.2a), the points in each line are from the distorted sequences that share the same original sequences. We can see that although there is linear relation for each sequence but model parameters are quite different. The similar results can be observed in Fig. (2.2b) for global distortion.

Therefore based on the distortion types and video contents, a linear model is proposed between DMOS and SSIM as

$$DMOS = \begin{cases} \alpha^G(S_i) \cdot SSIM + \beta^G(S_i), & \text{if } S_i \in Global \\ \\ \alpha^L(S_i) \cdot SSIM + \beta^L(S_i), & \text{if } S_i \in Local \end{cases} \tag{2.2}$$

where α^G, β^G and α^L, β^L are model parameters for global and local distortion types respectively and they vary according to different sequence contents; S_i represents the video sequences. We use SSIM to model the relation with DMOS rather than other metrics, because although SSIM doesn't have good performance when different distortion types and different video contents are involved, but it has better performance for the same video content with same type of distortion.

2.2.2 Distortion classification scheme

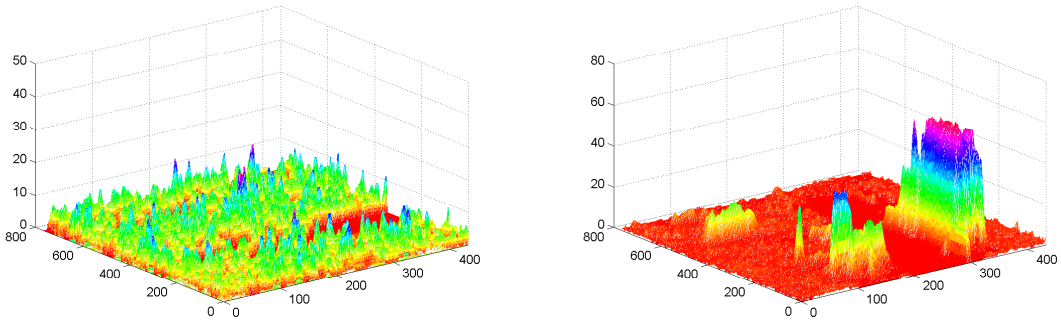
Since distortion types have significant effect on SSIM and it is critical to distinguish the distortion types before assessing the perceptual quality with SSIM. The distortion types can be identified both in temporal and spatial direction.

Since the local distortion only occurs within limited number of frames, the transition from distorted frames to undistorted frame will cause large peak signal-to-noise ratio (PSNR) change. Thus the potential frame that contains local distortion can be identified as

$$I^* = \underset{i=1\cdots N-1}{\operatorname{argmax}} |PSNR(i) - PSNR(i+1)| \tag{2.3}$$

where i is the frame index.

Detecting large PSNR change is not sufficient to determine the local distortion in video sequence, because global distortion is also possible to cause large PSNR change. For example, in H.264 compression, there is large difference in PSNR between I frames and P frames. Therefore we need to investigate spatial information inside the potential frame.



(a) Global (b) Local

Figure 2.3: Energy of filtered difference frame of sequence “Pedestrian Area"

The difference between original and distorted sequence is extracted and filtered with gaussian filter. The filtered difference is expressed as

$$\Delta F = Gaussian(|F_o(I^*) - F_d(I^*)|), \tag{2.4}$$

where $F_o(I^*)$ and $F_d(I^*)$ are the I^*th frame of original and distorted sequences respectively; $Gaussian(\cdot)$ is the gaussian filter. Typical filtered differences of two types of distortion are illustrated in Fig. 2.3. It can be observed that local distortion cause significant distortion in small area while global distortion cause small distortion over the entire frame. Therefore we calculate the variance of a selected set of pixel values as

$$V = var\left(\{p \mid p > \eta M, p \in \Delta F\}\right), \tag{2.5}$$

where $var(\cdot)$ is variance operation; p is the pixel value of difference frame; ΔF is filtered difference frame and M is the mean of ΔF; η is the constant parameter which is 1.2 in our work.

Table 2.1: Variance of difference frame in Eq. (2.5)

		pa	**rb**	**st**	**sf**	**bs**	**sh**	**mc**
L	**W1**	23.45	105.98	44.36	258.28	225.26	136.17	70.69
	W2	37.84	14.16	52.20	5.00	19.06	100.65	28.06
	W3	63.73	17.69	0.27	498.47	182.16	42.42	43.79
	W4	120.50	427.59	0.24	20.79	20.74	15.93	12.84
	IP1	51.30	352.12	61.82	129.70	116.95	2.11	28.03
	IP2	28.86	32.98	13.18	98.88	136.60	39.59	108.06
	IP3	92.15	19.19	8.34	7.40	93.84	9.19	91.10
G	**H1**	0.16	0.57	1.32	2.13	0.38	4.52	0.55
	H2	1.73	1.07	3.35	8.39	1.14	5.09	2.40
	H3	4.69	2.40	4.42	23.31	2.31	0.62	2.57
	H4	13.32	4.51	7.62	10.22	2.07	0.50	9.83
	M1	1.27	1.20	0.33	0.63	1.12	1.02	0.63
	M2	2.80	4.50	0.53	1.45	1.86	1.47	1.38
	M3	4.65	4.77	1.09	1.64	1.98	2.01	2.10
	M4	2.36	4.77	1.66	3.19	2.15	1.40	2.41

Table 2.1 presents the V of different sequences distorted by different distortions where “pr" to “mc" are test sequences from LIVE database; W1 to W4 and IP1 to IP3 are different levels transmission errors over wireless network and IP network

respectively, which is considered as local distortion; H1 to H4 and M1 to M4 are different levels compression distortion with H.264 codec and MPEG-2 codec, which is considered as global distortion. As shown in Table 2.1, most of V of local distortion are much larger than that of global distortion. Some of V of local distortion is as small as global distortion, that is because the transmission error only cause very small distortion that the local distortion is not obvious. Therefore we can distinguish the local distortion from global distortion by judging

$$V > th \tag{2.6}$$

where th is predefined threshold.

2.2.3 Content based parameters estimation

In order to improve the quality assessment, α^G, β^G and α^L, β^L in Eq. (2.16) need to be estimated according to the sequence content for each distortion type. Within same distortion type, the parameters only depends on the sequence contents. Since for each distorted sequence,we can access to its original sequence, four features are extracted from the original sequence. Then machine learning is employed to estimate the relation between the features and parameters α^G, β^G and α^L, β^L. Four features are described as follows.

1. Spatial Information (SI): Sobel filter is applied to each frame and standard deviation is calculated over all pixels within each filtered frame. SI is computed by averaging the standard deviation along the temporal direction as

$$SI = \frac{1}{N}\sum_{i=1}^{N} std_{space}[Sobel(F(i))] \tag{2.7}$$

where N is the number of frame in sequence and $std_{space}(\cdot)$ is the standard deviation operation.

2. Temporal Information (TI): it is based upon the motion difference feature, which is the difference between the pixel values (of the luminance plane) at the same location in space but at successive frames. The measure of TI is computed as the average over sequence of the standard deviation over frame

$$TI = \frac{1}{N}\sum_{i=1}^{N} std_{space}[|F(i) - F(i+1)|] \tag{2.8}$$

3. Contrast Information (CI): each frame is divided into NxN blocks, and the maximum pixel value and minimum pixel value of the blocks are used to compute the contrast and CI is calculated as the average over the sequence of contrast as

$$CI = \frac{1}{NM}\sum_{i=1}^{N}\sum_{j=1}^{M} \frac{Max(B(i,j)) - Min(B)(i,j)}{Max(B(i,j)) + Max(B(i,j))} \tag{2.9}$$

where $B(i,j)$ is the ith block of jth frame.

4. Luminance Mean(LM): mean of luminance of each frame is computed and LM is calculated as the average over sequence of mean of luminance

$$ML = \frac{1}{N}\sum_{i=1}^{N} mean(F(i)) \tag{2.10}$$

After extracting features, we applied machine learning approach to estimate the relationship between the features and model parameters. Among various machine

learning algorithms, support vector regression (SVR) is adopted due to its high performance. The basic concept of SVR is to find an optimal $\mathbf{w} = [w_1\ w_2\ w_3\ w_4\ w_5]^T$ such that the parameters can be predicted by linear function as

$$y = \mathbf{w}^T\mathbf{x} \tag{2.11}$$

where $\mathbf{x} = [SI \quad TI \quad CI \quad ML \quad 1]^T$ and y represents the parameters to estimate, *i.e.,* α^L, β^L, α^G, β^G. Among various kernel function for SVR, we applied Radial basis kernel function as

$$K(x_i, x_j) = exp(-\gamma||x_i - x_j||^2) \tag{2.12}$$

where γ is constant parameter. The optimal $\mathbf{w}$ is trained from training data, and it will be applied in Eq. (2.11) to predict the actual perceptual quality.

Finally, given the distortion types determined in Eq. (2.6) and model parameters estimated in Eq. (2.11), the DMOS can be predicted based on SSIM in Eq. (2.16).

2.2.4 Experimental Results

To evaluate the performance of the proposed objective quality assessment scheme, the Live video database is used, which consist of 150 distorted video sequences distorted by four types of distortions.

Parameter Estimation

In section 2.2.3, SVR is employed to model the relation between the features of sequence content and the parameters in Eq. (2.16). To verify its accuracy, five-fold cross validation is performed. First, least square linear regression is applied

between DMOS and SSIM to obtain actual α^G, β^G and α^L, β^L for each sequence. Then the data is divided into five sets and one set is selected as testing set while the rest sets are training sets.

In order to evaluate the accuracy of estimation, the following measurement is used as

$$E = \frac{1}{N}\sum_{i=1}^{N}\frac{|x_i - \bar{x_i}|}{x_i} \times 100\% \tag{2.13}$$

where E is the average error measured in percentage; x_i is one sample of actual value and $\bar{x_i}$ is the estimated value; N is the total number of samples. The results of each iteration is summarized in Table 2.2. We can see good accuracy is achieved that the average estimation errors for α^G, β^G and α^L, β^L are 7.05%, 6.02%, 10.93% and 9.31% respectively.

Table 2.2: Error of the model parameters prediction

Iter.	$\alpha^{\mathbf{G}}(\%)$	β^G (%)	α^L (%)	β^L (%)
1	10.04	8.80	6.54	5.76
2	6.78	5.95	9.93	8.94
3	11.36	9.19	25.66	20.84
4	6.35	5.38	5.91	5.09
5	0.71	0.80	6.63	5.94
Average	**7.05**	**6.02**	**10.93**	**9.31**

Performance Evaluation

To assess the performance of the proposed quality metric, several widely used measures: Pearson Correlation Coefficient (PCC), Spearman Rank Order Correlation Coefficient (SROCC), and Root-Mean-Squared-Error (RMSE) are used in our work. Several benchmark quality metrics, i.e., PSNR, SSIM, ST-MAD, VIF [112], VQM, VSNR [27], MOVIE are used for comparison.

The scatter plot of predicted DMOS vs DMOS along with the best linear fitting is shown in Fig. 2.4, including the sequences distorted with both local and global distortion.

Table 2.3 shows the performance of various quality metrics over LIVE database. We can see that the proposed quality metric achieves 0.869, 0.865 and 5.497 in terms of PLCC, SROCC and RMSE, which outperform all the rest of the benchmark quality metrics.

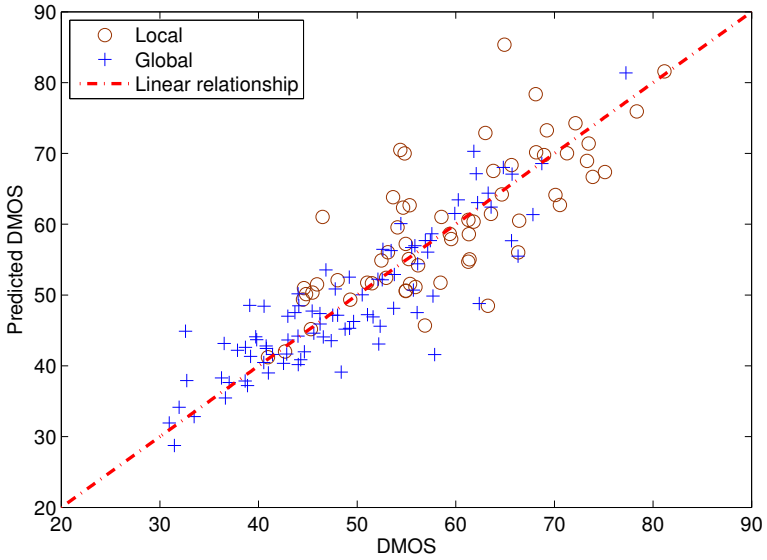


Figure 2.4: Predicted DMOS v.s. DMOS on the LIVE database

Table 2.3: Performance of various Video quality metrics

	PSNR	**SSIM**	**STMAD**	**VIF**	**VQM**	**VSNR**	**MOVIE**	**Our**
PCC	0.542	0.500	0.823	0.525	0.741	0.429	0.812	**0.869**
SROCC	0.523	0.525	0.825	0.527	0.725	0.422	0.789	**0.865**
RSME	9.175	10.977	6.118	10.977	7.349	9.914	6.413	**5.497**

2.3 Summary of Distortion Grouping Approach

We investigated the impact of video content and distortion types on the perceptual video quality. A detecting scheme was proposed and verified to effectively classify the distortions into either global or local distortion. A linear model is proposed based on SSIM to predicted the subjective quality for videos with the same types of distortion. The model parameters are sequence content dependent, which is predicted by machine learning approach with extracted feature from video content. The experimental results verify the effectiveness of the proposed objective video quality metric by achieving 0.869 in PCC on the LIVE database.

2.4 Overview of Video Content Grouping Approach

Reliable and accurate assessment of video quality plays an important role in improving the performance of a video processing system. Although subjective video quality assessment provides the most desired result, it is time-consuming, laborious and cannot be conveniently integrated in a fully automated system. A large amount of efforts have been put on objective quality assessment in recent research [134, 109]. Most of them have worked on the development of new universal quality indices in measuring distorted video of various distortion types and these metrics are usually tested in video quality databases such as the LIVE database [45] and the EPFL-PoliMI database [8], where several video distortion types are included. Despite these efforts, it remains a challenging problem for a video quality index to achieve good performance for multiple distortion types.

In practical applications, the number of distortion types of people's interest is actually limited, and the distortion caused by video coding is one of them. Compression is essential to video storage and delivery. Quality assessment of compressed video can be used to compare the performance of different coders. Besides, it plays a key role in developing perceptual coders. That is, it can assist an encoder to make the optimal decision in perceptual coding [121, 50].

The peak-signal-to-noise-ratio (PSNR) and the mean-squared-errors (MSE) indices are often used as quality indices in the coding community. Although there has been criticism on their suitability for ignoring the human perception factor, they do offer two attractive features: 1) computational simplicity and 2) fine granular scores. The latter is especially important since the quality of videos coded by different encoders could be quite close to each other. A coarse-scale mean opinioin score (MOS) system obtained from the traditional subject test may not be sufficient to differentiate their sutble difference. Instead, we may demand the pairwise comparison by a few gold eyes. Furthermore, PSNR and MSE still work well for some distortion such as the quantization noise as reported in [78, 77].

Due to aforementioned reasons, one way to develop new image/video quality indices is to improve PSNR- and MSE-based quality indices [38, 105]. We follow the same methodology in this work. First, we observe that there exists a linear relationship between the MOS and a logarithmic function of the MSE value of coded video for a range of coding rates. This linear model is validated by experimental data. The model contains one parameter to be determined by video characteristics. Next, we propose a two-stage algorithm to estimate this parameter based on machine learing. To reduce the prediction error, videos of similar characteristics are grouped together, which is achieved by selected features, in the first stage. Then, the model parameter is trained and predicted within each video group in

the second stage. Experimental results on a coded video database are given to demonstrate the effectiveness of the proposed algorithm.

The rest of this chapter is organized as follows. The linear model between the MOS and a logarithmic function of the MSE value is presented in Sec. 2.5.1. The process of estimating the model parameter is detailed in Section 2.5.2. Experimental results are provided to demonstrate the effectiveness of the proposed quality index in Sec. 2.5.3. Finally, concluding remarks are given in Sec. 2.6

2.5 Video Quality Metric Derivation with Video Content Grouping

2.5.1 Linear Relationship between MOS and Log-MSE

Consider the following log-MSE function:

$$D_{LOG-MSE} = \log\left(\frac{1}{NWH}\sum_{i=1}^{N}\sum_{x,y=1,1}^{W,H}(P_{(x,y,i)} - \hat{P}_{(x,y,i)})^2\right), \tag{2.14}$$

where i is the frame index, N is the total number of frames in a video clip, $P_{(x,y,i)}$ and $\hat{P}_{(x,y,i)}$ are pixel values of the original and distorted sequences, and W and H are the width and the height of each frame, respectively. We show the plot of the MOS value versus the $D_{LOG-MSE}$ value for coded video sequences from the MCL-V database [72] in Fig. 2.5. Each point in this figure corresponds to a coded video sequence. All connected points share the same original video yet they are coded with different bit rates.

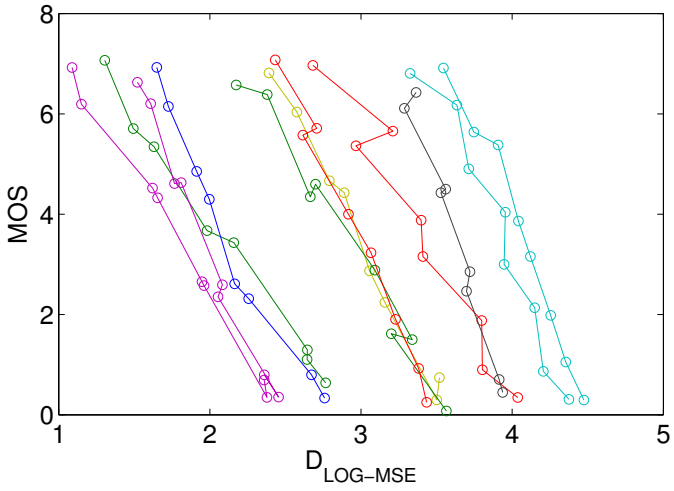


Figure 2.5: The MOS-versus-Log-MSE (MLM) plot for coded video sequences from the MCL-V database.

For the same video, we observe a linear relationship between the MOS value and the $D_{LOG-MSE}$ value in Fig. 2.5, which can be approximated as

$$MOS = \alpha(C_i) \cdot (D_{LOG-MSE} - \beta(C_i)), \tag{2.15}$$

where $\alpha(C_i)$ and $\beta_i(C_i)$ are parameters for the ith video content, denoted by C_i. Furthermore, as shown in Fig. 2.5, these lines are almost in parallel with each other. This means that $\alpha(C_i)$ is nearly a constant. Under such an assumption, one can further simplify the two-parameter MOS-versus-Log-MSE (MLM) model into a one-parameter MLM model in form of

$$MOS = \alpha \cdot (D_{LOG-MSE} - \beta(C_i)). \tag{2.16}$$

The simplified linear MLM model in Eq. (2.16) is depicted in Fig. 2.6. Each line has a horizontal shift determined by parameter $\beta(C_i)$.

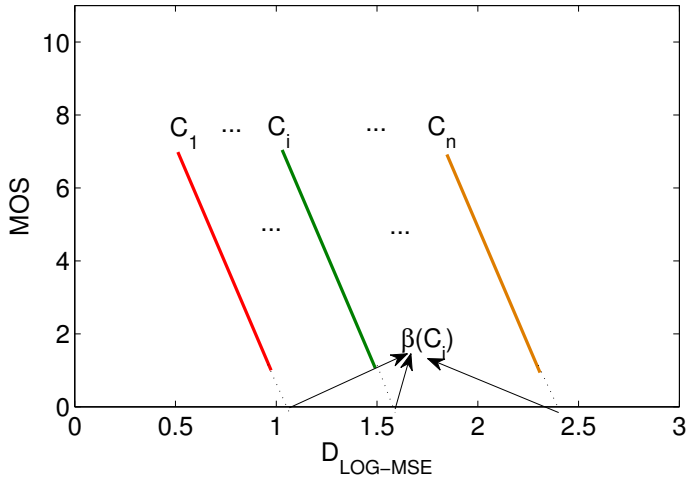


Figure 2.6: A simplified linear relationship between MOS and $D_{LOG-MSE}$ parameterized by $\beta(C_i)$.

2.5.2 Model Parameter Estimation via Machine Learning

Due to different video characteristics, the linear MLM model has a different parameter value, $\beta(C_i)$, in Eq. (2.16). In this section, we propose a two-stage method to estimate this parameter. It is worthwhile to point out that, although the value of $\beta(C_i)$ varies over a large range, lines in Fig. 2.5 are clustered into several groups, which means $\beta(C_i)$ of those videos are similar. To reduce the estimation error, it is desirable to classify video contents into groups in the first stage and then predict the value of $\beta(C_i)$ within each group in the second stage.

Stage I: Video Grouping

Proper features can be selected to capture different characteristics of video content for the grouping purpose. The horizontal shifts in the parallel lines in Fig. 2.5 are caused by the masking effect of the video content. Simply speaking, the masking

effect is related to video complexity. One well known measure for video content complexity is the Spatial Information (SI) [54], which is defined as

$$SI = max_{time}\{std_{space}[Sobel(F(i))]\} \tag{2.17}$$

where $F(i)$ is the ith frame, $Sobel(\cdot)$ stands for the sobel filter, std_{space} is the operation of computing the standard deviation in the space domain. SI provides a simple yet efficient way to estimate the complexity of video content. Typically, a video clip of a higher complexity has a larger SI value.

To estimate the local smoothness of the content, we introduce another feature, called homogeneity (H), in form of

$$H = \frac{1}{N}\sum_{i=1}^{N}\sum_{l,k=1,1}^{L,K}\sigma^2(l,k,i), \tag{2.18}$$

where $\sigma^2(l,k,i)$ is the variance of a block of size 8×8, and where (l,k) and i are, respectively, spatial and temporal indices of non-overlapping blocks in video sequences.

For a more complex video content, its H and SI values are larger and the line of its MLM model is shifted to further right in Fig. 2.6, leading to a larger value in $\beta(C_i)$. With this observation, we can classify videos into a different range of $\beta(C_i)$ based on these two features.

We classify videos into three groups with small, medium and large $\beta(C_i)$ based on the following decision:

$$C_i(SI, H) \in \begin{cases} G_S, & w_1 \cdot SI + w_2 \cdot H < Th_1, \\ G_M, & Th_1 \leq w_1 \cdot SI + w_2 \cdot H \leq Th_2, \\ G_L, & Th_2 \leq w_1 \cdot SI + w_2 \cdot H, \end{cases} \tag{2.19}$$

where G_S, G_M and G_L represent groups of video contents with small, medium and large $\beta(C_i)$, respectively, w_1 and w_2 are two weighting factors and Th_1 and th_2 (with $Th_1 < Th_2$) are two thresholds. The classification boundary in the feature space is shown in Fig. 2.7. The classification result in the MLM plot is shown in Fig. 2.8. Clearly, the $\beta(C_i)$ values are closer within the same group.

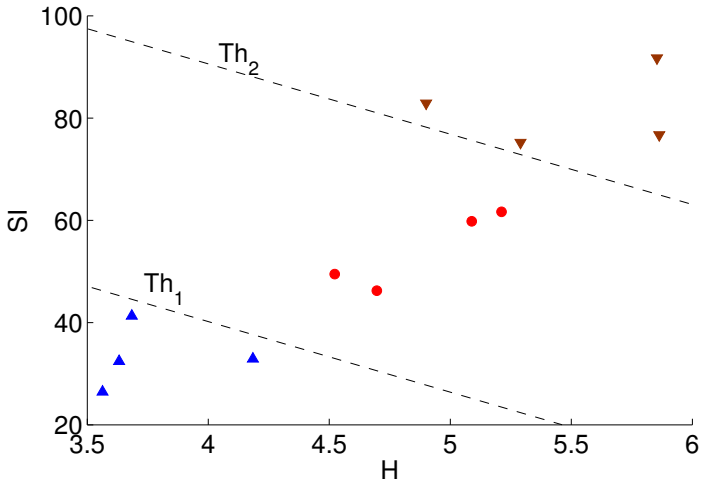


Figure 2.7: Classification boundaries in the joint feature space of H and SI.

Stage II: Parameter Estimation

With video content grouping, the variation of $\beta(C_i)$ within each group is reduced significantly. However, there still exist a small range of variations. To provide

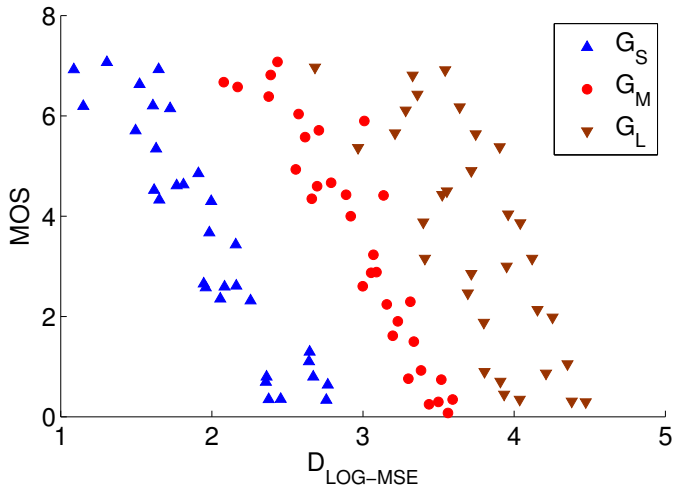


Figure 2.8: Classification results in the MOS-versus-Log-MSE (MLM) plot.

a more accurate estimate of $\beta(C_i)$ for video content C_i, two more features are introduced to capture different aspects of content characteristics.

Temporal properties are important characteristics for video content. We adopt the following temporal information (TI) [54]

$$TI = max_{time}\{std_{space}[M(i)]\}, \tag{2.20}$$

where $M(i)$ is the difference between the ith and the $i-1$th frames in the sequences. In words, we take the maximal value over time of the standard deviation of frame differences in the spatial domain. TI represents the complexity in the temporal domain. Usually, a video with a large amount motion will lead to a larger TI value.

Furthermore, we introduce the contrast feature

$$C = \frac{1}{WH}\sum_{i,j=1,1}^{W,H} LC(i,j) \tag{2.21}$$

where $B_{i,j}$ is a $L \times L$ block centered at position (i, j), K is the total number of pixels in block $B_{i,j}$ and

$$LC(i,j) = \frac{1}{K} \sum_{(l,k) \in B_{i,j}} \frac{|P(l,k) - P(i,j)|}{P(l,k) + P(i,j)} \tag{2.22}$$

is the local contrast of block (i, j).

The machine learning methodology is then used to build the relation between $\beta(C_i)$ and four selected features via

$$\hat{\beta}(C_i) = f(\mathbf{X}(C_i)), \tag{2.23}$$

where $\mathbf{X}(C_i) = [SI, H, TI, C]^T$ is the feature vector of content C_i, $f(\cdot)$ is the relation obtained by training and $\hat{\beta}(C_i)$ is the predicted parameter for the test video. Among various machine learning algorithms, we choose the Support Vector Machine (SVM) method with the radial basis kernel function [14]

$$K(x_i, x_j) = exp(-\gamma ||x_i - x_j||^2), \tag{2.24}$$

where γ is a constant parameter, due to its good performance. The leave-one-out cross-validation is used to estimate the parameters of all sequences within each group.

The mean absolute error and mean relative error are used to measure the accuracy of estimated $\beta(C_i)$. They are calculated as

$$AE = \frac{1}{N} \sum_{i=1}^{N} |\beta(C_i) - \hat{\beta}(C_i)|, \tag{2.25}$$

$$RE = \frac{1}{N} \sum_{i=1}^{N} \frac{|\beta(C_i) - \hat{\beta}(C_i)|}{\beta(C_i)} \times 100\%. \tag{2.26}$$

The prediction accuracy in each video group is shown in Table 2.4. We see that the prediction error is small in each group, *i.e*, AE is up to 0.20 and RE is up to 7.37%. The error in G_M is particularly small since the values of $\beta(C_i)$ in this group are closer than those in other groups. Moreover, the average $\beta(C_i)$ listed in Table 2.4 validates that G_S, G_M and G_L are groups of video content with small, medium and large $\beta(C_i)$ values.

Table 2.4: Estimation accuracy

Group	**Average** $\beta(C_i)$	**Absolute Error**	**Relative Error**
G_S	1.99	0.14	7.37(%)
G_M	2.96	0.06	1.91(%)
G_L	3.75	0.20	5.29(%)

Based on $\hat{\beta}(C_i)$ in Eq. (2.23) and $D_{LOG-MSE}$ in Eq. (2.14), the MOS value can be predicted via Eq. (2.16), where parameter α in Eq. (2.16) can be determined by the training video sequences.

2.5.3 Experimental Results

We evaluated the performance of the proposed video quality index based on the Log-MSE function against the MCL-V database. The MCL-V video quality database contains 12 source video sequences of resolution 1920×1080. The distortion is introduced by encoding each sequence with the H.264/AVC video codec and/or upscaling after compression.

The Pearson Correlation Coefficient (PCC) and the Spearman Rank Order Correlation Coefficient (SROCC) are used to assess the performance of the proposed quality index. Several well known indices, including PSNR, SSIM [124], PSNR-HVS-M [105], VQM and MOVIE[1] were computed in the experiment for the

[1]Due to the limited computational capability, the frame interval is set to 32, instead of default value 8 while running MOVIE.

purpose of performance benchmarking. The performance of these quality indices is compared in Table 2.5. As shown in the table, the proposed quality index achieves 0.871 in PCC and 0.887 in SROCC and outperforms all other benchmarking indices by a significant margin.

Table 2.5: Performance comparison of video quality indices with respect to the MCL-V video quality database.

Indices	**PCC**	**SROCC**
PSNR	0.472	0.464
SSIM	0.456	0.470
PSNRHVS	0.532	0.518
VQM	0.761	0.783
MOVIE	0.676	0.675
Ours	**0.871**	**0.887**

The scatter plot of the actual and predicted MOS values is shown in Fig. 2.9. We see that these points are concentrated on a narrower strip, indicating good correlation between them. Such improvement comes from accurate estimation of $\beta(C_i)$ for each individual sequence.

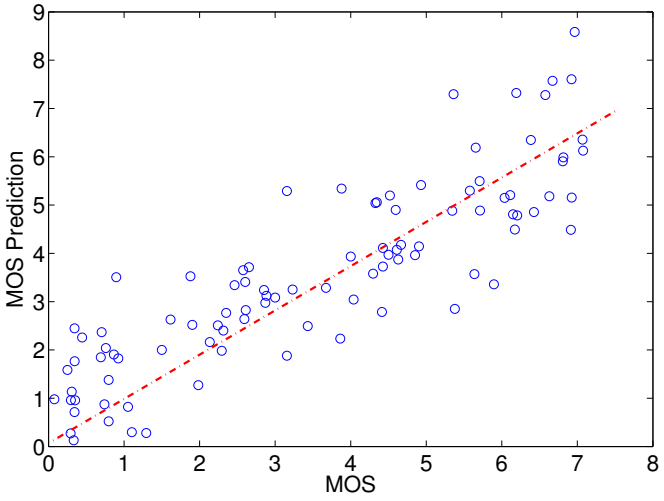


Figure 2.9: The scatter plot of the actual and predicted MOS values for various coded video sequences.

2.6 Summary of Video Content Grouping Approach

An objective video quality index was proposed by considering a log-MSE function with a horizontal shift, where the shift parameter was predicted based on video grouping and training/testing within the same video group. Experiments were conducted on the MCL-V video quality assessment database and the proposed quality index outperforms all benchmarking video quality indices by a significant margin. In the near future, we would like to develop even better MOS prediction algorithms for high quality coded video content such as UHDTV contents.

Chapter 3

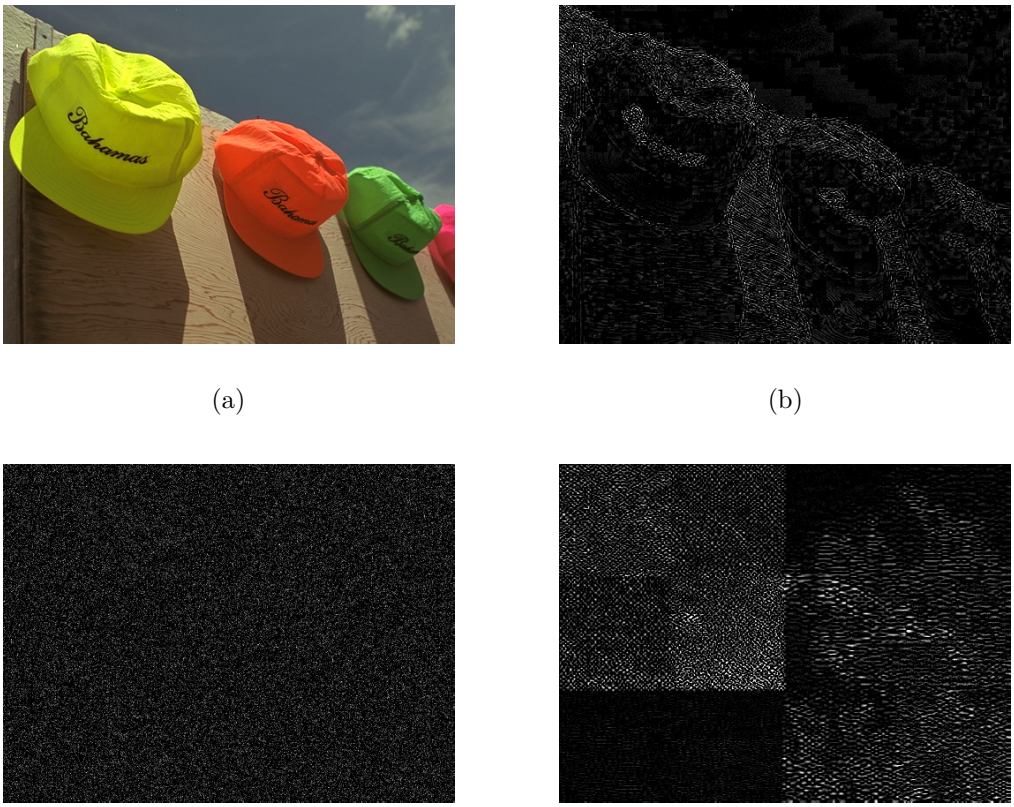
Quality Assessment for Compressed Images

Reliable assessment of image quality is important in improving the performance of image processing systems. Due to the inconvenience of subjective image quality assessment, a large number of objective image quality metrics (IQM) have been developed. Generally, there are two different categories of objective IQMs. In the first category, the characteristics of Human Visual System (HVS) are explored and incorporated into IQM algorithms [85, 21, 82, 130, 63, 27, 75, 137]. In [85], the luminance adaptation and the Contrast Sensitivity Function (CSF) of HVS are considered in human's perception to luminance difference. In [21], a wavelet CSF is employed and the distortion is analyzed in multiple channels after the wavelet transform. In [63], the Haar wavelet is used to model the space-frequency localization property of HVS responses. In [130], a model of noise detection threshold is proposed to determine the visibility of discrete wavelet transform noise in image compression, which is similar to the concept of just noticeable distortion (JND) [82]. In [27], the noise thresholds are determined on contrast via CSF, and two-stage schemes are proposed for the distortion less or larger the threshold. Recently, visual attention has been studied extensively for IQMs [75, 137]. Due to non-uniform distribution of the photo receptors on the retina and visual attention that drives the most sensitive part on interesting objects, images are not perceived with the same resolution for each region and the visual attention drive

the eye and make the most sensitive region of region focus on interenting objects. Therefore the distortion is not perceived equally and should be given different weights. In the second category, rather than simulating the process of HVS, IQMs are proposed from the view of signal processing by involving image properties like structure information [124, 125, 126], statistical information [111, 112]. In [124], the structural similarity is computed using local mean and variance and the overall performance is measured by averaging the local structural similarity. In [111] and [112], the information fidelity criterion is proposed by quantifying the information shared between a reference and a distorted image. Recently the edge or gradient similarity have been proved effective in modeling IQMs [140, 73, 141]. More HVS based image quality metrics could be found in the literature such as [106, 122].

Most of the above IQMs are aimed at handling a large range of distortion types and usually tested in databases with multiple distortion types such as the TID database. However developing an universal quality metric is quite challenge. Due to the wide application of image compression in image delivery and storage, the compression distortion is one of major distortion among various distortion types. Besides, IQM plays a key role in image coding in the processes such as Rate-Distortion Optimization (RDO) [56, 50, 121]. Therefore, it is highly desired to have accurate IQMs for compressed images.

Compression distortion could include various types of visual artifacts, which mainly are blurriness, blocking and ringing artifacts. In fact, compression distortion has its unique characteristics comparing to other distortion types. Masking effect is widely exploited in the image codecs, and that makes compression distortion content dependent. In codecs, high frequency components usually are quantized with larger quantizers than low frequency components. Moreover, for prediction based codecs, larger prediction residual in complicated area could also



(a) (b)

(c) (d)

Figure 3.1: Compression distortion is content dependent. (a) Original image. (b) Compression distortion. (c) Additive distortion. (d) Transmission error distortion.

result in larger distortions. In addition, most perceptual image codecs try to hide distortion in the area that has large masking effect. Therefore as shown in Fig. 3.1, the distortion relates to original image that it is larger in complex content than in smooth content. On the other hand, masking effect from complex content could significantly prevent the distortion being perceived. Therefore the masking effect become critical to the compression distortion and it is important to make a quantitative analysis of the masking effect on MSE.

Masking effect refers to human's reduced ability to detect a stimulus on a spatially or temporally complex background. The traditional way to measure the

masking effect is using a divisive gain control method, which decomposes the image into multiple channels and analyzes the masking effect among the channels by divisive gain normalization [66, 129, 119, 84, 107]. However, the mechanism of gain control mostly remains unknown. Additionally, since only simple masker such as sinusoidal gratings or white noise is used in the experiments to search for optimal parameters to fit the gain control model, there is no guarantee that these models are applicable to natural images [26]. In [128] and [47], it is pointed out that masking effect highly depends on the level of randomness created by the background. Usually the regular background contains predictable content and the stimulus will become distinct from neighborhood when it is different from human's expectation of its position. While in the random background, the content is unpredictable, and thus any change on it will be less noticed. Therefore, there is higher masking in the random background than the regular background. In [128], a concept of entropy masking is proposed to measure masking effect of background using zero order entropy. However, it fails to consider the spatial relation of pixel values. In addition, a single value might not be enough to indicate randomness of the whole background, because the content in the background may vary significantly. Furthermore, only with masking measurement is insufficient to predict the perceptual distortion, because it is unclear how the proposed masking measurement affects the perceived distortion.

In this chapter, we first propose a method to measure the randomness of the background with a spatial statistics model. Since a regular structure has strong spatial correlation among their neighborhood, which makes it easier to predict the background from the neighborhood. Therefore, the prediction error actually reflects the randomness of background. The random background is less spatially predictable, resulting in larger prediction error. Thus the spatial prediction error

is used as the measurement of randomness, indicating how much the background could mask the noise. With this method, we have a randomness map, rather than a single value, to indicate the randomness of the structure at each pixel. Then we investigate the model of masking modulation, which mathematically analyzes how distortion is reduced with the proposed randomness measurement based on the observation of perceptual qualities in terms of MOS in different databases. Meanwhile, we propose a simple but effective preprocessing scheme, which removes the imperceivable error signals.

3.1 Randomness Measurement

The visual signal is affected by masking effect and the visibility of compression distortion significantly depends on the background of the images. Usually the distortion is easy to be observed in the regular region and hard to be perceived in disordered regions. To measure the masking effect of the image content, the spatial randomness of image structure should be measured. In this section, the randomness is measured quantitatively using the spatial estimation error. Meanwhile proper selection of prediction neighborhood is discussed as well.

3.1.1 Randomness measured with spatial statistics

For regular structure, the pixels always have strong correlation with the neighboring pixels and the presence of particular combinations of neighboring pixels will increase the possibility of certain values of the current pixel. On the other hand, for a disordered structure, the neighboring pixels will provide less useful information to estimate the current pixel.

Let $Y(u)$ and $\mathbf{X}(u)$ be jointly distributed random variable and random vector standing for the current pixel and neighboring pixels, respectively. At a particular position, $y(i,j)$ is an example of $Y(u)$ and similarly $\mathbf{x}(i,j)$ is an example of $\mathbf{X}(u)$ representing the neighboring pixels. The reasonable estimation of $y(i,j)$ is $E(y(i,j)|\mathbf{X}(u) = \mathbf{x}) = \sum_{y(i,j)\in\mathcal{S}} y(i,j)P_{Y|X}(y|\mathbf{x})$ where $P_{Y|X}(y|\mathbf{x})$ is conditional probability of y given $\mathbf{X}(u) = \mathbf{x}$ and $\mathcal{S}$ is the set of all possible y. However the estimation of $P_{Y|X}$ is not easy and thus we assume a linear estimation that

$$\hat{Y}(u) = \mathbf{H}\mathbf{X}(u), \tag{3.1}$$

where $\mathbf{H}$ is an $1 \times n$ matrix. The optimal $\mathbf{H}^*$ is determined by achieving the minimum mean of the error $|(Y(u) - \hat{Y}(u)|$ over all possible combination of $Y(u)$ and $\mathbf{X}(u)$, which is expressed as

$$\mathbf{H}^* = \underset{\mathbf{H}\in\mathcal{R}^{1\times n}}{\operatorname{argmin}} \quad E[(Y(u) - \mathbf{H}\mathbf{X}(u))^2], \tag{3.2}$$

where $E[\cdot]$ is the expected value operator. To achieve the optimal value, the following equation must be satisfied as

$$\frac{\partial E[(Y(u) - \mathbf{H}\mathbf{X}(\mathbf{u}))^2]}{\partial \mathbf{H}} = 2\mathbf{H}^* \cdot E[\mathbf{X}(u)\mathbf{X}(u)^T] - 2E[Y(u)\mathbf{X}(u)^T] = \mathbf{0}, \tag{3.3}$$

where T is the transpose operator. From Eq. (3.3), we could have $H^* = E[Y\mathbf{X}(u)^T]E[\mathbf{X}(u)\mathbf{X}(u)^T]^{-1}$ and hence the optimal estimation of $y(i,j)$ given the neighboring pixels $\mathbf{x}$ is

$$\hat{y}(\mathbf{x})(i,j) = \mathbf{R}_{YX}\mathbf{R}_X^{-1}\mathbf{x}(i,j), \tag{3.4}$$

where $\mathbf{R}_{YX} = E[Y\mathbf{X}(u)^T]$ is the cross-correlation matrix between $\mathbf{X}(u)$ and $Y(u)$ and $\mathbf{R}_X = E[\mathbf{X}(u)\mathbf{X}(u)^T]$ is the correlation matrix of $\mathbf{X}$. $\mathbf{R}_{YX}$ and $\mathbf{R}_X$ carry the structure information of image content and vary as the image structure changes.

If the neighboring pixels x_i, (*i.e.,* the components in $\mathbf{X}$) are linear dependent, $\mathbf{R}_X$ is not full rank and thus it is not invertible in Eq. (5.27). For example, in exactly plain regions, the structural information is so limited that the rank of $\mathbf{R}_X$ is actually one. In such a case, $\mathbf{R}_X^{-1}$ in Eq. (5.27) could be replaced by pesudo-inverse $\tilde{\mathbf{R}}_X^+$, which is expressed as

$$\tilde{\mathbf{R}}_X^+ = \mathbf{U}_m \Lambda_m^{-1} \mathbf{U}_m^T, \tag{3.5}$$

where Λ_m is the eigenvalue matrix of all non-zero eigenvalues of matrix $\mathbf{R}_X$ and $\mathbf{U}_m$ is the corresponding eigenvector matrix. As proved in appendix A, the pesudo-inverse operation also provides the best estimation. Actually $\tilde{\mathbf{R}}_X^+$ is a generalized form of $\mathbf{R}_X^{-1}$. When $\mathbf{R}_X$ is full rank, they are equivalent.

The randomness of the structure could be measured by the estimation error from the neighborhood with structural correlation as

$$S(i,j) = |y(i,j) - \mathbf{R}_{YX}\tilde{\mathbf{R}}_X^+ \mathbf{x}(i,j)|. \tag{3.6}$$

The large value of $S(i,j)$ means the structure is more disordered and thus contains more randomness. On the other hand, for the regular structure, $S(i,j)$ will be close to zero.

3.1.2 Estimation of local statistics

$\mathbf{R}_{YX}$ and $\mathbf{R}_X$ are the local properties of image content patterns, and change with image content. They could be estimated from pairs of y and $\mathbf{x}$ within local regions. A block with the size of $M \times M$ centered at y is used to extract the samples

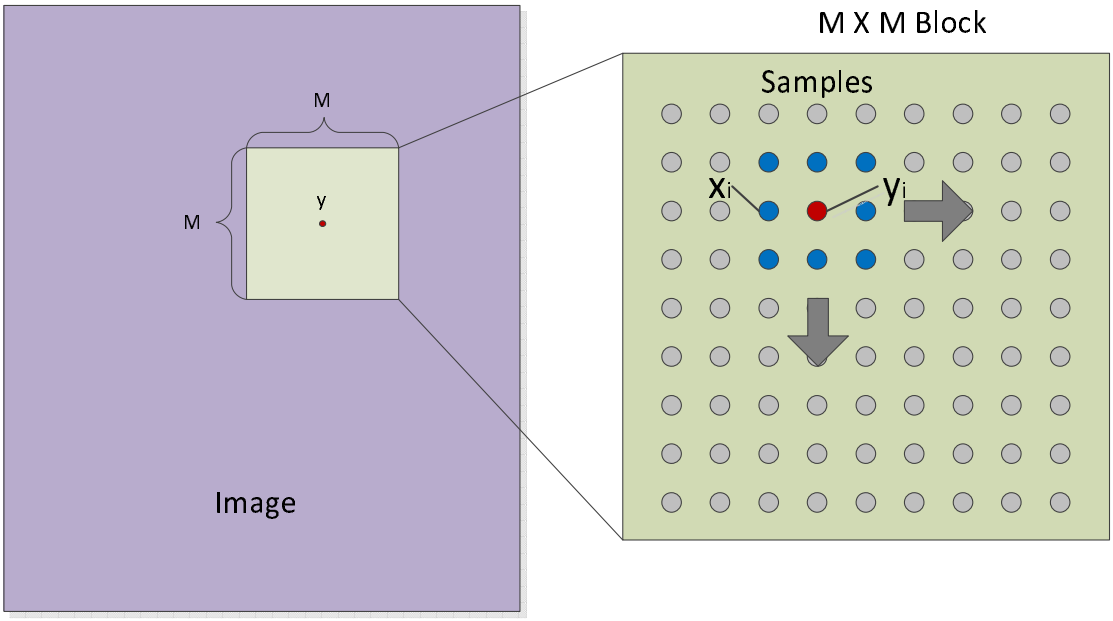


Figure 3.2: Demonstration of sample extraction for $y(i,j)$ and $\mathbf{x}(i,j)$.

as shown in Fig. 3.2. The extracted samples are $\mathbf{X_S} = [\mathbf{x_1}, \mathbf{x_2}, \cdots, \mathbf{x_N}]^T$, and $\mathbf{Y_S} = [y_1, y_2, \cdots, y_N]^T$, where N is the number of samples depending on the size of local block M and $\mathbf{x_i}$ and y_i are sample pairs in a particular position. The unbiased estimations of $\mathbf{R}_X$ and $\mathbf{R}_{YX}$ could be calculated from the sample correlation matrix and the sample cross-correlation matrix as

$$\hat{R}_X = \frac{1}{N-1} X_S X_S^T, \quad \hat{R}_{YX} = \frac{1}{N-1} Y_S X_S^T, \tag{3.7}$$

By replacing $\mathbf{R}_{YX}$ and $\mathbf{R}_X$ in Eq. (3.6) with their estimation in Eq. (3.7), we could estimate the randomness with local structure information.

3.1.3 Sparse sampling of neighborhood

The choice of neighboring pixels is not limited to the adjacent pixels. Only the closest neighboring pixels are not enough to capture the structure information of

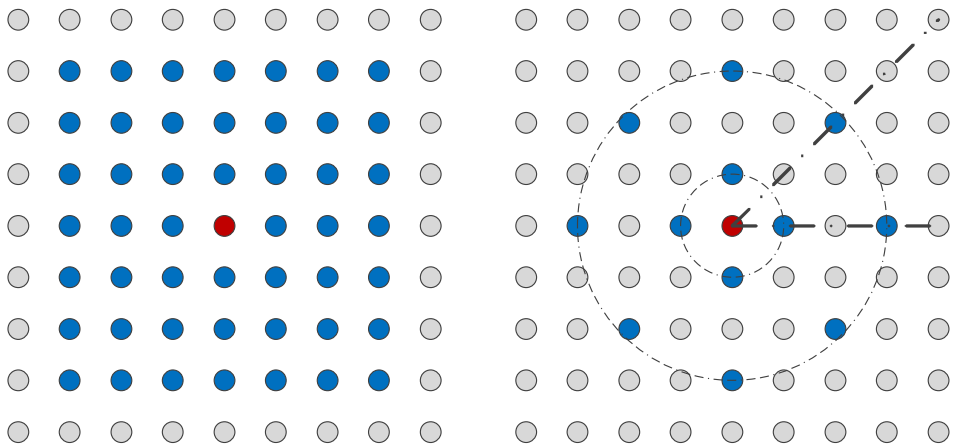


Figure 3.3: Different neighborhood sampling. (a) Dense sampling. (b) Sparse sampling.

the patterns with large size. Thus more neighboring pixels within reasonable distance should be included as shown in Fig. 3.3 (a). A large size of neighborhood will increase the number of neighboring pixels and consequently will increase the computational complexity to estimate the randomness. Usually the dense neighboring pixels as shown in Fig. 3.3 (a) may contain significant redundancy. In order to achieve a proper size of neighborhood while maintaining a small number of neighboring pixels, the neighboring pixels are evenly sampled from the neighborhood as shown in Fig. 3.3 (b), and the sampled neighboring pixel set could be expressed in a polar coordinate system as

$$V = \left\{(\theta, r)|\theta = \frac{k\pi}{2}; r = 2l + 1 \leq L\right\} \bigcup \left\{(\theta, r)|\theta = \frac{(2k+1)\pi}{4}; r = 2\sqrt{2}l \leq L\right\}, \quad (3.8)$$

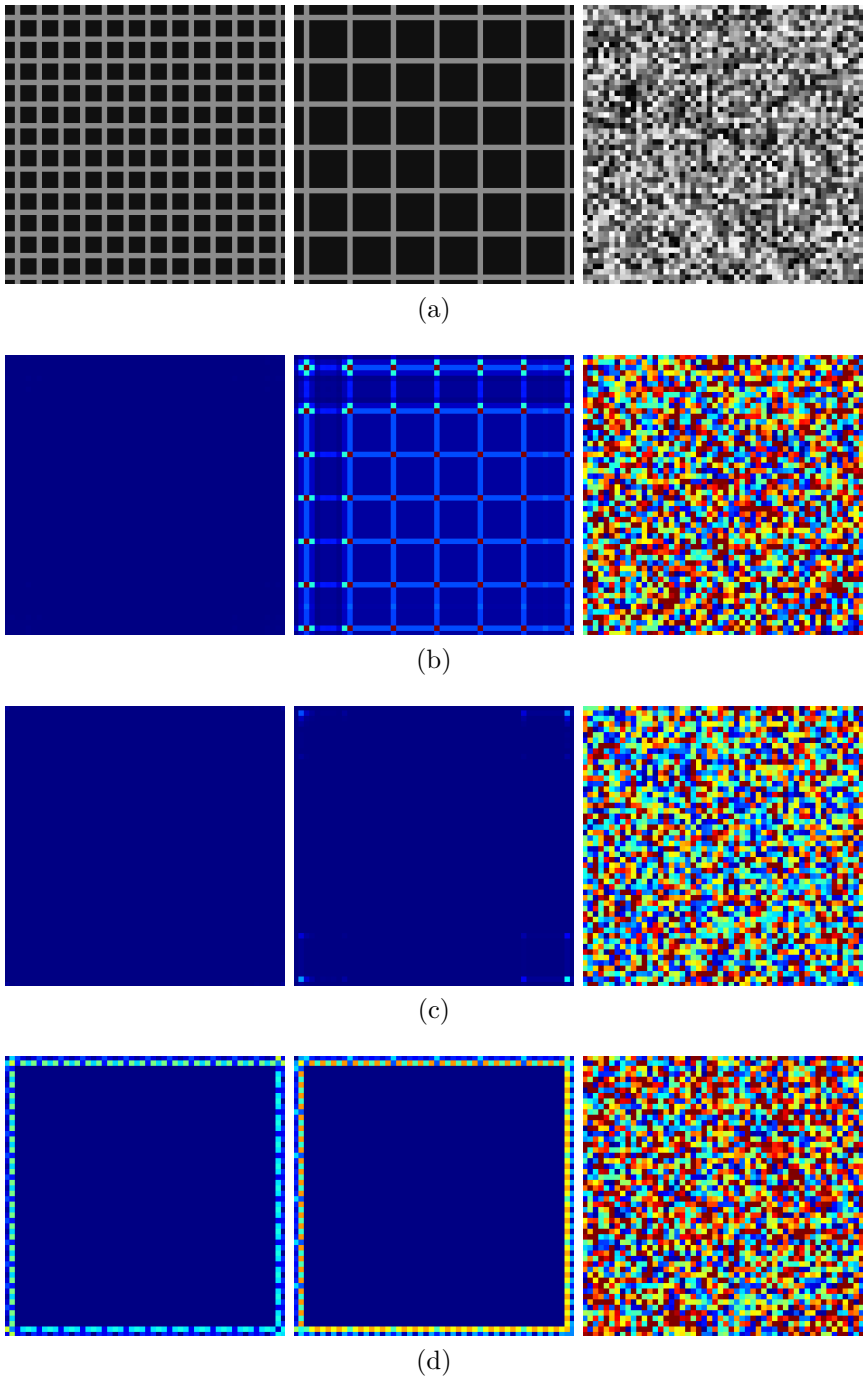
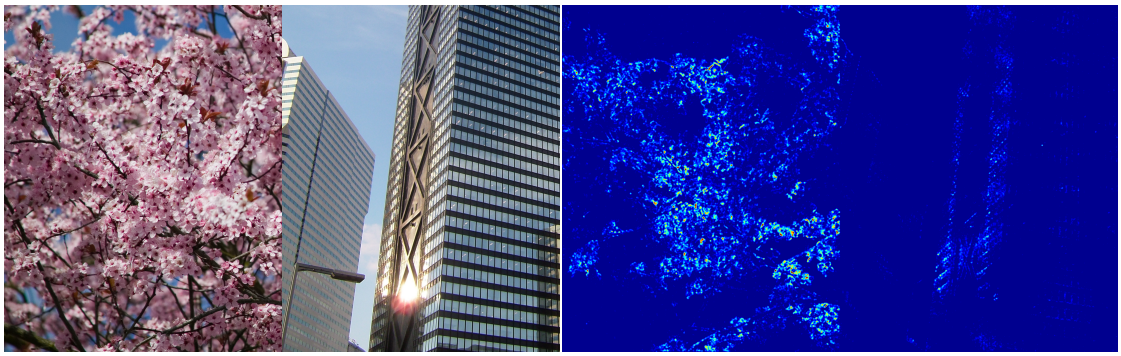


Figure 3.4: Different patterns and the heat maps of randomness with different size of neighborhood. The images in each column are original images and the corresponding randomness maps with different methods. (a) Regular patterns with the size of 16 and 32 pixel respectively and a random pattern (b) Dense sampling within a block of 9×9 size. (c) Dense sampling within a block of 17×17 size. (d) Sparse sampling within 17×17 block.



(a) (b)

Figure 3.5: Illustration of randomness. (a) Original image. (b) Heat map of randomness.

where $k = 0, 1, 2, 3$, and $l = 1, 2, \cdots$, N; L is the size of neighborhood. Please note that the sampling method is not unique and the sampling method as illustrated in Fig. 3.3 (b) is adopted due to its simplicity and effectiveness.

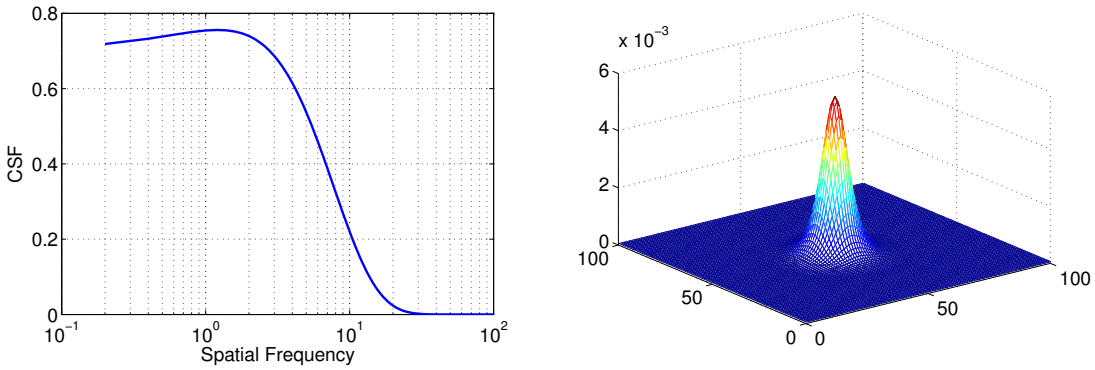
To investigate the effect of neighboring pixels on the randomness calculation, different neighborhood sizes and different sampling methods are tested on simple patterns and the results are shown in Fig. 3.4. Fig. 3.4 (a) shows a regular pattern with a small size and a large size and a random pattern where the pixel values are independently uniform distributed. In Fig. 3.4 (b), the neighboring pixels are dense sampled within a small neighborhood size. We could see that the proposed randomness measure could correctly estimate the randomness of the pattern with small size, but fails for large size. That is because the small size of neighborhood only covers information of limited area. A large size of neighborhood with dense sampling is used in Fig. 3.4 (c), where the randomness is correctly estimated for both small and large size of pattern. While in Fig. 3.4 (d), large neighboring size is used and neighboring pixels are sampled sparsely as shown in Fig. 3.3 (b). We could see that the calculated randomness correctly captures the characteristics of images and achieves similar performance with dense sampling except for some

errors due to the boundary effects. For the random pattern in Fig. 3.3, since its structure is random and neighboring pixels are independent with each others, all estimations give high randomness.

Usually a larger neighborhood could provide better estimation. However the scope of visual attention is limited, the optimal size of neighborhood L in Eq. (3.8) varies according to the pixel density and the viewing distance. Since in this work we assume these parameters are fixed, a constant size of neighborhood is adopted. The randomness estimation on natural images are shown in Fig. 3.5, where the left half of image is more disordered while the right half is more regular and the corresponding calculated randomness with consistent with human perception.

3.2 Masking Modulation with Randomness

After estimating the masking effect with proposed randomness quantitatively, it is critical to investigate the relation of the perceptual distortion and the randomness. Intuitively, the distortion at the pixel with high randomness should be reduced more than with low randomness. However, the exact model of how randomness modulates the actual distortion is not clear. Besides, different coding methods and image content could result in distortion with very different properties. Some distortion may contain more imperceivable distortion and some may contain less. That makes MSE inconsistent among various coding methods. Therefore, to simulate the processing occurred in the initial parts of HVS, proper preprocessing that removes imperceivable distortion is required. In this section, we first preprocess the error with a low-pass filter. Then we investigate the masking modulation at image level and later extend the developed modulation relation to pixel level.



(a) (b)

Figure 3.6: The CSF in frequency and spatial domain (a) Frequency domain. (b) Spatial domain.

3.2.1 Preprocessing with low-pass filtering

The initial visual signal processing in HVS includes two steps. In the first step, the visual signal goes through eye's optics, forming an image on the retina. Because of the diffraction and other imperfections in the eye, such processing would blur the passed image. In the second step, the image will be filtered by neural filter as it is received by photoreceptor cells on retina and then passed on to lateral geniculate nucleus (LGN) and the primary visual cortex. These processes are more like low-pass filtering and will hide parts of signal from perception.

We assume the initial vision processing could be characterized by a linear transfer function and the magnitude of input and output signal in frequency domain is modeled as

$$I_F(\Omega) = G(\Omega) \cdot I(\Omega), \tag{3.9}$$

where $I(\Omega)$ and $I_F(\Omega)$ are the input image and output image in frequency Ω; $G(\Omega)$ is a modulation transfer function (MTF), reflecting the gain of the initial visual processing to various spatial frequencies. $G(\Omega)$ is the concatenation of the two

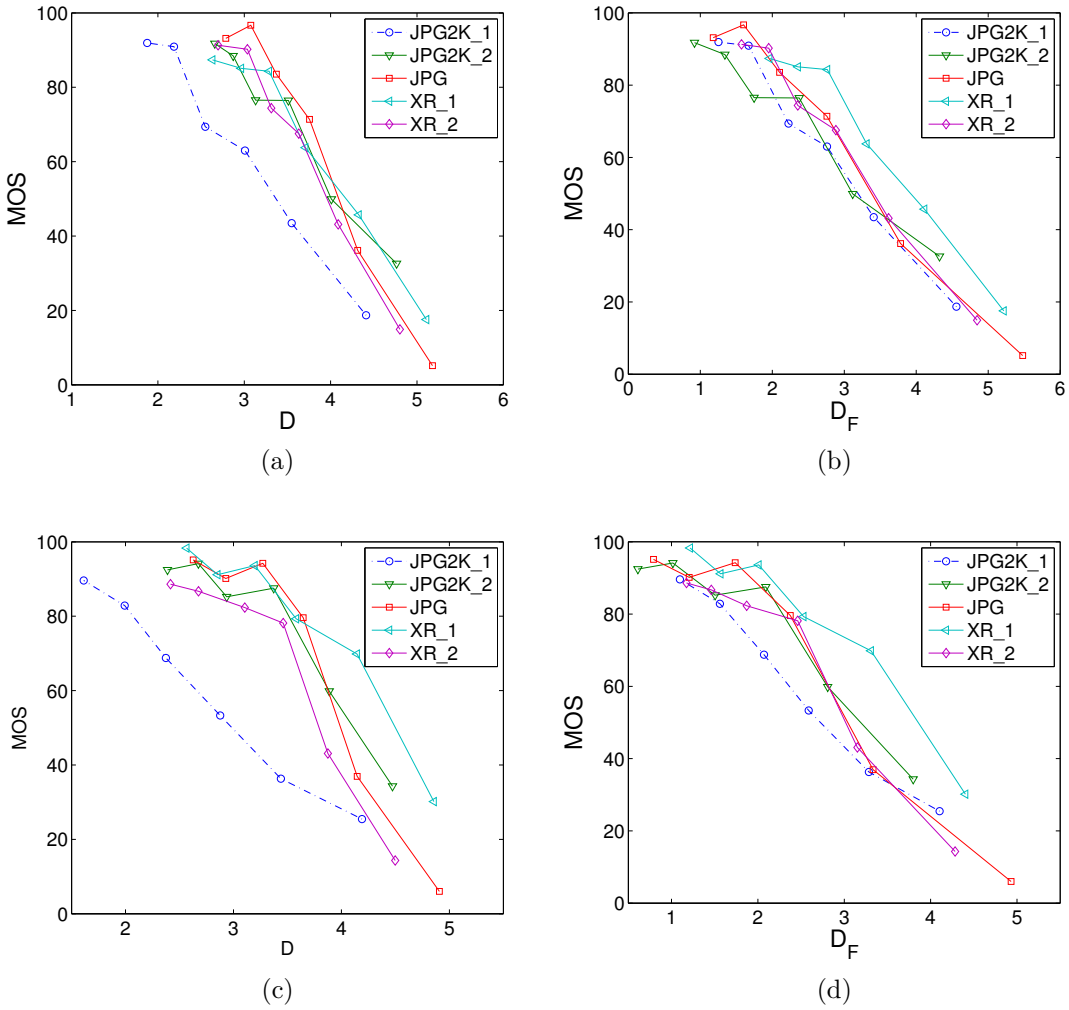


Figure 3.7: The relation of MOS and distortion measurement for different coding methods. The images are coded with different coding methods: including encoding with JPEG2000 using two different setting, denoted as “JPG2K_1" and “JPG2K_2"; with JPEG XR using two different setting denoted as “XR_1" and “XR_2"; and JPEG coding denoted as “JPG". Details are included in [36]. (a) and (c) Without LPF for the image “bike" and “woman", respectively. (b) and (d) With LPF for the image “bike" and “woman", respectively.

MTFs at each step in the initial visual processing. In the first step, the eye's optics could be modeled as a simplified pinhole imaging system and its optical MTF could be expressed as a Gaussian blur function [100]. However the neural

MTF in the second step that occurs in the neural system is hard to measure and model.

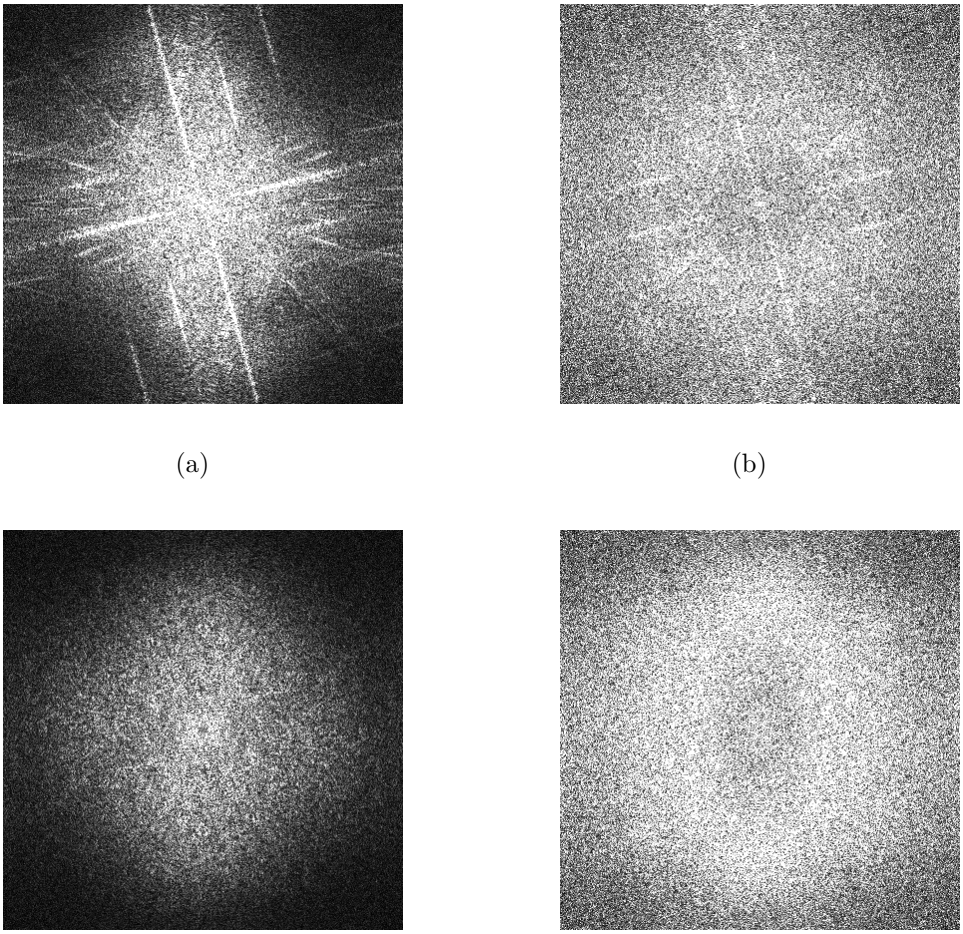
The CSF, which is defined as the inverse of contrast threshold of detectable contrast at a given frequency, provides a comprehensive measure of spatial vision. Although it is not exactly equivalent to MTF, it reflects the same trend as the modulation gain. For instance, a higher sensitivity at particular frequencies always means a higher modulation gain at the corresponding frequencies and *vice versa*. Therefore, many researchers have treated the CSF as the spatial MTF, and used it to define characteristics of initial processing in HVS [31, 13, 127]. In this work, we adopt CSF as the MTF of initial visual processing. There are various CSF models proposed in past [61, 44, 34, 41, 95, 64, 97], and a generalized model is proposed in [95, 97] as

$$G(\Omega) = (a + b\Omega)e^{-c\Omega}, \tag{3.10}$$

where Ω is the spatial frequency and a, b, c are constant model parameters and according to [95], they are set to 0.31, 0.69, and 0.29, respectively. The CSF is a low-pass filter which peaks at a certain frequency and then drops significantly as shown in Fig. 3.6 (a). The CSF indicates that the human eye is less sensitive to higher frequency distortion. Therefore, the perceived distortion could be expressed as

$$\begin{aligned} \Delta\mathbf{I}_F &= \mathbf{g} * \mathbf{I} - \mathbf{g} * \mathbf{I_C} \\ &= \mathbf{g} * \Delta\mathbf{I}, \end{aligned} \tag{3.11}$$

where $\mathbf{I}$ and $\mathbf{I}_C$ are the original and compressed images; the operator $*$ means the convolution; $\Delta\mathbf{I}$ is the actual distortion that $\Delta\mathbf{I} = \mathbf{I} - \mathbf{I}_C$; $\mathbf{g}$ is the spatial low-pass

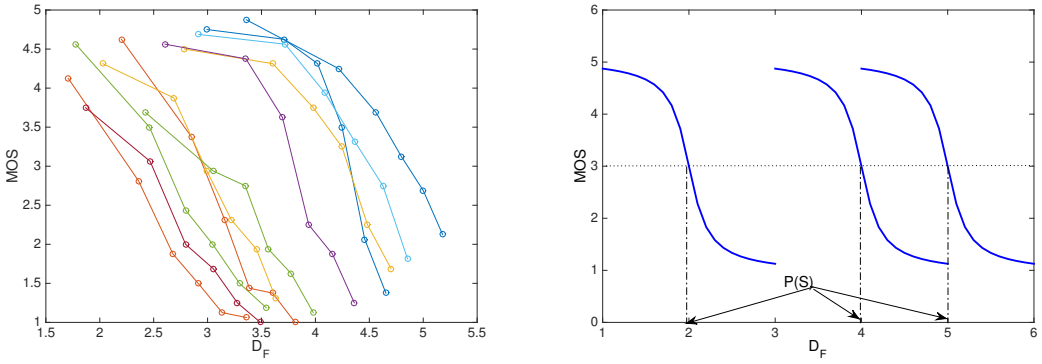


(a) (b)

(c) (d)

Figure 3.8: Frequency magnitude of the distortion ΔI. DC component locates at the center. (a) and (b) show the image “bike" coded with “JPG2K_1" and “JPG2K_2", respectively. (c) and (d) show the image “woman" coded with “JPG2K_1" and “JPG2K_2" , respectively.

filter of the CSF in Eq. (4.1) as shown in Fig. 3.6 (b). $\Delta\mathbf{I}_F$ reflects the observed distortion after initial visual processing. In this way, we could remove the high frequency noise that could not be perceived by humans.



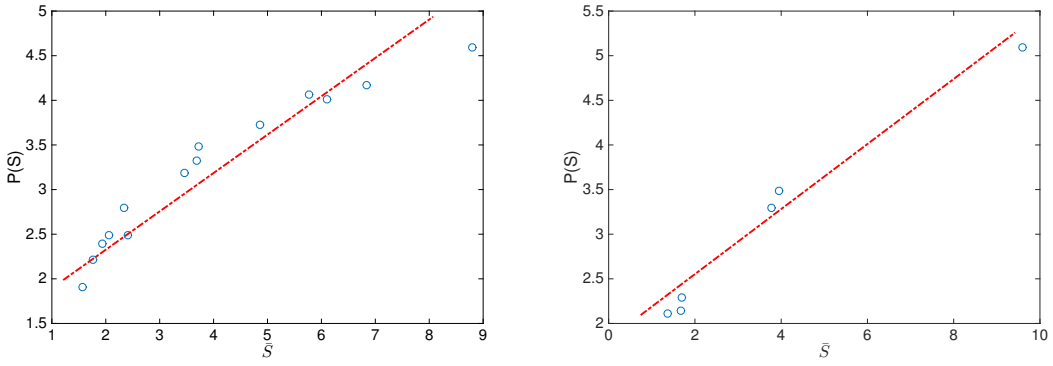
(a) (b)

Figure 3.9: Plot of MOS vs. D_F. Each line corresponds to one original image. (a) Actual plot of MOS vs. D_F from database Toyama. (b) Idealized plot of MOS vs. D_F.

Different encoding methods could yield distinct properties that MSE may not be able to capture. To investigate the effect of low-pass filtering, the distortion measurement before and after low-pass filtering are defined as

$$D = ln\,(\mathrm{MSE})\,, \quad D_F = ln\,(\mathrm{MSE_F}) \tag{3.12}$$

where MSE and MSE_F are the mean squared error without and with low-pass filtering, *i.e.*, mean squared value of $\Delta\mathbf{I}$ and $\Delta\mathbf{I}_F$. Fig. 3.7 (a) and 3.7 (c) show the plots of MOS vs. D, where the images are coded with different coding methods at different quality levels. We could find that given the same D, the images coded with "JPG2K_1" has smaller MOS than with other coding methods, which means the distortion from "JPG2K_1" is more obvious. This is because as shown in Fig. 3.8, for "JPG2K_1", the most distortion energy locates on low frequencies while for "JPG2K_2" the distortion energy spreads out to higher frequencies at which humans are less sensitive. After low-pass filtering, the most parts of imperceivable



(a) (b)

Figure 3.10: The linear relationship between mean randomness $\bar{S}$ and horizontal displacement $P(S)$. (a) On Toyama. (b) On MMSPG.

distortion are removed, and hence D_F becomes more consistent among different coding methods as shown in Fig. 3.7 (b) and Fig. 3.7 (d).

3.2.2 Imagewise masking modulation

To investigate how the masking effect reduces the visibility of distortion at image level, The relationship between D_F and MOS is shown in Fig. 3.9 (a) for various images compressed at different quality levels. Each circle represents a coded image and the circles connected by the same lines share the same original images. In other words, the connected circles in Fig. 3.9 (a) are the images compressed from the same original images but with different compression levels, hence they are affected by the same masking effect.

As we could see in Fig. 3.9 (a), for the image set sharing a particular original image, their MOS values monotonically decrease with D_F and each image set has similar MOS-D_F relation but with different horizontal displacement. The mean MOS and mean D_F of each set is calculated and summarized in Table 3.1, where

Table 3.1: Average MOS and average D_F of each image

Image	MOS	D_F	Image	MOS	D_F
Kp01	3.0	2.44	Kp13	2.8	2.91
Kp03	2.6	0.85	Kp16	2.7	1.32
Kp05	3.0	2.55	Kp20	3.2	0.95
Kp06	3.0	1.93	Kp21	2.3	1.74
Kp07	3.0	1.13	Kp22	2.6	1.72
Kp08	2.8	2.62	Kp23	3.0	0.58
Kp12	2.6	0.99	Kp24	2.8	2.22

we could see the average perceptual quality of coded image is around at 3.0 in MOS, however the mean D_F is quite different from each other.

Such difference in horizontal displacement comes from the different masking effect of different images. Given the same MOS, the lines of the images on the right side have more distortion than the lines on left as shown in Fig. 3.9 (a), which means the image on the right side has more masking which makes it appear the same quality as the images on the left side. Therefore, the image sets with strong masking effect are more likely to have curves on the right side, and the relative displacement of these curves to the left reflects the significance of masking effect.

To investigate these horizontal displacement of these curves, the small difference in the shapes of curves is neglected by idealizing the curves as in Fig. 3.9 (b). Consequently the MOS-D_F relation could be expressed as

$$\widehat{\mathrm{MOS}} = F(D_F - P(S)), \tag{3.13}$$

where $\widehat{\mathrm{MOS}}$ is the predicted MOS; $F(\cdot)$ is a nonlinear monotonic decreasing function representing the shape of these curves and $P(S)$ is the displacement of the

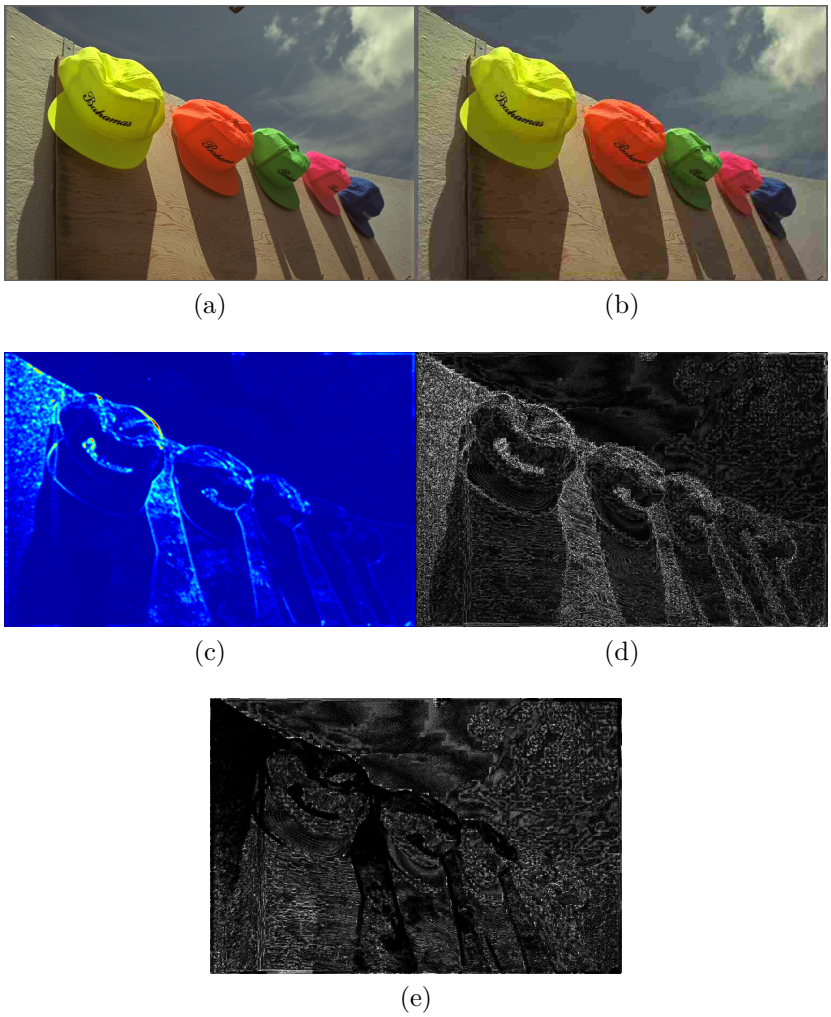


Figure 3.11: Distortion modulated at pixel level. (a) Original image (b) Distorted image. (c) Heat map of randomness. (d) Distortion before modulation. (e) distortion after modulation (properly scaled for better illustration).

curves, which is a function of randomness S of the corresponding images, since S reflects the significance of masking effect.

The actual horizontal displacement of the curves could be measured by the intersection of the curves and any horizontal lines such as MOS = 3.0 as shown in Fig. 3.9 (b). Using other lines will result in a constant adding to $P(S)$, but it will not affect the following equations. To investigate the relation between $P(S)$ and

randomness S, the image level randomness is calculated by averaging pixel level randomness as

$$\bar{S} = \frac{1}{WH} \sum_{i=1}^{W} \sum_{j=1}^{H} S(i,j) \tag{3.14}$$

and the plot of $P(S)$ vs. $\bar{S}$ is shown in Fig. 3.10. In Fig. 3.10 (a), for the Toyama database we could observe that $P(S)$ increase linearly with $\bar{S}$. The same observation could be obtained in Fig. 3.10 (b) for the MMSPG database. Therefore their relationship could be expressed as

$$P(S) = \lambda \bar{S} + b, \tag{3.15}$$

where λ and b are model parameters. Then by substituting Eq. (3.12) and Eq. (3.15) into Eq. (4.12), we could have

$$\begin{aligned} \widehat{\mathrm{MOS}} &= F\left(ln(\mathrm{MSE_F} \cdot e^{-\lambda \bar{S}}) - b\right) \\ &= G\left(\mathrm{MSE_F} \cdot e^{-\lambda \bar{S}}\right) \end{aligned} \tag{3.16}$$

where $G(\cdot) \equiv F(ln(\cdot) - b)$ is a nonlinear mapping. It is acceptable for a IQM to predict MOS through a nonlinear mapping, because the mapping is easy to be found and it depends on various environmental factors like the range of MOS and evaluation methodology. Therefore, in [1] and [2], a nonlinear mapping is not considered as part of IQM, rather it is left to the final stage of performance evaluation. $G(\cdot)$ could be obtained by fitting the objective prediction scores to the subjective quality scores as described in [1, 2].

From Eq. (3.16), we can conclude that Image-wise Perceptually Weighted MSE (IPW-MSE) is a good indicator of MOS, which is calculated as

$$\text{IPW-MSE} = \text{MSE}_\text{F} \cdot e^{-\lambda \bar{S}} \tag{3.17}$$

Without considering the masking effect, MSE_F is not accurate enough to indicate the perceptual quality as we have observed in Fig. 3.9. Eq. (3.17) gives the exact relation how MSE_F should be modified with randomness S. It is also consistent with our intuition that the increase of image level randomness $\bar{S}$ will reduce the visibility of distortion MSE_F.

3.2.3 Pixelwise masking modulation

In the above section, we discuss the same distortion (*i.e.*, MSE_F) does not mean equal perceptual quality in different images due to the masking effect. Rather it should be modulated with randomness as in Eq. (3.17). Even within the image, the distortion is not equally perceived because of the various masking effect in different image regions. To obtain the precise IQM, we consider the masking effect at a finer level, *i.e.*, pixel level. Since the subjective test can be hardly conducted at pixel level, we assume that the obtained modulation relationship at image level in Eq. (3.17) is also applied to pixels. It is validated by the performance improvement in the experiments of Section 3.3. In Eq. (3.17), by replacing MSE_F and mean randomness ($\bar{S}$) with filtered squared error $\Delta I_F(i,j)^2$ and randomness $S(i,j)$ of each pixel measured in Eq. (3.6), we have modulated the squared error at each pixel as

$$SE_M(i,j) = \Delta I_F(i,j)^2 \cdot e^{-\lambda_2 \cdot k \cdot |y(i,j) - R_{YX} \tilde{R}_X^{-1} \mathbf{x}(i,j)|} \tag{3.18}$$

where λ_2 is a constant model parameter and k is related to image resolution, i.e. $k = 1$ if $W \times H > 768 \times 511$ and $k = 0.083$ elsewise. In this way, the normalized distortion at each pixel has equal perceptual effect.

Fig. 3.11 (a) and (b) show a original image and the compressed image. Fig. 3.11 (d) shows the filtered distortion, where we can see that even though the actual distortion in the sky area is much small compared to that in other parts, the perceived distortion is still comparable to other parts. This is because the sky area is smoother than other areas, and thus the masking effect is much weaker than other parts. That could be reflected by the corresponding randomness map as shown in Fig.3.11 (c). After modulating the actual distortion with the randomness map, we can see the distortion in the sky area is enhanced relatively. This is consistent with perceptual observation.

Since the modulated distortion is perceptually normalized, the perceptually weighted MSE (PW-MSE) is calculated by even pooling as

$$\text{PW-MSE} = ln\left(\frac{1}{HW}\sum_{i=1}^{H}\sum_{j=1}^{W} SE_M(i,j)\right). \tag{3.19}$$

Similarly MOS could be predicted with PW-MSE through a proper nonlinear mapping.

3.3 Experimental Results

To evaluate the performance of PW-MSE, six databases with various types of compression distortion are used, including Toyama [6], MMSPG [36], TID2008 [104], TID2013 [103] and CSIQ [67]. In the Toyama database, there are 14 original images with solution of 768×512. Each original image is encoded with JPEG [53] and JPEG2000 [115] at six different quality levels, generating 168 distorted

images. In the MMSPG database, there are 6 original images with the solution of 1280×1600. Three different codecs JPEG, JPEG 2000 and JPEG XR are used in the database. For JPEG 2000 and JPEG XR two different coding strategies are adopted, which are denoted as “JPG2K_1" and “JPG2K_2", “XR_1" and “XR_2", respectively. For each coding method, original images are coded at 6 different quality levels. Therefore, there are totally 160 distorted images. There are a broad spectrum of distortion types in the TID2008, TID2013 and CSIQ databases. Since we are only intereted in compression distortion, only JPEG and JPEG 2000 distortion are investigated on these databases.

As for metrics of performance evaluation, the Pearson linear correlation coefficient (PLCC), Spearman rank order correlation coefficient (SROCC) and root mean squared error (RMSE) are employed as described in [1, 2]. PLCC generally indicates the goodness of linear relation. SROCC is computed on ranks and thus depicts the monotonic relationships. RMSE computes the prediction errors and thus depicts the prediction accuracy. To put the MOS and its prediction on the same scale for various algorithms, a monotonic logistic function is used to find nonlinear mapping between the prediction and subjective quality scores as [2]:

$$q(x) = \alpha_1 \left(0.5 - \frac{1}{1 + exp(\alpha_2(x - \alpha_3))}\right) + \alpha_4 x + \alpha_5, \tag{3.20}$$

where α_1 to α_5 are the parameters obtained by regression between the input and output data.

Table 3.2: Performance evaluation at each step

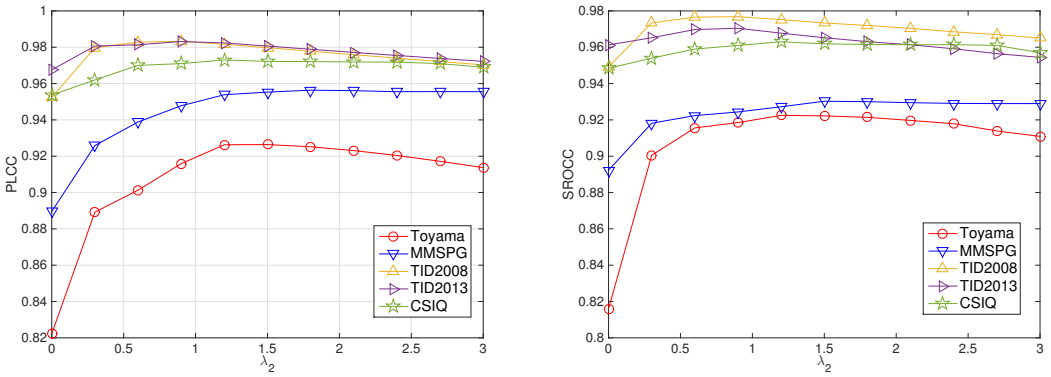
	PLCC				SROCC				RMSE			
	D	D_F	IPW-MSE	PW-MSE	D	D_F	IPW-MSE	PW-MSE	D	D_F	IPW-MSE	PW-MSE
Toyama	0.626	0.822	0.872	**0.926**	0.613	0.816	0.873	**0.922**	0.976	0.712	0.612	**0.470**
MMSPG	0.775	0.890	0.921	**0.954**	0.797	0.891	0.866	**0.927**	16.769	12.139	10.358	**7.965**
TID2008	0.870	0.952	0.961	**0.983**	0.866	0.949	0.963	**0.977**	0.933	0.577	0.478	**0.343**
TID2013	0.899	0.967	0.972	**0.983**	0.917	0.916	0.956	**0.970**	2.199	0.414	0.389	**0.300**
CSIQ	0.861	0.954	0.970	**0.973**	0.916	0.948	0.956	**0.963**	0.158	0.094	0.079	**0.072**

3.3.1 Validation at each stage

The proposed algorithm consists of several steps to simulate the different stages of HVS. To evaluate the effectiveness of the proposed IQM at each step, intermediate results are summarized in Table 3.2 for all six databases.

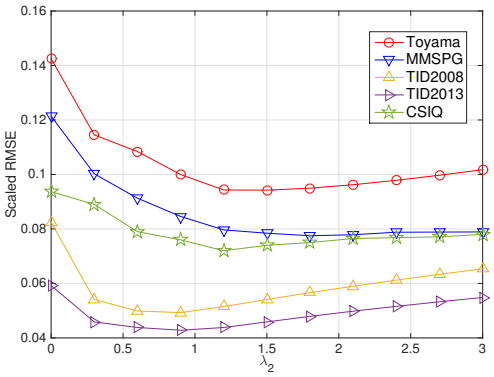
We evaluate the performance of D_F in Eq. (3.12) after applying low-pass filter. Then frame level masking effect is considered and the performance of IPW-MSE is measured, and finally the performance of PW-MSE is measured. As shown in Table 3.2, the performance on compression distortion of all databases are presented, where we can see, as the starting point, MSE has the worst performance comparing to other steps of the proposed algorithm. This is expected because MSE does not incorporate any characteristics of HVS. Then from D_F to PW-MSE, the performance on the overall database is improved from 0.822 to 0.926 in PLCC for the Toyama database and from 0.890 to 0.954 in PLCC for the MMPSG database. Similarly, we can observe the similar trend on other databases and in other performance metrics, *i.e.*, SROCC and RMSE.

The performance of D_F is significant improved from MSE. This is because with the low-pass filtering, D_F removes the most parts of imperceivable distortion, making it more consistent with the human perception. IPW-MSE and PW-MSE improves the performance further, because in addition to low-pass filtering, the masking effect is considered. Moreover, we could find that the performance of



(a)

(b)



(c)

Figure 3.12: The effect of model parameter λ_2 on various performances. (a) PLCC (b) SROCC (c) RMSE

PW-MSE is generally better than IPW-MSE either under each type of distortions or under the overall database. This is because in PW-MSE , the masking effect is considered at a finer scale than in IPW-MSE, as a consequence, the predication is more accurate.

3.3.2 Parameter investigation

Parameters are critical to the performance of the proposed algorithm. λ_2 in Eq. (3.18) is an important parameter that would affect the overall performance. To

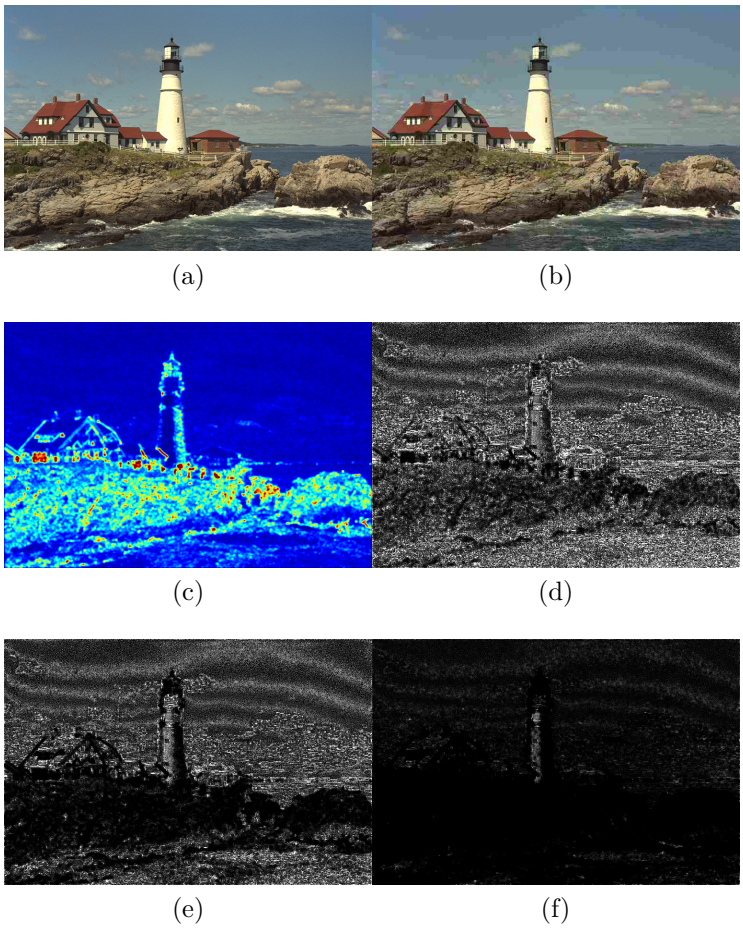


Figure 3.13: Visual illustration of distortion modulation at pixel. (a) Original image. (b) Distorted image. (c) Randomness map. (d) Distortion modulated with $\lambda_2 = 0.2$. (d) Distortion modulation with $\lambda_2 = 1.2$. (e) Distortion modulation with $\lambda_2 = 2.2$.

investigate its influence on the final performance, experiments are carried out by varying it in the range of [0, 3].

The curves of the overall performance on the six databases are shown in Fig. 3.12 for PLCC, SROCC and RMSE, respectively. When $\lambda_2 = 0$, the masking modulation with randomness is actually eliminated, resulting in the same performance as D_F. As shown in Fig. 3.12 (a), when λ_2 increases slightly, the performance

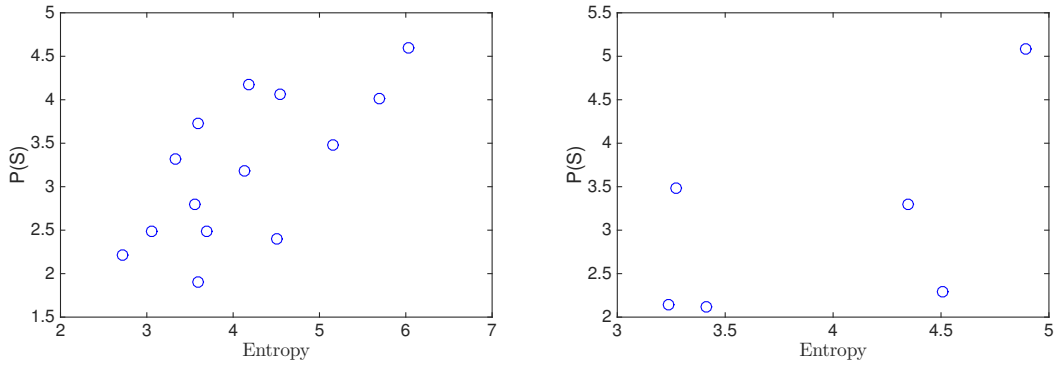
increases significantly on all the databases. During this stage, the masking modulation starts affecting and the parts masked by strong maskers reduce its impacts on the overall quality index. When λ_2 becomes larger, after peaking at a certain value, the performance starts decreasing. This is because some distortion is over-masked and thus it is not consistent with the HVS. The same observation could be obtained in SROCC and RMSE in Fig. 3.12 (b) and (c). As for the best λ_2, it is almost constant on each database that it generally falls in the range [1, 2]. In the proposed algorithm, it is fixed at 1.2.

Fig. 3.13 visually illustrates the masked distortion with different parameters. We can see that in the distorted images in Fig. 3.13 (b), the distortion is more obvious in the sky region where the content is simple, while less obvious in the rock region. If the parameter λ_2 is too small as in Fig. 3.13 (d), the distortion in the complex region is not masked enough. Thus the measured quality index is not accurate enough. When λ_2 is too large as in Fig. 3.13 (f), the distortion in the complex region is over masked that it totally disappears, which is also inaccurate.

3.3.3 Validation of effectiveness of randomnness map

To further verify the effectiveness of proposed randomness, a entropy map and a masking map generated from division gain normalization [81] are used to replace randomness map in the proposed metric and their performance are compared. The entropy map is calculated based on 9×9 blocks, pixels within each non-overlap 9×9 block share the same entropy value.

The linear relation in Eq. (3.15) is critical to the accuracy of proposed quality metric. We can see that the randomness could generally achieve a good linear relation as shown in Fig. 3.10. The relation between entropy map and displacement



(a) (b)

Figure 3.14: Relation between displacement of metric curves and entropy map. (a) On the Toyama database. (b) On the MMSPG database.

of metric curves are visually shown in Fig. 3.14, where we can see that there is neither strong linear relation nor other proper relation.

The performance of the proposed metric with different masking maps is evaluated on various databases. The results are shown in Table 3.3 and it is obvious that the proposed metric with randomness map has better performance. This is because randomness has better prediction for the displacement of metric curves and the performance of the proposed metric significantly relies on such relation, otherwise the metric could not effectively estimate the masking effect.

3.3.4 Comparison with benchmark algorithms

In this section, the performance of PW-MSE is compared with that of the seven benchmarks including: PSNR, SSIM [124], MS-SSIM [126], VIFp [112], GSMD [136], FSIM [140] and VSI [139]. Default setting is used for all the benchmark IQMs. FSIM and VSI are computed in color space and the rest IQMs are computed in gray images, where color images in RGB space are converted into YCbCr color space and only the luminance component Y is used. In TID2008, TID2013, and

Table 3.3: Overall performance on different databases

	Database	PSNR	SSIM	MS-SSIM	VIFp	GSMD	FSIM	VSI	EntropyDGN		Our
PLCC	Toyama	0.588	0.849	0.852	0.779	0.825	0.863	0.858	0.822	0.832	**0.926**
	MMSPG	0.790	0.927	0.936	0.853	0.898	0.899	0.926	0.849	0.886	**0.954**
	TID2008	0.869	0.963	0.974	0.953	0.982	0.975	0.981	0.935	0.951	**0.983**
	TID2013	0.916	0.962	0.970	0.952	0.975	0.971	0.981	0.938	0.967	**0.983**
	CSIQ	0.918	0.967	**0.981**	0.978	0.977	0.979	0.976	0.954	0.955	0.973
SROCC	Toyama	0.578	0.841	0.848	0.778	0.850	0.856	0.855	0.816	0.817	**0.922**
	MMSPG	0.797	0.904	0.897	0.820	0.914	0.892	0.900	0.760	0.892	**0.927**
	TID2008	0.866	0.961	0.969	0.949	**0.979**	0.969	0.978	0.929	0.949	0.977
	TID2013	0.917	0.948	0.955	0.938	0.968	0.958	0.968	0.931	0.961	**0.970**
	CSIQ	0.916	0.951	**0.968**	0.967	0.963	0.964	0.967	0.948	0.949	0.963
RMSE	Toyama	1.012	0.661	0.656	0.785	0.708	0.633	0.642	0.712	0.710	**0.472**
	MMSPG	16.277	9.925	9.345	13.851	11.656	11.611	10.014	15.563	12.288	**7.965**
	TID2008	0.937	0.510	0.431	0.571	0.354	0.424	0.372	0.777	0.583	**0.343**
	TID2013	0.658	0.445	0.397	0.502	0.366	0.393	0.318	0.731	0.418	**0.300**
	CSIQ	0.123	0.080	**0.060**	0.065	0.066	0.063	0.068	0.094	0.096	0.072

Table 3.4: Results of statistical significance test

	PSNR	SSIM	MS-SSIM	VIFp	GSMD	FSIM	VSI	PW-MSE
PSNR	–	11111	11111	11111	11111	11111	11111	11111
SSIM	00000	–	00111	00001	00111	11111	00111	11111
MS-SSIM	00000	00000	–	00000	00100	11000	00110	11110
VIFp	00000	11010	11110	–	01110	11110	11110	11110
GSMD	00000	11000	11000	10000	–	11000	11010	11010
FSIM	00000	00000	01000	00000	00100	–	00110	11110
VSI	00000	00000	00001	00000	00000	10000	–	11000
PW-MSE	00000	00000	00001	00000	00000	00000	00000	–

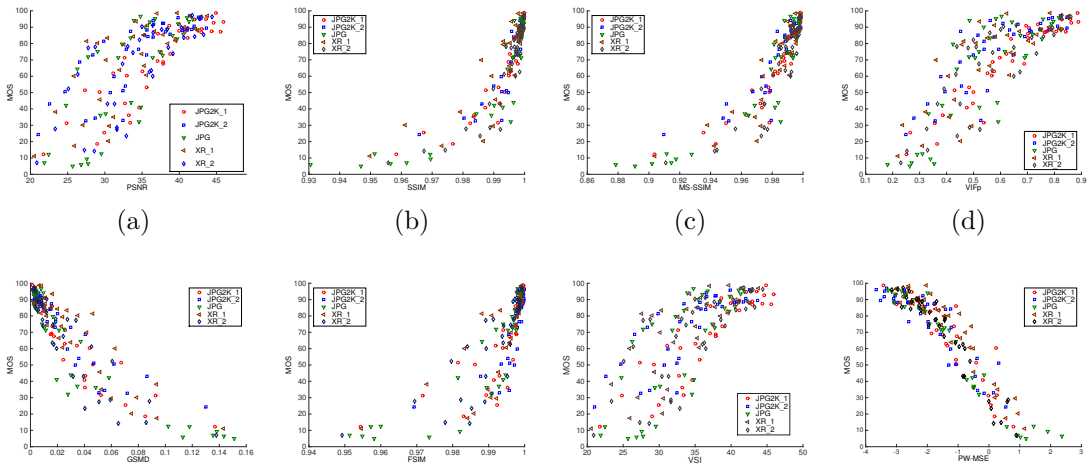
CSIQ databases, only the images with compression distortion, *i.e.,* JPEG and JPEG 2000 distortion are used for evaluation.

Generally PLCC, SROCC and RMSE are consistent in performance evaluation, but not always. For example, in Table 3.3, PW-MSE achieve the best performance on TID2008 in terms of PLCC, but not the best in terms of SROCC. That is

because these evaluation methods measure different aspects of performance, and they are not exactly the same.

For the overall performance, from Table 3.3, we can see that PSNR has the worst performance in PLCC among all IQMs. This is reasonable, because all the other IQMs incorporates with the characteristics of HVS while PSNR merely computes the pixel errors. We can have the similar observation in other performance metrics, *i.e.,* SROCC and RMSE. SSIM and MS-SSIM have similar performances on both databases, this is because both of them measure the structure distortion. In PLCC, PW-MSE outperforms other seven benchmarks, except on the CSIQ database, where it also achieves close performance to the best performer MS-SSIM. In general, PW-MSE has excellent performance comparing with other benchmarks under various evaluation methods.

To obtain statistical conclusions on the performance of PW-MSE, we followed similar approaches of hypothesis testing in [136, 113]. The hypothesis tests are carried out on the MOS prediction residual of two quality metrics, which is assumed to follow Gaussian distribution. The left-tailed F-test to the residuals of every two metrics on different databases and the results are shown in Table 3.4. A test result of $H = 1$ for the left-tailed F-test at a significance level of 0.05 means that the metric in the column has better performance than the model in rows with a confidence greater than 95%. A value of $H = 0$ means the metric in the column has indistinguishable or significant worse performance than the metrics in rows. Each cell of Table 3.4 contains 5 flags, which from left to right stand for the test results on the Toyama, the MMSPG, the TID2008, the TID2013, and the CSIQ databases, respectively. We can see that PW-MSE has the most positive flags, *i.e.,* 1, indicating it has significant better performance than other metrics on most databases.



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 3.15: Scatter plot of MOS vs. IQMs. (a) PSNR (b) SSIM (c) MS-SSIM (d) VIFp (e) GSMD (f) FSIM (g) VSI (h) PW-MSE

Table 3.5: Compression distortion and its visual artifacts

Compression distortion	Visual distortion
JPEG	Blocking, Ringing
JPEG 2000	Blurriness, Ringing
JPEG XR	Blocking, Blurriness, Ringing

To provide a visual comparison among the benchmark IQMs and the proposed algorithm, the scatter plots of the quality index versus the MOS are shown in Fig. 3.15, where each point corresponds to a distorted image. We could see that for SSIM, MS-SSIM, GSMD and FSIM, the quality scores of the good quality images are very close to each other. For example, in SSIM, for the images with quality higher than 50 in MOS, its SSIM scores are in the range of 0.99 to 1.00. For PW-MSE, quality scores are evenly distributed.

Table 3.6: Performance on JPEG2000 distortion

		PSNR	SSIM	MS-SSIM	VIFp	GSMD	FSIM	VSI	Proposed
PLCC	Toyama	0.856	0.853	0.858	0.833	0.865	0.839	0.912	**0.926**
	MMSPG	0.834	0.938	0.883	0.876	0.905	**0.975**	0.948	0.902
	TID2008	0.867	0.968	0.976	0.965	0.986	0.98	0.986	**0.987**
	TID2013	0.917	0.967	0.971	0.961	0.979	0.973	**0.983**	0.974
	CSIQ	0.947	0.963	**0.982**	0.978	0.980	0.981	0.975	0.978
SROCC	Toyama	0.865	0.845	0.848	0.83	0.892	0.826	0.908	**0.939**
	MMSPG	0.826	0.937	0.926	0.864	0.940	**0.956**	0.933	0.915
	TID2008	0.813	0.964	0.970	0.958	0.981	0.977	**0.985**	0.980
	TID2013	0.884	0.949	0.954	0.941	0.967	0.958	**0.971**	**0.971**
	CSIQ	0.936	0.956	**0.973**	0.97	0.972	0.969	0.969	0.970
RMSE	Toyama	0.652	0.66	0.648	0.699	0.633	0.687	0.517	**0.463**
	MMSPG	12.768	7.999	10.87	11.177	9.829	**5.093**	7.353	9.995
	TID2008	0.972	0.492	0.428	0.514	0.327	0.387	0.320	**0.312**
	TID2013	0.679	0.435	0.407	0.473	0.351	0.392	**0.312**	0.385
	CSIQ	0.102	0.085	**0.060**	0.066	0.063	0.062	0.071	0.066

3.3.5 Performance on individual distortion types

The compression distortion consists of various visual distortion types, *e.g.,* blurriness, blocking and ringing artifacts. As pointed out in [76, 86, 99], different compression distortion types may be dominated by very different visual distortion types. For example, JPEG distortion mainly include blocking and ringing artifacts, while JPEG 2000 distortion include blurriness and ringing artifacts. Table 3.5 summarizes the compression distortion and their main visual distortion types.

To have a comprehensive understanding of the performance of the proposed metric on individual type of distortion, especially on the distortion types that are visually different, we compare the performance with benchmark metrics on JPEG 2000, JPEG and JPEG XR, respectively and the results are listed in Table 3.6, 3.7, and 3.8, respectively. We can see that for JPEG 2000, PW-MSE hits the top 8 times, which is better than other quality metrics. Similarly for JPEG and JPEG

Table 3.7: Performance on JPEG distortion

		PSNR	SSIM	MS-SSIM	VIFp	GSMD	FSIM	VSI	Proposed
PLCC	Toyama	0.391	0.849	0.849	0.736	0.786	0.892	0.809	**0.954**
	TID2008	0.868	0.957	0.97	0.939	0.977	0.974	**0.986**	0.969
	TID2013	0.914	0.957	0.968	0.941	0.97	0.971	**0.985**	0.981
	CSIQ	0.847	0.976	**0.984**	0.982	0.984	0.984	0.981	0.971
SROCC	Toyama	0.332	0.844	0.853	0.730	0.814	0.899	0.809	**0.951**
	MMSPG	0.764	0.882	0.870	0.769	0.916	0.905	0.914	**0.944**
	TID2008	0.876	0.930	0.941	0.916	0.953	0.937	**0.962**	0.956
	TID2013	0.919	0.922	0.933	0.916	0.951	0.938	0.954	**0.959**
	CSIQ	0.888	0.953	0.966	**0.967**	0.965	0.965	0.962	0.955
RMSE	Toyama	1.138	0.654	0.653	0.838	0.764	0.558	0.728	**0.370**
	MMSPG	19.991	10.364	10.817	14.766	13.012	8.543	9.092	**4.678**
	TID2008	0.847	0.495	0.416	0.587	0.361	0.384	**0.284**	0.420
	TID2013	0.611	0.437	0.375	0.508	0.366	0.358	**0.256**	0.295
	CSIQ	0.163	0.066	**0.055**	0.057	0.055	0.055	0.060	0.073

XR, PW-MSE also has the best performance in terms of being the best metric on a specific database.

Besides, we also compare the performance on other non-compression distortions such as Gaussian blur and white additive noise. The results are shown in Table 3.9 and 3.10, respectively and the top 3 performers are highlighted in bold font. As we can see, the proposed metric still has the comparable performance with other benchmark metrics.

Table 3.8: Performance on JPEG XR distortion

		PSNR	SSIM	MS-SSIM	VIFp	GSMD	FSIM	VSI	Proposed
PLCC	MMSPG	0.783	0.915	0.927	0.829	0.883	0.933	0.901	**0.956**
SROCC	MMSPG	0.775	0.878	0.883	0.806	0.885	0.908	0.88	**0.928**
RMSE	MMSPG	16.212	10.503	9.769	14.552	12.227	9.394	11.314	**7.618**

Table 3.9: Performance on Gaussian blur

		PSNR	SSIM	MS-SSIM	VIFp	GSMD	FSIM	VSI	Proposed
PLCC	TID2008	**0.934**	0.818	0.821	0.781	0.885	0.783	**0.924**	**0.914**
	TID2013	**0.952**	0.88	0.882	0.859	0.911	0.904	**0.952**	**0.937**
	CSIQ	0.952	0.953	0.954	**0.957**	**0.968**	0.929	**0.964**	0.951
SROCC	TID2008	**0.908**	0.827	0.830	0.805	**0.923**	0.857	**0.924**	0.910
	TID2013	**0.929**	0.878	0.879	0.855	**0.949**	0.898	**0.946**	0.925
	CSIQ	0.936	0.953	0.954	0.957	**0.969**	0.926	**0.964**	0.947
RMSE	TID2008	**0.219**	0.351	0.349	0.381	0.285	0.540	**0.234**	**0.248**
	TID2013	**0.217**	0.337	0.334	0.363	0.293	0.304	**0.218**	**0.247**
	CSIQ	0.051	0.051	0.05	**0.048**	**0.042**	0.062	**0.045**	0.052

Table 3.10: Performance on white noise

		PSNR	SSIM	MS-SSIM	VIFp	GSMD	FSIM	VSI	Proposed
PLCC	TID2008	0.872	**0.947**	**0.951**	0.943	0.887	0.945	0.946	**0.947**
	TID2013	0.895	0.880	**0.964**	**0.962**	0.892	0.955	**0.956**	0.948
	CSIQ	0.908	0.939	0.866	**0.957**	0.969	**0.957**	0.876	**0.958**
SROCC	TID2008	0.879	0.879	**0.955**	0.943	0.901	0.901	**0.953**	**0.947**
	TID2013	0.915	0.915	**0.968**	**0.964**	0.915	0.915	**0.961**	0.953
	CSIQ	0.929	0.929	**0.975**	0.967	**0.971**	**0.971**	0.968	0.968
RMSE	TID2008	0.575	**0.378**	**0.361**	0.389	0.541	0.382	0.381	**0.372**
	TID2013	0.556	0.592	**0.33**	**0.342**	0.565	0.370	**0.365**	0.397
	CSIQ	0.120	0.098	0.143	**0.083**	**0.071**	0.083	0.304	**0.082**

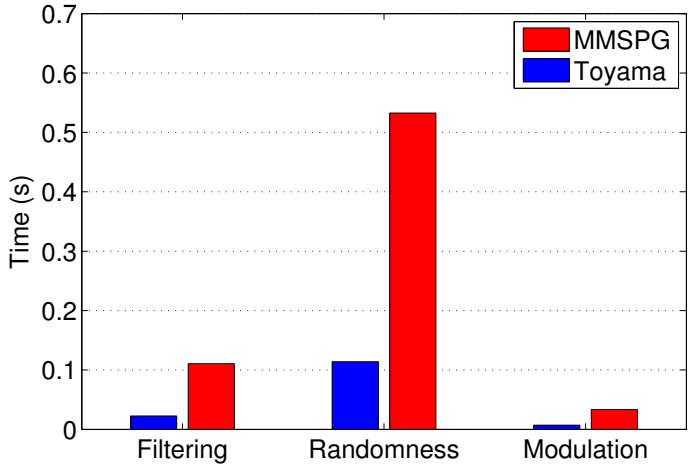


Figure 3.16: Average consumed time in each stage of the PW-MSE

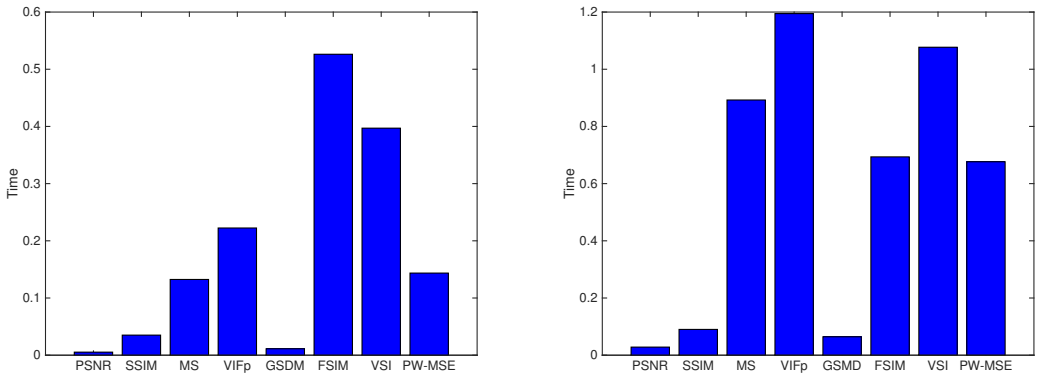
3.3.6 Computational complexity

The computational complexity of the proposed PW-MSE is also analyzed in this section. Since PW-MSE consists of three stages: namely they are low-pass filtering, randomness calculation and modulation, their time consumption is investigated respectively. The average processing time over all images of each database was measured for each stage. The results are illustrated in Fig. 3.16, where we can see that, because of the larger image resolution, the time consumption on the MMSPG database is higher than on the Toyama database. Moreover, on both databases, we can find that the randomness calculation takes a large portion of computation in the proposed algorithm.

Meanwhile, we also compared the total time consumption of PW-MSE with other benchmark algorithms. The mean of consumed time for each image was measured and the results on both databases are shown in Fig. 3.17. Among these IQMs, since PSNR is the simplest in computation complexity, it has the least computing time as expected. Because SSIM and GSMD calculate the similarity of pixel and edge information respectively, their time consumption is slightly larger than PSNR and less than other algorithms. For PW-MSE, since the randomness is computed for the entire image, it increases the computational complexity, but it still has less or comparable time consumption comparing with the rest IQMs.

3.4 Summary

In this chapter, PW-MSE is proposed for compressed images. The masking effect as well as the low-passing filter characteristics of the initial process of HVS is explored. To mathematically model and simulate the initial process in HVS , the CSF is adopted as the transfer function in frequency domain. The error signal from



(a) (b)

Figure 3.17: Average total consumed time of the benchmark algorithms and the PW-MSE. (a) On the Toyama database (b) On the MMSPG database.

the compression distortion is filtered with the proposed transfer function in spatial domain, which removed most errors in high frequency that can not be perceived by humans. Furthermore, after processing through the initial part of HVS, the error signal is highly affected by various masking effects from different image contents. To study the masking effect quantitatively, the randomness is proposed to measure it by considering the spatial correlations. Moreover, a modulation relation among the randomness and the distortion before masking and after masking is investigated. By observing the relation of MOS and the distortion before masking effect, a modulation model is proposed at image level. Later, it is extended into pixel level, providing finer scale masking analysis. PW-MSE is tested on the databases with various compression distortions. By validating at every step, we could found that each step of PW-MSE contributes to overall performance improvement. The performance comparison with other benchmark IQMs demonstrates the effectiveness of PW-MSE.

Chapter 4

Quality Assessment for Compressed Videos

One of the most important applications of video quality metric is assisting the video codec to select proper coding parameters and thus achieve the optimal rate-distortion balance. Since the assessment to the compressed video is critical to the coding performance, it is highly desired to develop video quality metrics that could precisely predict human's perception on compressed videos. Due to the large time and human resource consumption of the subjective video quality assessment, great efforts have been dedicated to developing various objective video quality metrics.

A number of video quality metrics have been designed to simulate the characteristics of the HVS. Contrast sensitivity is one of the most important properties of the HVS, which varies to different spatial and temporal frequencies, and has been psychophysically studied and modeled in the contrast sensitivity function (CSF) [44, 41, 61, 34, 23, 55, 131]. Video quality metrics employ the CSF to analyze the visibility of impairs [87, 70]. In [87], the video is preprocessed with separable filters both in temporal and spatial domains. A low-pass and a bandpass filter are used for temporal filtering, while spatial filtering is implemented in the Discrete Wavelet Transform (DWT) domain. In [70], distortion is decoupled into detail losses and additive impairments with DWT, and the sensitivity of the distortion is analyzed through a comprehensive spatial-temporal CSF and the weighting factors are calculated to adjust the distortion according to the sensitivity at different

DWT frequencies. In these CSF models, the contrast sensitivity is only modeled as a function of frequency, without taking the visual attention into consideration.

Actually, the contrast sensitivity is not uniform distributed over the video content. Instead, it peaks at gazed region and decreases away from it. While static images might give viewers enough time to watch the details in different regions, videos release tremendous information within very short time, that makes the HVS unable to receive all of it. Consequently, only the parts within visual attention are perceived throughout while the other parts may be ignored. Therefore, visual attention plays an important role in quality assessment and it starts being concerned in recent researches [123, 68, 74, 39]. In [123], the difference of wavelet coefficients between an undistorted image and its distorted version is weighted with the foveation error sensitivity according to the visual attention. In [68], a video presentation is transferred from its original Cartesian coordinate to the curvilinear coordinate by a foveation filtering operation, and then the distortion is calculated with weighted signal-to-noise ratio. In [74], various quality metrics are modified by weighting the original metrics with a saliency map derived from the eye tracking data of visual attention, and improvements in performance was observed, comparing with the metrics without visual attention. An overview of applying visual attention in quality assessment is given in [39]. In these methods, it simply gives greater weights to the distortion in the attended areas at the pooling stage, and the weight is usually designed intuitively. Therefore, it is hard to justify and develop a proper and accurate weighting scheme that could work the same way as the HVS in balancing the attended and unattended distortions.

Another important characteristics to consider in video quality is the masking effect, which refers to human's reduced ability to detect a stimulus on a spatially or temporally complex background. The traditional way to measure the masking

effect is using a divisive gain control method, which decomposes the video into multiple channels and analyzes the masking effect among the channels by divisive gain normalization [66] and [129]. However, the mechanism of gain control mostly remains unknown. Additionally, since only simple masker such as sinusoidal gratings or white noise is used in the experiments to search for optimal parameters to fit the gain control model, there is no guarantee that these models are applicable to natural images [26]. In [128] and [47], it is pointed out that masking effect highly depends on the level of randomness created by the background. Usually the regular background contains predictable content and the stimulus will become distinct from neighborhood when it is different from human's expectation of its position. While in the random background, the content is unpredictable, and thus any change on it will be less noticed. Therefore, there is higher masking in the random background than the regular background. In [128], the concept of entropy masking is proposed to measure masking effect of background using zero order entropy. However, it only measures masking in spatial domain, for videos, which is obviously inadequate, because the temporal activities will also affect the visibility of distortion significantly. Usually distortion is highly masked in the massive and random motions while less masked in regular and smooth motions. In [135], the mismatch between two consecutive frames is used to measure temporal activities. However it may not reflect the regularity of motion precisely, since smooth and regular motion could also produce large mismatch. Therefore it is desired to develop the method that could measure the regularity of motion and thus measure the masking effect of videos.

On other hand, although MSE has been criticized for the low correlation to the HVS, due to its low computational cost, it is still used widely in practice. The inaccuracy of MSE in perceptual quality prediction comes from the lack of

psychophysical designs in HVS, like counting the imperceptible distortions. In this work, we revise the MSE by incorporating important HVS characteristics. First, to remove the imperceptible distortion from MSE, a low-pass filter is designed based on the CSF and visual attention. Since the contrast sensitivity is affected both by frequency and visual attention, visual saliency is introduced to adjust the cutoff frequency in the CSF so that the developed low-pass filter could adaptively remove the imperceptible distortion according to the location that is attended or not. In this way, the problem of non-uniform sampling of visual acquisition is solved naturally by removing less high frequency distortion in salient regions and more in non-salient regions. In addition, the masking modulation is applied afterward to reduce the imperceptible distortion covered by masking. Because a smooth and regular motions will hide less distortion than massive and irregular motions, we first propose a method to measure the randomness of video with a dynamic model. Since video content is easier to predict with regular motion than random motion, the prediction error actually reflects the randomness of video and can be used as the measurement of randomness to indicate how much the background could mask the noise. Furthermore, we investigate the model of masking modulation, which quantitatively analyzes how the modified MSE should be compensated according to the proposed randomness. The analysis is performed based on the relation between the modified MSE and perceptual quality scores across different video contents.

4.1 Foveated Low-pass Filter

The initial visual signal processing in HVS includes two steps. In the first step, the visual signal goes through eye's optics, forming an image on the retina. Because

of the diffraction and other imperfections in the eye, such processing would blur the passed image. In the second step, the image will be filtered by neural filters as it is received by photoreceptor cells on retina and then passed on to Lateral Geniculate Nucleus (LGN) and the primary visual cortex. These processes are more like low-pass filtering and will hide considerable high frequency information from perception.

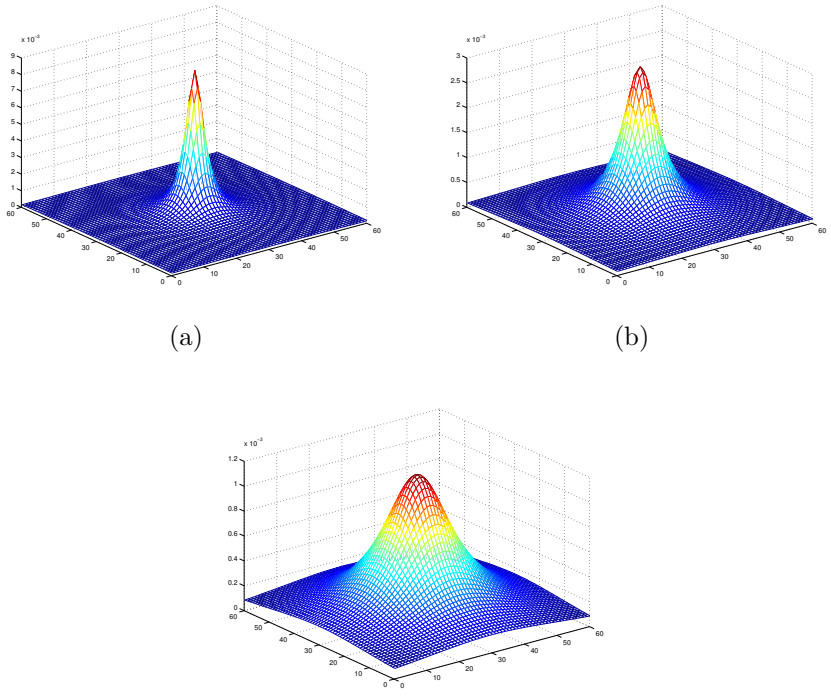
4.1.1 Low-pass filtering with CSF

The CSF, which is defined as the inverse of contrast threshold of detectable contrast at a given frequency, provides a comprehensive measure of spatial vision. Although it is not exactly equivalent to modulation transfer function (MTF), it reflects the same trend as the modulation gain. For instance, higher sensitivity at particular frequencies always means higher modulation gain at the corresponding frequencies and vice versa. Therefore, many researchers have treated the CSF as the spatial MTF, and used it to define characteristics of initial processing in HVS [31, 13, 127]. There are various CSF models [98, 132], and the typical CSF could be modeled as a function of frequency [132]:

$$\mathrm{CSF}(f) = (a + b \cdot f)e^{-c \cdot f}, \tag{4.1}$$

where a, b and c are model parameters and f is spatial frequency. Therefore the processed visual signal after passing through the initial part of HVS can be modeled as

$$I' = F^{-1}(\mathrm{CSF}(f)) * I, \tag{4.2}$$



(a) (b)

(c)

Figure 4.1: The foveated low-pass filter. (a) $e = 0$. (b) $e = e_2$. (c) $e = 3e_2$.

where I' and I are the processed and original visual signal, respectively; F^{-1} is the inverse Fourier transform and $*$ is the convolution operation.

4.1.2 Foveated low-pass filter

Our gaze is mainly driven to follow the most salient regions, and the distortions that occur outside the salient areas are assumed to have a lower impact on the overall quality. This is because the photoreceptor cells are not equally distributed, but they are dense in the fovea and sparse on the peripheral retina. Therefore, the gazed regions on an image has better visual resolution in the HVS and consequently it is less blurred while the regions outside foveation will lose much more details. However, in Eq. (4.2), the whole image is processed with the same filter, without

considering the effect of visual attention. Since the contrast sensitivity changes with location of the image projected onto the retina, the filter should be adaptively changed rather than using a constant filter.

The contrast sensitivity is a function of both spatial frequency and the position projected on the retina. In [42], a model of contrast threshold is developed based on the spatial frequency of the visual signal and its retinal eccentricity to the fixation. Since the contrast sensitivity is the inverse of the contrast threshold, the corresponding CSF could be expressed as

$$\mathrm{CSF}(f,e) = \frac{1}{CT_0} \cdot exp\left(-\mu \cdot f \cdot \frac{e+e_2}{e_2}\right), f > 0, \tag{4.3}$$

where f is the spatial frequency in cycles/deg, e is the retinal eccentricity, CT_0 is a constant presenting the minimum contrast threshold; e_2 is the half-resolution eccentricity; μ is the spatial frequency decay constant. The retinal eccentricity e is the angle between the fixation and the location of the signal and it is related to the distance between the two points and the viewing distance. According to Eq. (4.3), the contrast sensitivity decreases as the retinal eccentricity increases.

By transforming the CSF in Eq. (4.3) into spatial domain, we have the impulse response of initial processing system in the HVS as

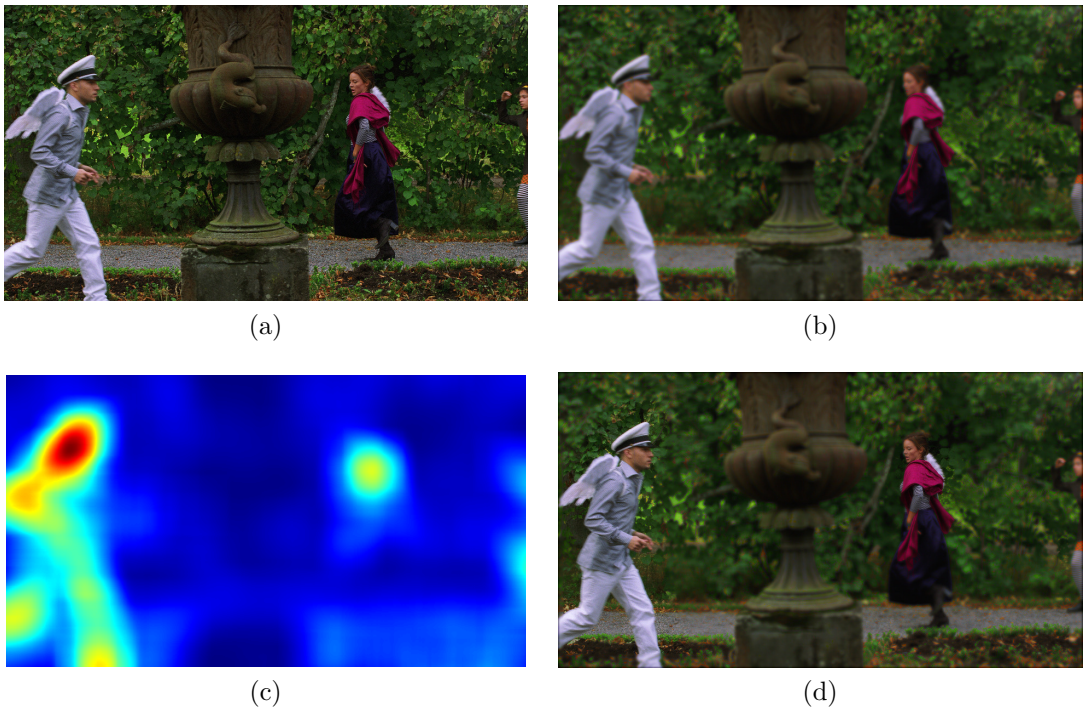
$$h(d,e) = \frac{1}{\pi CT_0} \cdot \frac{\mu(e+e_2)e_2}{e_2^2 d_F^2 + \alpha^2 (e+e_2)^2}, \tag{4.4}$$

where d_F is the distance from the filter center, *i.e.,* $d_F = \sqrt{x^2+y^2}$. Fig. 4.1 shows the impulse response of the developed filter at different locations. We can see that on the fixation (*i.e.,* $e = 0$), the impulse response is sharp, which means the content is less blurred, while as the distance increases, the impulse response spread

into larger nearby areas, making the content more blurred. This is consistent with the characteristics of the HVS that there is high acuity on the fixation location.

4.1.3 Computational model of eccentricity

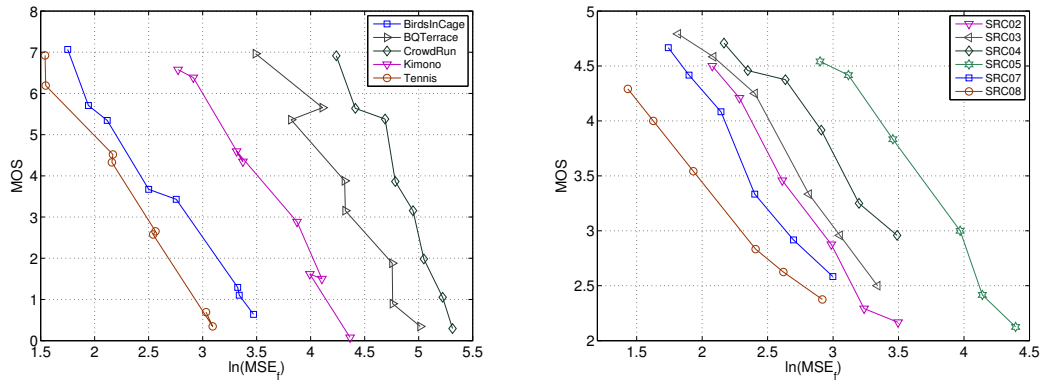
Since the visual acuity varies on the different location of a video, accurate prediction of visual attention is critical. Recording eye movements is so far the most reliable means for studying human visual attention and it provides the ground truth of the fixation locations on videos. It is highly desirable to incorporate these information into the developed foveated low-pass filter. However, recording such data requires extra equipment like eye tracking devices and the experiments are expensive and time consuming. More importantly, since human is involved in the process, it is impossible to develop it into objective quality metrics where each component should be automatic. An alternative way is using saliency detection algorithms. In general, saliency is defined as what attracts human perceptual attention. Computational visual attention models trying to predict the gaze location of human with features from images or videos could be generally classified into two categories: a bottom-up approach [20, 52, 11, 44, 22] and a top-down approach [57]. Top-down methods are task dependent and based on prior knowledge about scenes and objects. In contrast, bottom-up methods are scene-dependent and based on stimulus-driven mechanism. Usually, bottom-up approaches are computationally efficient. For example, an efficient saliency detection method is proposed in [52], where a set of feature maps from three complementary channels as intensity, color, and orientation are normalized respectively and then linearly combined to generate the overall saliency map. In this work, this algorithm is adopted based on its good performance and low computational complexity.



(a) (b) (c) (d)

Figure 4.2: Visual illustration of foveated low-pass filtering. (a) Original image. (b) Filtered with constant low-pass filter. (c) Saliency map. (d) Filtered with foveated low-pass filter.

The saliency map quantifies the possibility of the locations being the gazed locations. A location with a large value in the saliency map is more likely to be gazed and hence the eccentricity of that location projected on the retina will be small, *vice versa*. Therefore, the retina eccentricity of a location increases as its visual saliency value decreases. In [74], the saliency value is assumed to be gaussian distributed around the fixation as $s = exp\,(-d_E^2/\sigma^2)$, where d_E is the distance from fixation and σ is the model parameter. Since our saliency map is generated by computational saliency models and the actual distribution depends on the employed computational saliency models, instead of using gaussian distribution, we apply a more general distribution as



(a) (b)

Figure 4.3: Relation of MOS and $ln(\mathrm{MSE_f})$ for different video sequences. (a) On the MCLV database [72]. (b) On the VQEG database [43].

$$s = exp\left(-\frac{d_E^{\theta}}{\sigma^2}\right), \tag{4.5}$$

where θ is the model parameter depending on different saliency detection algorithms and in our experiments $\theta = 4$. Based on Eq. (4.5), it is straightforward to use the visual saliency value to approximate the retina eccentricity as

$$\begin{aligned} e(i,j) &= arctan\frac{(-\sigma^2 ln(s(i,j)))^{\vartheta}}{L} \\ &\approx \gamma \cdot ln(1/s(i,j))^{\vartheta}, \end{aligned} \tag{4.6}$$

where $s(i,j)$ is the visual saliency value at position (i,j) and L is the viewing distance. $\gamma = \sigma^{2\vartheta}/L$, $\vartheta = 1/\theta$. The values of $s(i,j)$ within each frame are normalized into the range of $[0,1]$.

Table 4.1: The slopes and goodness of fitting

	SeqName	BB	BC	BQ	CR	DK	EA	EB	FB	KM
MCLV	Slope	-3.300	-2.186	-3.034	-4.267	-3.218	-3.957	-5.140	-3.252	-2.966
	R^2	0.937	0.987	0.987	0.971	0.994	0.987	0.994	0.978	0.984
	SeqName	SRC01	SRC02	SRC03	SRC04	SRC05	SRC06	SRC07	SRC08	SRC09
VQEG	Slope	-1.092	-1.333	-1.306	-1.434	-1.536	-2.206	-1.431	-0.975	-1.646
	R^2	0.986	0.988	0.986	0.977	0.991	0.937	0.980	0.993	0.959

4.1.4 Blockwise filtering

Since the contrast sensitivity is different in positions, the low-pass filtering that simulates the initial processing of the HVS could be applied with adaptive filters based on Eq. (4.3) and Eq. (4.6). For the constant filters, it is equivalent to apply filtering in frequency domain or spatial domain. However, since the proposed low-pass filter changes spatially, the spatial information will be lost in Fourier frequency domain, it only could be implemented in spatial domain as

$$\Delta I_f(i,j) = h(e) * (I_d - I_o) = h(e) * \Delta I. \tag{4.7}$$

Eq. (4.7) is computationally heavy, since for each pixel we have to generate a new filter according to the corresponding saliency values. Usually the saliency map is continuous and smooth, thus we could assume that the saliency value within a neighborhood is similar. Each frame of video is partitioned into $N \times N$ blocks and larger N could reduce the computational complexity but with coarser eccentricity estimation, while smaller N could provide finer estimation but with higher computational cost. In our experiments, block size is set to 32×32 for a good balance between accuracy and computational complexity. For the kth block B_k, the average eccentricity of the block

$$\bar{e}_k = \frac{1}{N^2} \sum_{(m,n) \in B_k}^{N} e(m,n) \tag{4.8}$$

is used to present to visual attention. Thus a constant filter is applied within a block as

$$\Delta I_f(i,j) = h(\bar{e}_k) * \Delta I(i,j), \tag{4.9}$$

where $(i,j) \in B_k$.

The visual illustration of foveated low-pass filtering is shown in Fig. 4.2. We can see that in Fig. 4.2(b), the high frequency signals are equally removed cross the content, even in the regions that we are interested in. However, in Fig. 4.2(d), they are removed adaptively according to the saliency map shown in Fig. 4.2(c) and high frequencies remain in the salient regions.

After the adaptive low-pass filtering, MSE is calculated as the mean of sum of squared difference between the original and compressed video sequences as

$$\mathrm{MSE_f} = \frac{1}{WHL} \sum_{t=1}^{L} \sum_{i=1,j=1}^{WH} \Delta I_f(i,j,t)^2, \tag{4.10}$$

$$D = ln(\mathrm{MSE_f}), \tag{4.11}$$

where W, H and L are the width and the height and the duration of the video sequences; ΔI_f is the distortion after low-pass filtering.

4.2 Perceptual Modulation

The visibility of distortion highly depends on the content of background. Usually strong masking effect could prevent the distortion from being observed and thus reduce the distortion perceptually. Therefore it is important to measure the masking effect. In [128], it is pointed out that masking effect highly depends on the

level of randomness created by the background. For videos, randomness should be measured both in spatial and temporal domains.

4.2.1 Displacement of metric curves

The relationship between the MOS and D in Eq. (4.11) is shown in Fig. 4.3 for various sequences from different databases. Each point corresponds to a distorted video sequence and metric curves are formed by connecting the points sharing the same original video. In other words, the connected points in Fig. 4.3 are video sequences compressed from the same original sequence but with different compression levels. Under the same video content, D is a good predictor of perceptual quality (*i.e.,* MOS), since the MOS monotonically decreases with D.

However such relation can not be applied to distorted videos with different contents. As we can observe in Fig. 4.3, there are different horizontal displacements for the metric curves of different video contents. Such difference in horizontal displacement mainly comes from the different masking effect of various video contents. Given the same MOS, the points of metric curves on the right side have more actual distortion *i.e.,* $\mathrm{MSE_f}$, than on left as shown in Fig. 4.3, which means the video in the right metric curve has more masking and that makes it have the same perceptual quality as the videos on the left side. Therefore, the videos with strong masking effect are more likely to have metric curves on the right side, and the displacement of these curves with respect to the left side reflects the significance of masking effect.

To quantitatively analyze the masking effect, we assume the shapes of the curves in Fig. 4.3 are identical by neglecting the small differences among them. The points of same contents are fitted with linear curves and the slopes of different curves are presented in Table 4.1 as well as the goodness of fit R^2. We can see that

within each database, the slopes of most video sequences are close to each other, which means that the shapes of these curves are almost the same. R^2 describes how well the linear model fits to the actual data and the closer to 1 its value is, the better the mode is. Although the values of R^2 in Table 4.1 are all so close to 1 that means linear model is accurate, it is not necessary to limit the model to linear. Instead, as long as the shape of these curves are the same, we could generalize the relation of D and MOS as

$$\widehat{\text{MOS}} = F(D - P), \tag{4.12}$$

where P is the horizontal displacement depending on the video content and $F(\cdot)$ is could be a linear function or other monotonic decreasing function representing the shape of these curves. P reflects the masking effect of the video content. Strong masking effect always results in large P values. Since $F(\cdot)$ is fixed in Eq. (4.12), an accurate estimation of P is critical to the MOS prediction. Due to the difference of masking effect, P varies significantly from sequence to sequence.

(a) (b) (c) (d) (e)

(f) (g) (h) (i) (j)

Figure 4.4: Visual illustration of temporal randomness on two different video sequences.

4.2.2 Temporal and Spatial Randomness

To measure the masking effect of video content, the regularity of video content is analyzed quantitatively both in spatial and temporal domains. As an important characteristics of video, motion information is highly related to masking activities. Usually distortion is highly masked in the massive and random motions while less masked in regular and smooth motions.

For regular motion, the future frames can be predicted from the past frames by learning the temporal behavior of a short video clip in past. Thus the prediction error reflects the randomness of motion. To capture the temporal activities of past video, the video sequence can be modeled as a discrete-time dynamic system [32]. To simplify the problem, the video signal is modeled as a linear dynamic system as in [19]. Let $Y_k^l = [y(k), \cdots, y(l)] \in \mathbb{R}^{m\times(l-k)}$ denote a short sequence from the kth frame to the lth frame and each frame is rearranged into a column vector $y \in \mathbb{R}^m$, where m equals to the number of pixels within a frame, *i.e.,* $m = W \times H$. The motion in video is simulated as evolution process of a dynamic system, described as

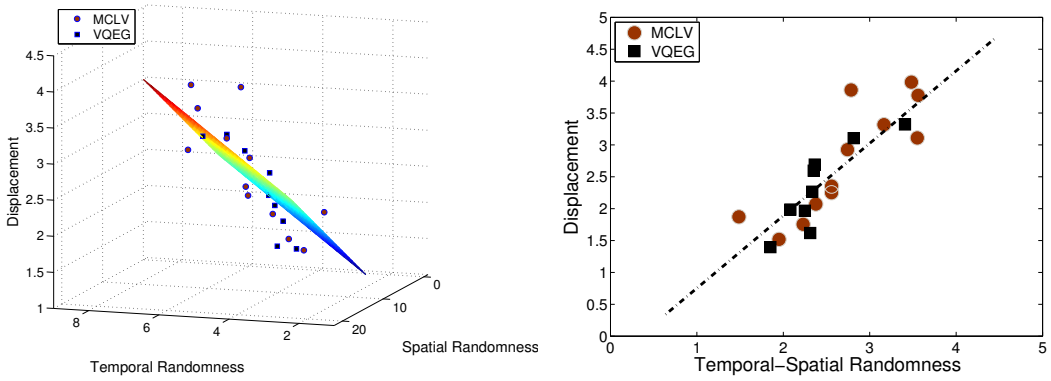
$$\begin{cases} Y_k^l = CX_k^l + W_k^l \\ \\ X_k^l = AX_{k-1}^{l-1} + V_k^l \end{cases}, \tag{4.13}$$

where $X_k^l = [x(k), \cdots, x(l)]$ and $X_{k-1}^{l-1} = [x(k-1), \cdots, x(l-1)] \in \mathbb{R}^{n\times(l-k)}$ are the state sequences of Y_k^l and Y_{k-1}^{l-1}, respectively, and $m > n$. $A \in \mathbb{R}^{n\times n}$ is the state transition matrix encoding the regular motion information and $V_k^l \in \mathbb{R}^{n\times(l-k)}$ is the sequence of motion noise that can not be represented by regular information A. $C \in \mathbb{R}^{m\times n}$ is the observation matrix encoding the shapes of objects within the frames and $V_k^l \in \mathbb{R}^{n\times(l-k)}$ is the sequence of observation noise that can not

be represented by regular shape information C. Given the video sequence Y_k^l, the model parameters A, C and the state sequence X_k^l is not unique. There are infinite choice of these matrix which could give exactly the same video sequence Y_k^l. An efficient method was proposed in [12], which employs singular value decomposition and keeps the n largest singular values as,

$$Y_k^l = U\Sigma V^T + W_k^l, \tag{4.14}$$

where $\Sigma = \text{diag}[\sigma_1, \cdots, \sigma_n]$ contains the n largest singular values and $U \in \mathbb{R}^{m\times n}$, $V \in \mathbb{R}^{(l-k)\times n}$ are corresponding decomposition vectors. By setting $X_k^l = \Sigma V^T$ and $C(l) = U$, we could determine the state sequence and the model parameter C. Since the redundancy in Y_k^l is removed by reducing the dimension from m to n, X_k^l is the compact representation of Y_k^l with a loss of information W_k^l .



(a) (b)

Figure 4.5: (a) The relation between horizontal displacement P and temporal randomness and spatial complexity. (b) Combined temporal and spatial randomness.

Moreover A is expected to capture the motion information and thus predict future frames. The optimal A could be found by minimizing the squared prediction error as

$$\hat{A}(l) = \underset{A}{\operatorname{argmin}} ||X_{k+1}^{l} - AX_{k}^{l-1}||. \tag{4.15}$$

Therefore the optimal solution could be obtained as

$$\hat{A}(l) = X_{k+1}^{l} X_{k}^{l-1^{+}}, \tag{4.16}$$

where $X_{k}^{l-1^{+}}$ is the pseudoinverse of X_{k}^{l-1}. We could predict future frame $y(l+1)$ based on the obtained model parameters, *i.e.,* $A(l)$, $C(l)$ that characterize the temporal activities of sequence Y_{k}^{l}. The prediction error could be calculated as

$$R_T(l+1) \quad = \quad |y(l+1) - C(l)A(l)x(l)|, \tag{4.17}$$

where $R_T(l+1) \in \mathbb{R}^m$ actually is the noise that could not be predicted with regular information. This value reveals the predictability of the next frame according to trajectory of moving objects in the past frames and thus reflect its temporal randomness. Usually smooth and regular motions in videos will make future frames more predictable than massive and random motions. Fig. 4.4 shows the temporal randomness for two sequences. Fig. 4.4 (a)-(d) and (f)-(i) show the frames of the sequence “ElFuente2" and “OldTownCross", respectively, and the Fig. 4.4(e) and Fig. 4.4(j) show the corresponding temporal randomness calculated from Eq. (4.17). In the background of sequence “ElFuente2", the motion of water drops is unpredictable and thus its temporal randomness is large. While in the sequence “OldTownCross", the motion is smooth and regular. Consequently its temporal randomness is much smaller than that of the sequence “ElFuente2". Finally, the average temporal randomness is used to represent the overall temporal randomness of the whole video as

$$\bar{R}_T = \frac{1}{m \cdot L} \sum_{l=1}^{L} \sum_{i=1}^{m} R_T^i(l), \tag{4.18}$$

where $R_T^i(l) \in \mathbb{R}^m$ is the ith component of $R_T(l)$ and L is the total number of frames.

Besides the temporal domain, the spatial activities of the frame also affect the masking effect. The pixel variance of $N \times N$ block is computed to indicate the local spatial randomness and the logarithm of the mean of the local spatial randomness is utilized as spatial randomness of the whole video as

$$\bar{R}_S = ln \left(\frac{1}{M \cdot L} \sum_{t=1}^{L} \sum_{i=1}^{B} \sigma^2(i,t) \right), \tag{4.19}$$

where $\sigma^2(i)$ is the variance of the ith $N \times N$ block in the tth frame; B and L are the total number of blocks within a frame and total number of frame within a sequence.

4.2.3 Modulation

As discussed above the displacement of metric curves in Eq. (4.12) reflects the masking effect and it relates to the temporal and spatial activities of the video sequences. To investigate its relation to temporal randomness $\bar{R}_T$ and spatial randomness $\bar{R}_S$, we have to measure the actual horizontal displacement first. That can be determined by measuring horizontal position of the crossing points of the metric curves with any horizontal lines such as $MOS = 3.0$. The relation of actual displacement P with temporal randomness $\bar{R}_T$ and spatial randomness $\bar{R}_S$ is shown in Fig. 4.5. In Fig. 4.5(a), each point represents a video sequence either from the database MCLV or the database VQEG, we could see that the displacement has

linear relation with $\bar{R}_T$ and $\bar{R}_S$, respectively. Thus, it could be approximated with a linear surface and the displacement could be predicted as

$$\hat{P}_i = \alpha \bar{R}_T + \beta \bar{R}_S, \tag{4.20}$$

where α and β are model parameters and fixed at 0.315 and 0.372, respectively. Fig. 4.5 (b) shows the relation between the actual and the predicted displacement. Combining the Eq. (4.11), (4.12) and (4.20), we have

$$\begin{aligned} \widehat{\mathrm{MOS}} &= F\left(ln(\mathrm{MSE_f}) - \alpha \bar{R}_T - \beta \bar{R}_S\right), & (4.21) \\ &= G(\mathrm{MSE_f} \cdot e^{-(\alpha \bar{R}_T + \beta \bar{R}_S)}). & (4.22) \end{aligned}$$

where $G(\cdot) = F(ln(\cdot))$. It is acceptable for a quality metric to predict MOS through a nonlinear mapping, because the mapping is easy to be found and it depends on various environmental factors like the range of MOS and evaluation methodology. Therefore, in [1] and [2], a nonlinear mapping is not considered as part of VQM, rather it is left to the final stage of performance evaluation. $G(\cdot)$ could be obtained by fitting the objective prediction scores to the subjective quality scores as described in [1, 2]. We use the perceptually weighted distortion

$$\mathrm{MD} = \mathrm{MSE_f} \cdot e^{-(\alpha \bar{R}_T + \beta \bar{R}_S)} \tag{4.23}$$

as the MOS predictor. In this way, the MSE is modified according to the HVS characteristics and thus become more correlated with the perceptual quality.

4.2.4 Context effect

The MOS of a video mainly is determined by the its perceptual quality, but also affected by the perceptual quality of other videos during subjective tests. For example, when a video with medium quality is evaluated in a pool of severely impaired videos, it will get a higher MOS than it is evaluated in a pool of high quality videos. Such phenomenon is called context effect. Although various subjective tests are designed carefully to reduce such effect, it can not be removed completely in subjective tests [101, 7]. Usually the quality of former displayed videos will affect MOS of latter videos, but since the display order of the videos are random for each subject, it is reasonable to assume that each video has equal chance to be affected by other videos in subjective tests. Assuming that the MOS of a video would be equally affected by other videos, a slight shift in MOS might be caused with the general perceptual quality of the context, which is expressed as

$$\text{MOS} = Q - \eta \cdot \bar{Q}, \tag{4.24}$$

where $\bar{Q}$ is the average perceptual quality of all videos displayed in subjective tests and η is a penalty coefficient reflecting how much other videos would affect the quality of current video. For example, $\eta = 0$ means MOS is not affected by quality of other videos. So far such shift in MOS will not affect the performance evaluation of quality assessment.

However, in the actual subjective test, the MOS of a particular video may receive different impact from different videos. The MOS of a video is more likely to be affected by the videos with similar contents and distortion types. In other words, when the subjects provide quality scores, they intend to compare the quality of current video with previous similar videos with similar distortion types and the

resulting quality score will be affected by these videos more than others. In this work, we focus on the same distortion types, *i.e.,* compression distortion, and thus only the content is concerned. To measure similarity of videos, besides the temporal randomness in Eq. (4.17) and spatial randomness measured in Eq. (4.19), the color information is also extracted, due to the fact that color also plays important role in quality assessment as described in [17]. Therefore color feature for each frame is extracted as

$$cv = det \begin{pmatrix} \sigma_Y^2 & \sigma_{YU}^2 & \sigma_{YV}^2 \\ \sigma_{YU}^2 & \sigma_U^2 & \sigma_{UV}^2 \\ \sigma_{YV}^2 & \sigma_{UV}^2 & \sigma_V^2 \end{pmatrix}, \tag{4.25}$$

where $\sigma_Y^2, \sigma_U^2, \sigma_V^2$ are the variance of Y, U, V components in YCbCr color space, respectively; $\sigma_{YU}^2, \sigma_{YV}^2, \sigma_{UV}^2$ are the covariance of three component, respectively. The mean value $\bar{cv}$ along the temporal domain is used for each sequence. Therefore we measure the distance between the ith and the jth videos in the feature space as

$$d(i,j) = \frac{\kappa_1 |\bar{cv}_i - \bar{cv}_j|}{\bar{cv}_i + \bar{cv}_j} + \frac{\kappa_2 |\bar{R}_{Ti} - \bar{R}_{Tj}|}{\bar{R}_{Ti} + \bar{R}_{Tj}} + \frac{\kappa_3 |\bar{R}_{Si} - \bar{R}_{Sj}|}{\bar{R}_{Si} + \bar{R}_{Sj}}, \tag{4.26}$$

where κ_1 - κ_3 are constant model parameters indicating the importance of the features and they are set to 1 in our experiments. The videos with smaller distance $d(i,j)$ will affect the MOS of each other more than the videos with larger distance.

To simulate the impact of other video quality on the MOS and meanwhile take the content distance into consideration, we modify the quality metric in Eq. (4.23) and propose the Perceptually Weighted MSE as

$$\text{PW-MSE}(i) = MD(i) - \eta \left(\frac{1}{\Delta_i} \sum_{j \in V, j \neq i} e^{-d(i,j)} \cdot MD(j) \right), \tag{4.27}$$

where $e^{-d(i,j)}$ is the weighting factor and $\Delta_i = \sum_{j \in V, j \neq i} e^{-d(i,j)}$ is used for normalization; V is the set of videos in context and $\eta = 1$. If the content similarity among videos is identical, Eq. (4.27) becomes Eq. (4.24) and the context effect vanishes in terms of quality prediction, because a constant added to the metric will not affect the final performance.

Table 4.2: Intermediate performance at each stage

	PCC			**SROCC**			**RMSE**		
	MSE	**FoveatedPW**		**MSE**	**FoveatedPW**		**MSE**	**FoveatedPW**	
MCLV	0.4526	0.6416	0.9576	0.4442	0.6265	0.9649	2.7975	1.7022	0.6391
VQEG	0.6907	0.7310	0.9323	0.6816	0.7412	0.9030	0.6309	0.5710	0.3155
IRCC	0.7960	0.9098	0.9351	0.8050	0.8944	0.9167	0.6369	0.4511	0.3853

4.3 Experimental Results

Table 4.3: Overall performance on various databases

	MCLV			**VQEG**			**IRCCyN**		
	PCC	**SROCC**	**RMSE**	**PCC**	**SROCC**	**RMSE**	**PCC**	**SROCC**	**RMSE**
PSNR	0.472	0.464	1.957	0.690	0.668	0.631	0.810	0.805	0.637
SSIM	0.452	0.470	1.979	0.641	0.536	0.670	0.834	0.830	0.600
VIFp	0.518	0.511	1.898	0.704	0.663	0.619	0.837	0.832	0.595
MS	0.681	0.663	1.625	0.854	0.855	0.454	0.917	0.911	0.433
ST-MAD	0.579	0.623	1.810	0.670	0.558	0.648	0.912	0.909	0.446
MOVIE	0.625	0.627	1.733	0.768	0.877	0.559	0.753	0.900	0.716
VQM	0.763	0.783	1.433	0.880	0.876	0.415	0.918	0.910	0.431
PW	**0.958**	**0.965**	**0.639**	**0.932**	**0.903**	**0.315**	**0.935**	**0.917**	**0.385**

4.3.1 Subjective Databases and Performance Metrics

The performance of the proposed video quality matric was evaluated in the three databases including the MCLV [72], the VQEG [43], the IRCCyN databases. In

the MCLV, there are 12 original video sequences with the solution of 1920×1080. Two types of compression distortion are involved in the MCLV database. In the first type of distortion, the original sequences are compressed with H.264/AVC codec, generating four different quality levels. In the second type of distortion, the original sequences are first downscaled and compressed with H.264/AVC codec at four quality levels. Then the compressed sequences are upscaled to the original resolution. There are totally 96 distorted video sequences in the MCLV database. In the VQEG database, the original sequences are from the VQEGHD 3 of the VQEG project and there are 9 original sequences with the resolution of 1920×1080. In the database VQEGHD 3, besides the compression distortion types, there are several other distortion types like transmission error. Since we are only interested in compression distortion, only six distorted sequences with compression distortion were selected for each original sequences. There are totally 54 distorted video sequences. In the IRCCyN database, there are sixty original sequences with the resolution of 640×480. The videos are encoded with H.264/AVC and the codec of scalable video coding (H.264/SVC). Each original video is encoded at four different quality levels. Thus there are totally 240 distorted videos.

Since some performance metrics such as the linear correlation coefficient requires to compare linear correlation, for fair comparison the nonlinear mapping is carried out between the objective score and MOS. The following nonlinear function is employed before performance evaluation for all video quality metrics.

$$q(x) = \alpha_1 \left(0.5 - \frac{1}{1 + exp(\alpha_2(x - \alpha_3))} \right) + \alpha_4 x + \alpha_5, \tag{4.28}$$

where α_1 to α_5 are the parameters obtained by regression between the input and output data. As for metrics of performance evaluation, the Pearson correlation coefficient (PCC), Spearman rank order correlation coefficient (SROCC) and root

mean squared error (RMSE) are employed as described in [1, 2]. PCC generally indicates the goodness of linear relation. SROCC is computed on ranks and thus depicts the monotonic relationships. RMSE computes the prediction errors and thus depicts the prediction accuracy.

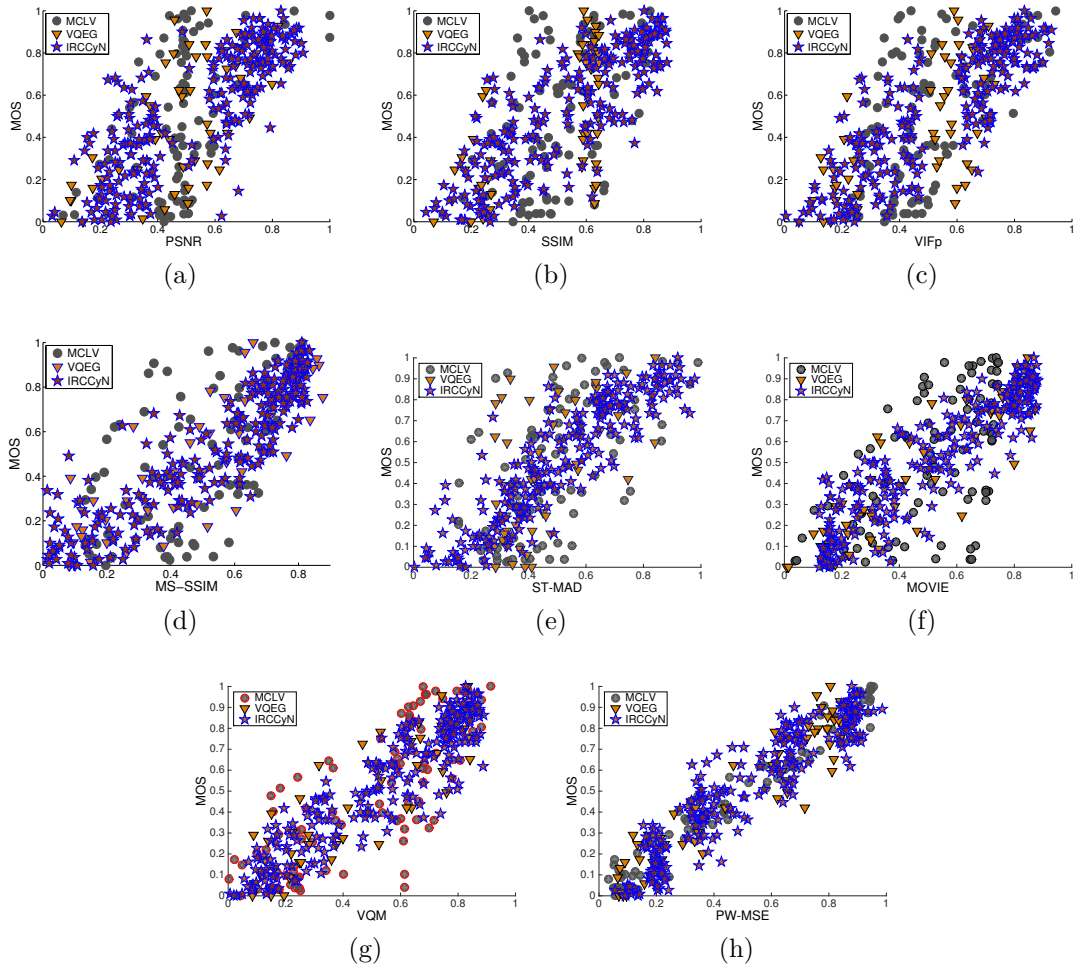


Figure 4.6: Scatter plot of MOS vs predicted MOS by various quality metrics.

4.3.2 Performance at two stages

The proposed algorithm consists of two main stages to simulate the visual signal processing in the HVS. In the first stage, the foveated low-pass filtering is

implemented to simulate the initial processing of the HVS. Then masking effect is considered to simulate high level processing in the HVS. To verify the effectiveness of each step in the proposed algorithm, the intermediate results of each step was investigated. The results are summarized in Table 4.2, where the performance of the MSE is listed in the first column of three performance evaluation methods, followed by the performance of the foveated low-pass filtering (denoted as Foveated) and PW-MSE.

As we can see in Table 4.2, the performance under each performance evaluation method is improved at each stage under all databases. In the MCLV database, MSE does not perform well comparing with in other databases, only achieving around 0.45 and 0.44 in PCC and SROCC respectively. Even after processing with the foveated low-pass filtering, the performance is not improved significantly. This is because in the MCLV database, the video contents are quite diverse. That makes the masking effect vary dramatically among different sequences and as a consequence MSE becomes inconsistent over different video content. When masking effect is taken into consideration by introducing the masking modulation at the second step, we could see that the performance is improved to 0.9576, 0.9649 and 0.6391 in PCC, SROCC and RMSE, respectively. As far as the VQEG and IRCCyN databases are concerned, MSE achieves better performances than in the MCLV database and the performance is further improved at each step.

4.3.3 Overall performance

In this section, we compare the performance of the proposed method with other benchmarks including: PSNR, SSIM [140], VIFp [112], MS-SSIM [126], ST-MAD [120], VQM [102], MOVIE [109]. Among them, ST-MAD, VQM and MOVIE are video quality metrics and the rest are image quality metrics. For the image

quality metrics, the final quality score was computed with average pooling after quality scores were calculated for each frame. Default settings were used for all the benchmarks, except for MOVIE[1]. Only the luminance component is used for analysis. Table 4.3 summarizes the performance of all the video quality metrics in the MCLV, the VQEG and the IRCCyN databases, where the best performance is highlighted in boldface.

From Table 4.3, we could see that the proposed PW-MSE achieves the best performance among all the video quality metrics and performs consistently well that it obtains PCC and SROCC above 0.9 on all the three databases. In addition, except MS-SSIM, the performances of video quality metrics are generally better than that of image quality metrics. This is because temporal characteristics are considered in video quality metrics but not in image quality metrics.

The scatter plots of subjective quality score against objective quality score are shown in Fig. 4.6 for the three databases. In order to plot in the same scale, the MOS was normalized and the objective scores were obtained after applying non-linear fitting to MOS. We can see the width of the PW-MSE's scatter plot is the narrowest among the quality metrics, which implies it has better correlation between the objective and subjective quality scores than other metrics.

4.3.4 Computational complexity

The computational complexity of the proposed PW-MSE was investigated on a computer with a CPU of Intel Xeon 2.4 GHz and 64GB Memory. Expect MOVIE, all the quality metrics were implemented in Matlab and were running with Matlab

[1]Due to the limited computational capability, the frame interval of MOVIE is set to 32 for the MCLV and VQEG databases, instead of default value 8.

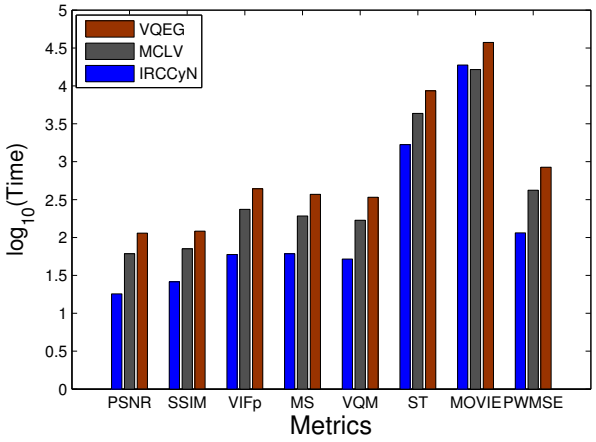


Figure 4.7: Comparison of the average consuming time for single video among different metrics. Since the consumed time of different metric varies significantly, the logarithmic scale is used for the consumed time.

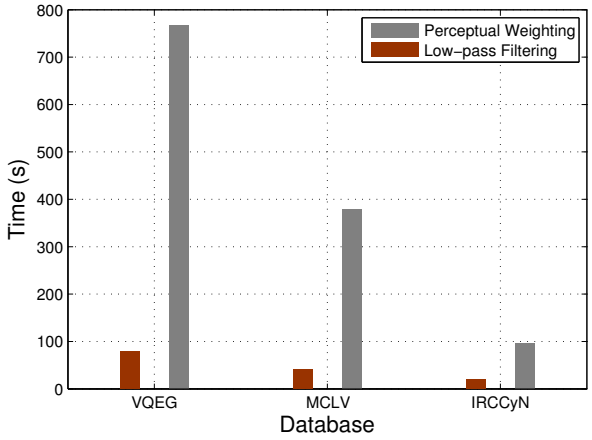


Figure 4.8: The average consumed time for single video at each stage of PW-MSE.

2014a. MOVIE was implemented in C and compiled with the GNU Compiler Collection (GCC).

The average consumed time for each video was measured and the results on the three database are shown in Fig. 4.7. We can see that due to video size such as resolution and frame number, the consuming time of each metrics increases from the IRCCyN database to the MCLV and VQEG databases, except for MOVIE,

which employed different parameters that reduce running time in the MCLV and VQEG databases. Meanwhile we can see that the consumed time of MOVIE is obviously much higher than other metrics, because it applies a set of time costing 3D filters convoluting with videos. All the image quality metrics consumed much less time, because they limit the analysis on spatial domain rather than temporal-spatial domain. The PW-MSE has comparable complexity with these image quality metrics and VQM, but much less than MOVIE and ST-MAD.

Furthermore, Since PW-MSE consists of two main stages, *i.e.*, low-pass filtering and masking modulation, their time consumption was investigated respectively. The average processing time videos was measured for each stage. The results are illustrated in Fig. 4.8, where we can see that masking modulation consumes more time than the foveated low-pass filtering, because the calculation of temporal and spatial randomness could relatively cost more time. Also in the adaptive low-pass filtering, the filter is updated at block level rather than pixel level, its computation is reduced significantly.

4.4 Summary

In this chapter, PW-MSE is proposed for compressed videos. The masking effect as well as the low-passing filter characteristics of the initial process of HVS is explored. To mathematically model and simulate the initial process in HVS, the foveated CSF is adopted as the transfer function in frequency domain. The error signal from the compression distortion is filtered with the proposed transfer

Chapter 5

Advanced Video Coding Techniques

5.1 Proposed Screen Content Video Coding

5.1.1 HEVC Edge Mode (EM) Scheme

For HEVC intra modes, the content of a square block, or more specifically a prediction block, is predicted using modes having different prediction angles. The mode yielding the least distortion, typically measured as mean-square error or mean absolute error, is selected to code the associated prediction block. Usually, regions with complex content are likely to be coded with a smaller block size such as 4×4 or 8×8, because prediction over a larger block would generate residuals having high energy. For screen content, many of the blocks contain smooth areas separated by a straight or curved edge. There are common methods outside of HEVC to represent such curves using multiscale straight or curved kernels, such as ridgelets and curvelets [117]. Within the HEVC framework, it is therefore reasonable to approximate edges as straight lines with different orientations in blocks of such small sizes. Furthermore, the prediction mode oriented along the edge is likely to produce less residual energy than a mode that predicts across the edge, as pixel values from neighboring blocks used during the prediction process are not good predictors of pixels on the opposite side of an edge. Consequently, if we have

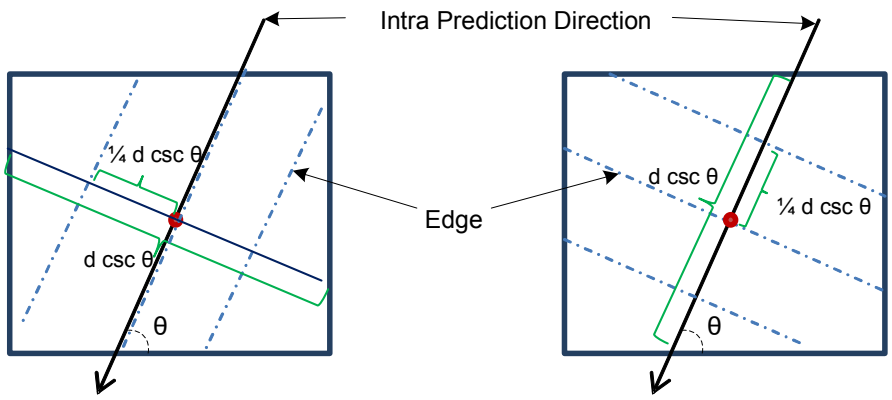


Figure 5.1: Dependency between prediction direction and the edge modes

selected the best intra prediction for a given block, it is likely that the edge orientation is in parallel with the intra prediction direction. The remaining unknown is the position of the edge within the block.

Along the edge direction, we consider three different edge positions as illustrated in Fig. 5.1: one passing through the center of block and the other two having a distance of $\frac{1}{4}d\csc\theta$ with respect to the center, where d is the block width and θ is the angle between the intra prediction direction and the horizontal line. Fig. 5.1 shows both the case when the edges are aligned with the intra prediction direction and the case when they are not aligned.

Computed over the first 50 frames of *SlideEditing*, Fig. 5.2 illustrates the histograms of four typical edge directions selected by the encoder during the RD-optimization process, when intra prediction is horizontal. As expected, the edge direction aligned with the prediction direction is used most frequently. The histogram also shows that the edge orientations are sometimes not aligned with the prediction direction. To cover these possibilities, three edge positions that are orthogonal to the intra predictions are also checked as illustrated in Fig. 5.1.

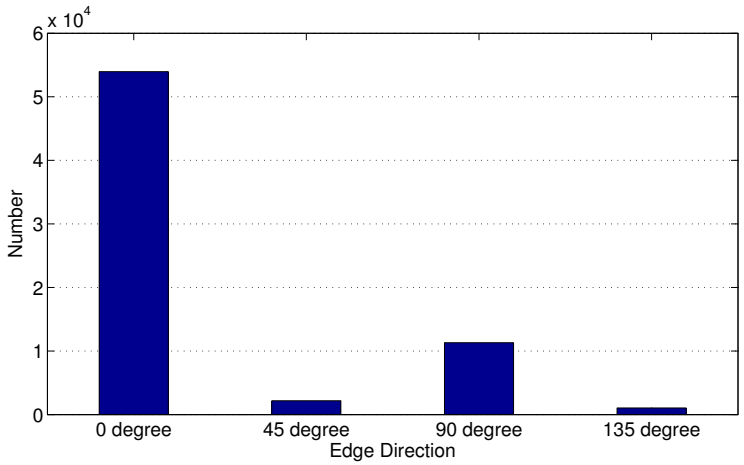


Figure 5.2: Histogram of edge direction when intra prediction is horizontal in *SlideEditing*

Thus, depending on the direction of a specific intra prediction mode, six edge positions are considered and the best one is selected by minimizing the RD cost

$$J = D_p + \lambda R_p, \tag{5.1}$$

where D_p and R_p are the corresponding distortion and number of bits associated with coding a block using an edge position denoted by p.

Simplification using Mode Classification

In order to achieve more accurate spatial prediction, HEVC supports a total of 35 intra prediction modes. The mode numbers and the corresponding prediction methods are shown in Fig. 5.3, where mode 0 is DC prediction, mode 1 is planar intra prediction, and modes 2 to 34 are directional modes covering 33 different prediction angles. Using six edge positions for each of the 33 directional prediction modes would require the encoder to perform up to almost 200 additional RD tests for each prediction block, which is impractical. To simplify the implementation

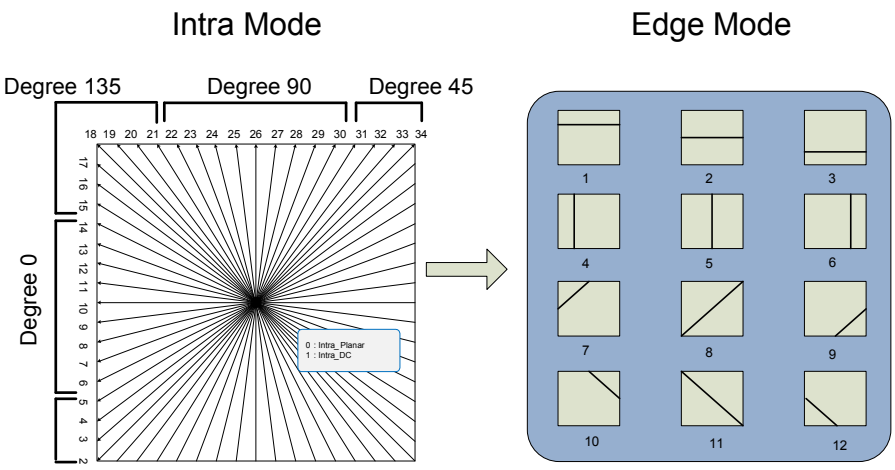
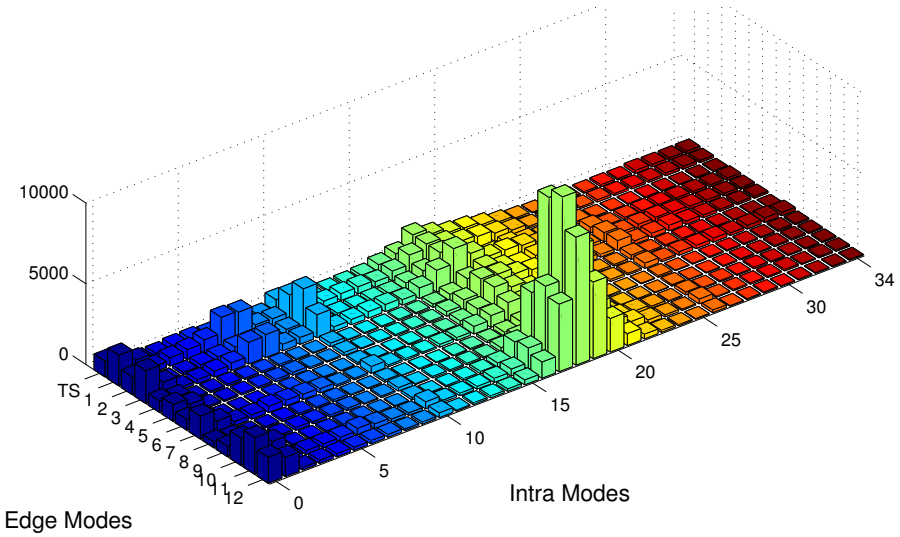


Figure 5.3: Intra modes classification and edge modes

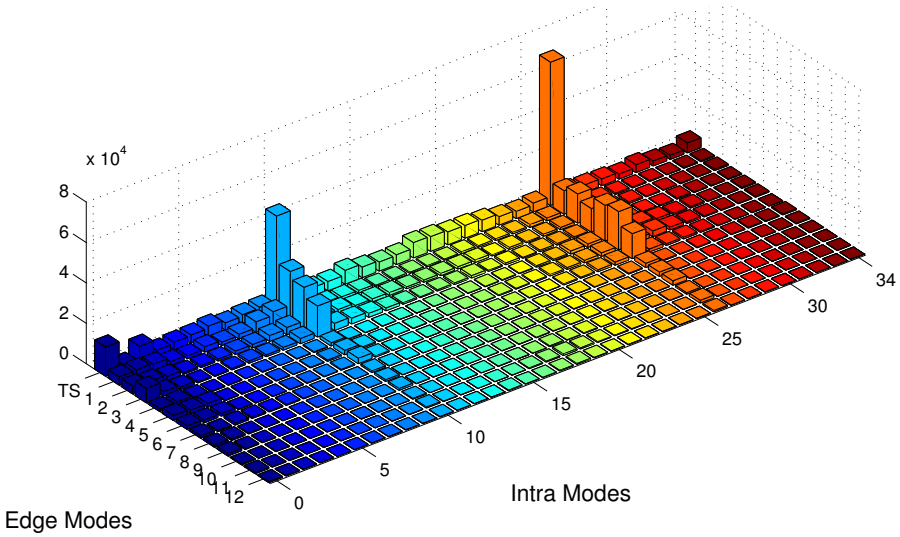
of the proposed EM scheme, we classify intra modes into four main directions as described below.

Intra modes with an approximately horizontal prediction direction, modes 6-14, are classified to a group called *Degree 0*. Similarly, modes 22-30 are classified as *Degree 90*; modes 2-5 and 31-34 are classified as *Degree 45*; and modes 15-21 are classified as *Degree 135*, as shown in Fig. 5.3. With the above simplification, the number of allowed edge positions is restricted to 12 edge modes, indexed from 1 to 12 as shown in Fig. 5.3. For the intra DC mode and the intra planar mode, edge modes 1 to 6 are more likely to be used based on our observations, so they are classified to *Degree 0*.

To verify the dependency between intra prediction modes and edge modes, we ran encoding experiments to check the optimal edge mode for each intra prediction mode using the RD optimization process. In the experiments, 50 frames of various screen content videos are encoded using Intra mode The histograms of the best edge modes for two representative sequences, *BasketballDrillText* and *SildeEditing*, are shown in Figs. 5.4(a) and (b), respectively. As shown in Fig. 5.4a, we see that blocks associated with diagonally-oriented prediction modes (e.g., modes 16-20)



(a) BasketballDrillText



(b) SlideEditing

Figure 5.4: Histogram of edge mode occurrence for different intra prediction modes

tend to be coded using diagonal edge modes. In contrast, diagonal edge modes are less likely to be chosen for vertical and horizontal intra prediction modes. The same observation applies to Fig. 5.4b. Here we see that many vertical and horizontal edge modes are used with the vertical and horizontal prediction modes. These

observations confirm that when vertical/horizontal intra predictions are used, vertical/horizontal edge modes are best suited for coding the residuals. When diagonal intra predictions are used, diagonal edge modes are more appropriate.

Applying Transforms to Sub-blocks

For each block that uses an edge mode, we partition it into two sub-blocks and then apply separable 2D DCT transforms. For edge modes 1-6, as pictured in Fig. 5.3, each sub-block is an $M \times N$ rectangle, so the existing horizontal and vertical transforms from HEVC can be applied. For edge modes 7-12, which partition the block into non-rectangular regions, several options similar to the shape-adaptive transform [114] can be considered. In this work, we develop directional 2D transforms that apply the DCT along separable paths in each partition

The directional 2D DCT used here comprises the following two steps:

1. First, a set of 1D DCTs is applied diagonally in alignment with the edge orientation. For example, as shown in Fig. 5.5, sub-blocks for edge modes 7 and 8 are first transformed along a diagonal direction. The DCT coefficients are also arranged from low to high frequencies along these paths. As a result, the lowest-frequency DCT coefficients are located along the top row and right column.

2. Next, the second set of 1D DCTs is applied along paths so that the DC coefficients of the first transform are covered by one DCT. The transform path for the AC coefficients are similar, as shown in Fig. 5.5.

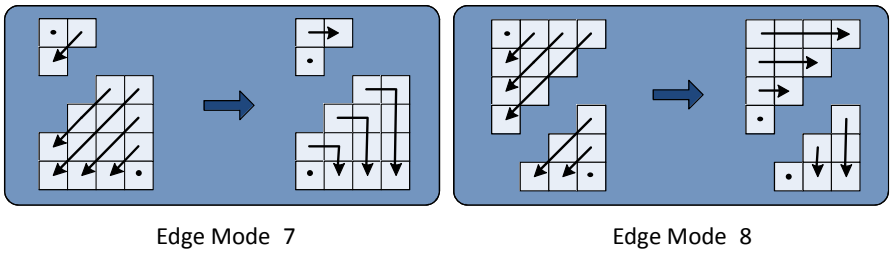


Figure 5.5: 2D DCT transforms for diagonal edge modes

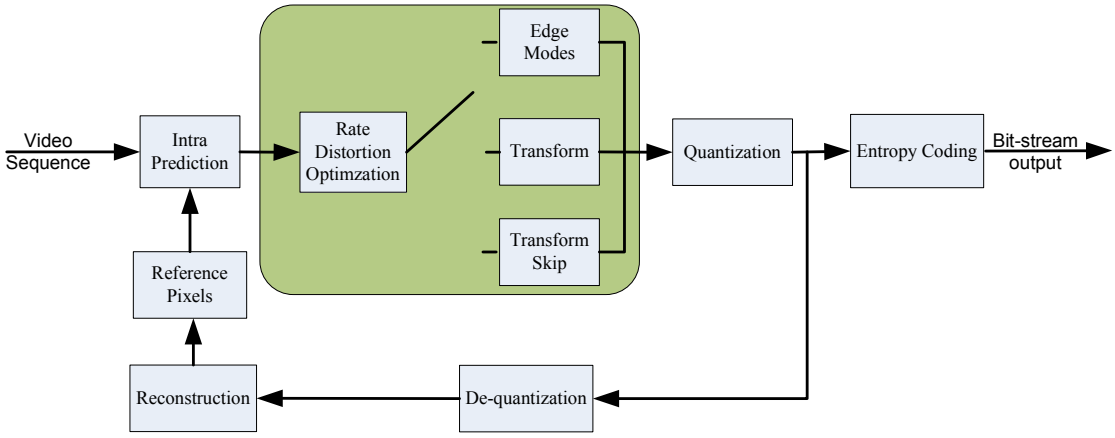


Figure 5.6: Flowchart of the proposed HEVC/EM

Integration and mode coding in HEVC

The proposed edge mode coding scheme is effective for blocks containing strong edges. For smooth blocks, the existing HEVC modes work best. Because screen content video often contains mixed natural and graphics material, we adaptively select among the existing and new edge modes for coding each intra prediction block. This combined scheme is called HEVC-plus-edge-mode (HEVC/EM). The encoder is illustrated in Fig. 5.6.

Up to three modes are checked for each intra block. As described earlier, a subset of edge modes are evaluated during RD-optimization, depending upon the intra prediction mode. Additionally, the unmodified HEVC transform is tested on the block, and transform-skip mode is checked as well, if enabled.

Table 5.1: Codewords for edge modes

Code	Edge Mode	Code	Edge Mode
000	HM	100	3 or 9
001	TS	101	4 or 10
010	1 or 7	110	5 or 11
011	2 or 8	111	6 or 12

Recall that the edge modes are separated into a horizontal/vertical set and a diagonal set, where the intra prediction mode determines which set is used. Therefore, we only need signal one of six edge modes. We also need to signal whether the existing HEVC transform or transform skip mode are applied. Three bits are therefore needed to signal which of these eight modes to use. Table 5.1 lists the codewords for each mode. If the first two bits are zero, then the least significant bit is *0* when the existing transform from HEVC should be applied (denoted HM), and *1* indicates that the transform-skip mode (TS) from HEVC should be used. The remaining values are used to signal which edge mode to use from the subset of six possible modes.

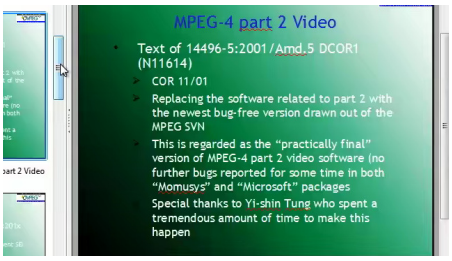
5.1.2 Experimental Results

The proposed HEVC/EM scheme was implemented using the HEVC test model reference software HM 7.0 [4]. It was tested using the intra main common test conditions [25]. The proposed HEVC/EM algorithm was tested against a reference using unmodified HM 7.0, for the four screen content sequences: *BasketballDrill-Text*, *ChinaSpeed*, *SlideEditing* and *SlideShow*. Note that transform skip (TS) [24] is not enabled for the intra main common test conditions. For comparison, we also present coding performance results for when TS is enabled. When enabled, TS or edge modes can be applied to 4x4 blocks.

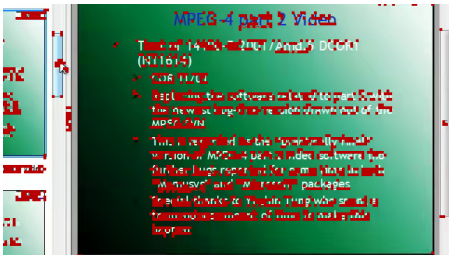
For a typical encoding run, Fig. 5.7 shows where the edge modes and TS modes are used on one frame of *SlideEditing*. Fig. 5.7(a) shows a portion of the original picture. First, only TS mode is enabled.Blocks encoded with TS are labeled in red in Fig. 5.7(b). We can see that TS mode is used over most text areas, and the conventional HM transform was used over the smooth areas. This behavior is consistent with the expectation that the energy in blocks containing very sharp transitions is spread over many transform coefficients, thus reducing the coding efficiency of the transform as compared to using no transform at all.

Fig. 5.7(c) shows mode usage results for when both TS and edge modes are enabled. Blocks using TS are marked in red, and edge-mode blocks are marked in blue. We observe that both the edge mode and TS are applied in areas containing strong edges. Moreover, many blocks marked in red in Fig. 5.7(b) are now covered by blue, which indicates that the edge modes are more efficient, in a rate-distortion sense, than TS for coding these areas.

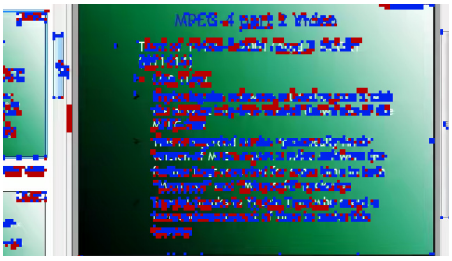
To evaluate the RD performance, four test sequences (Class F) were coded using QP values of 22, 27, 32, and 37, respectively. 500 frames were coded for sequences *BasketballDrillText*, *ChinaSpeed*, and *SlideShow* and 300 frames were coded for sequence *SlideEditing*. The performance of the original HM is used as the reference, and BD-rate [18] changes are calculated for when TS and HEVC/EM are enabled separately. To investigate the effectiveness of edge modes alone, additional experiment was carried out with TS disabled for EM, which is called EMD. The experimental results are summarized in Table 5.2. A negative change in BD-rate indicates reduction in bit-rate for the underlying method to achieve the same quality performance as the benchmark method. As shown in Table 5.2, TS and EMD are both better than the original HM with bit reduction 7.5% and xx% respectively, while the EM has the best performance, *i.e.* 10.4%, which indicates



(a) Original



(b) TS



(c) TS and EM

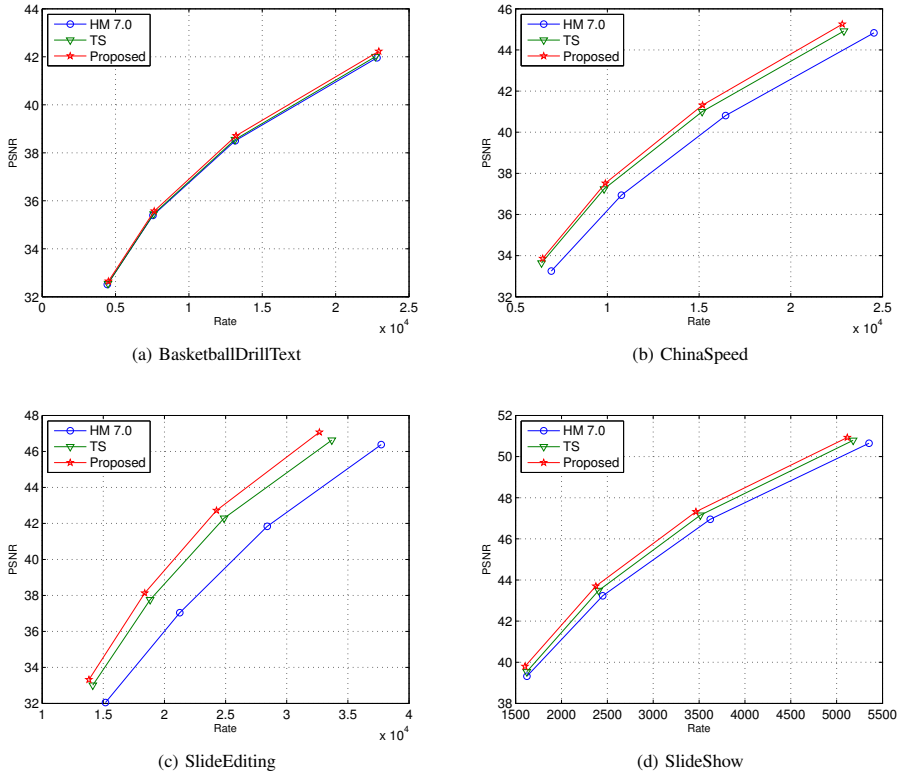
Figure 5.7: Location of blocks encoded with TS (red) or edge (blue) modes

their improvements are additive. Among the four screen content test sequences, less improvement is achieved in *BasketballDrillText*. This is reasonable given that the majority of its content is natural, while only a strip of graphics material is embedded in the sequence. HEVC/EM still outperforms TS for that sequence, as edge modes improve performance for several diagonal edges such as those found on the basketball net. RD curves for this experiment are shown in Fig. 5.8, showing that the proposed HEVC/EM scheme offers the best RD performance.

For completeness, we also investigated the performance of HEVC/EM and TS on natural video sequences, which include the class A through class E material

Table 5.2: BD-Rate changes for screen content sequences using transform skip (TS) or edge modes (EM)

	Δ BD-Rate (%)		
Sequence	**TS**	**EMD**	**EM**
BasketballDrillText	-0.7	-2.5	-2.9
ChinaSpeed	-10.5	-9.1	-13.0
SlideEditing	-14.5	-14.7	-18.0
SlideShow	-4.4		-7.7
Average	**-7.5**		**-10.4**



(a) BasketballDrillText

(b) ChinaSpeed

(c) SlideEditing

(d) SlideShow

Figure 5.8: Comparison of RD curves for HM, TS and HEVC/EM

listed in the common test conditions [25] Experimental results are summarized in Table 5.3. For classes A, B, and E, there is little or no change in RD performance using edge modes, while there is slight improvement for the lower-resolution Class C and D sequences. Although the proposed HEVC/EM require additional bits to

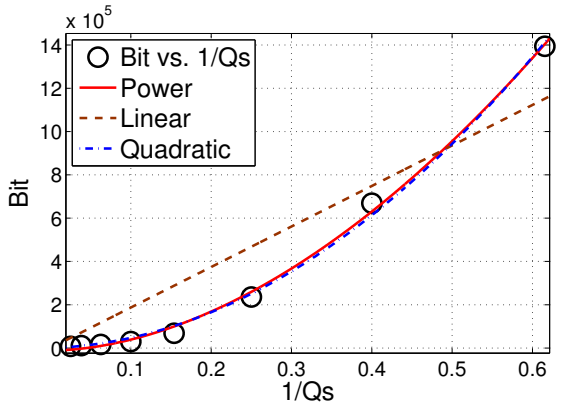
Table 5.3: BD-Rate change for different classes of natural sequences

	Δ BD-Rate (%)	
Sequence set	**TS**	**EM**
Class A	0.0	0.0
Class B	0.0	-0.1
Class C	0.0	-0.8
Class D	0.0	-1.0
Class E	0.1	0.0
Average	**0.0**	**-0.4**

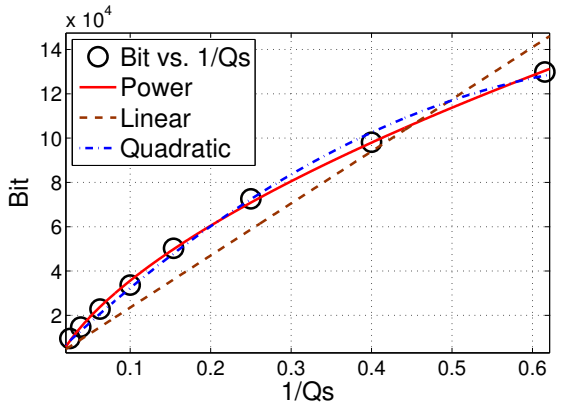
signal the modes, its coding performance for natural sequences does not decrease. Transform skip has little to no impact on performance for all classes.

5.2 Summary of Screen Content Video Coding

A new coding tool using edge modes, HEVC/EM, was introduced and integrated into the HEVC test model (HM). Several edge orientations and positions based on intra prediction directions were tested on screen content material to show that performance could be improved by partitioning a prediction block into two sub-blocks. After partitioning, a rectangular 2D DCT or a directional 2D DCT is applied to these sub-blocks. Experiments show that the proposed HEVC/EM scheme is effective in coding blocks with strong edges, yielding up to a 17.9% reduction in bit-rate and an average reduction of 10.4% for screen content video. HEVC/EM offers a significant coding gain over unmodified HM or the existing HEVC transform skip mode



(a) Texture Map



(b) Depth Map

Figure 5.9: The R-Q relationship in the texture and depth map for the sequence "Kendo".

5.3 Proposed Rate Control Schemes for 3D Video Coding

The tradeoff between the output bit rate (R) and the quality (D) of compressed video is determined by quantization step size (Q_s), which is indexed by quantization parameter (Q). The R-Q_s and D-Q_s model have been studied extensively for previous video coding standards such as MPEG-2 and H.264/AVC.

For the R-Q_s model, the classic quadratic model is developed in [29, 71] and a linear model is widely studied and applied for its simple form [83, 37, 49]. In 3DV, the R-D characteristics is different from that in previous coding standards. First, the depth map is a grey image, which has no chrominance (UV) components for YUV color space. Second, the inter-view prediction is employed to reduce the redundancy among the different views. We employ the power R-Q_s model [59, 48] for the depth and texture map as

$$R = \rho Q_s^{\tau} + c \tag{5.2}$$

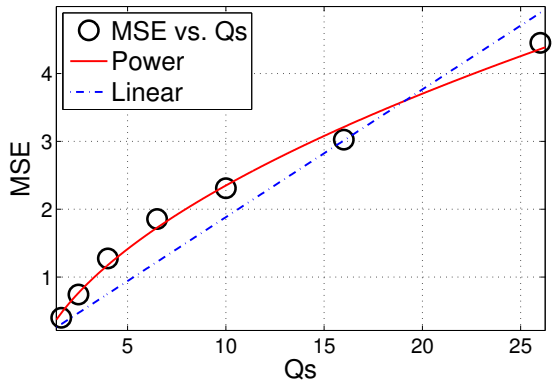
where model parameter ρ and τ depend on the video content and the sequence types (*i.e.* texture or depth map); c represents the bit to code the header information. At high bit rate, header bits usually take a small part of the total output bits, therefore we simplify (5.2) by ignoring the header bits as

$$R \approx \rho Q_s^{\tau} \tag{5.3}$$

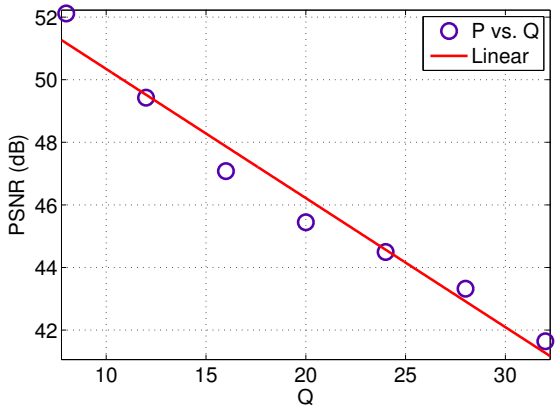
In Fig. 5.9, video sequence "Kendo" is coded with Q from 8 to 36. We can see that both the power model and the quadratic model fit the actual data well. For its simple form, the power model is adopted in our work. In Fig. 5.9, the texture and depth map exhibit different R-D characteristics. For example, parameter τ is quite different in the texture map and depth map.

In H.264/AVC and its MVC extension, Q_s and Q have nonlinear relationship, *i.e.*, Q_s double in size for every increment of 6 in Q [108]. This relationship can be approximated as

$$Q_s \approx e^{c_1 Q + c_2} \tag{5.4}$$



(a) The MSE-Q_s relation



(b) The $PSNR$-Q relation

Figure 5.10: The D-Q relationship in the texture map for the sequence "Balloons". In (a) the MSE-Q_s relationship is illustrated. In (b) the $PSNR$-Q relationship is illustrated.

where c_1 and c_2 are constants that $c_1 = \frac{1}{6}ln2$ and $c_2 = -\frac{2}{3}ln2$. Therefore R-Q relationship can be derived by substituting (5.4) into (5.3) as

$$R = \rho \cdot e^{\tau(c_1 Q + c_2)} \tag{5.5}$$

As for the D-Q model, we investigate the relationship between Q and the quality of texture map. Since the depth map will not be presented for viewing, Q of depth map will have no direct effect on view quality, but it will have indirect

influence on the quality of virtual view. Such influence will be discussed in Section 5.3.1. For the texture map, the D-Q model is adopted as

$$MSE = \chi {Q_s}^{\varphi} \tag{5.6}$$

where MSE is mean square error indicating the quality of reconstructed pictures; χ and φ are model parameters related to the video content. The MSE-Q_s relationship is illustrated in Fig. 5.10 (a), where Q varies from 6 to 32. We can see that the actual relationship can be precisely depicted by (5.6). MSE and peak signal noise ratio (PSNR) have following relationship

$$P = 10log_{10}(\frac{255^2}{MSE}) \tag{5.7}$$

where P refers to PSNR. By substituting (5.4) and (5.7), we obtain the P-Q relation as

$$P = \alpha Q + \eta \tag{5.8}$$

where $\alpha = -\frac{10}{ln10} c_1 \varphi$ and $\eta = \frac{10}{ln10}(ln\frac{255^2}{\chi} - \varphi c_2)$. The P-Q relationship in (5.8) is illustrated in Fig. 5.10 (b). As we can see, the PSNR decreases almost linearly with increase of Q value.

5.3.1 Quality Analysis for Virtual Views

At the receiver side, the virtual views can be synthesized from nearby coded views with DIBR. In this work, we investigate in the multiview video captured by parallel camera array with small intervals. When DIBR is applied at receiver side, distortion will be introduced due to the compression error in the texture and depth map. In [88], analysis on the effect of geometry distortions caused by depth coding

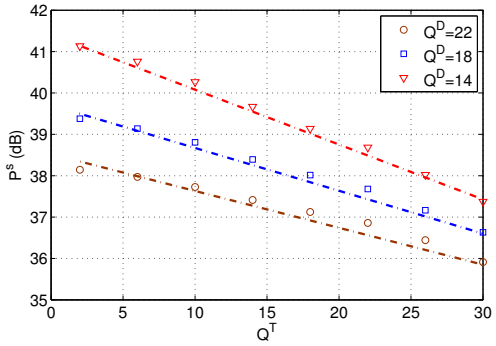


Figure 5.11: The linear relationship between the quality of virtual view and Q^T on the sequence *"Champagne_tower"*. View 37 is coded while view 38 is synthesized with DIBR. Q^T is changed from 2 to 30 when Q^D is fixed at 14, 18 and 22 respectively.

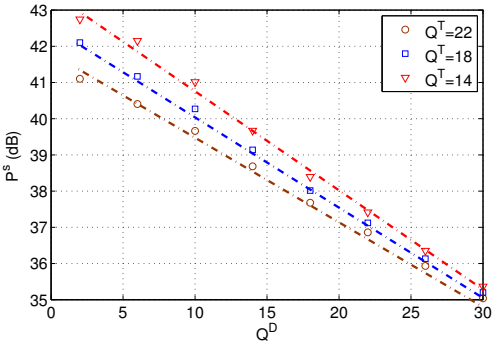


Figure 5.12: The linear relationship between the quality of virtual view and Q^D on the sequence *"Champagne_tower"*. View 37 is coded while view 38 is synthesized with DIBR. Q^D is changed from 2 to 30 when Q^T is fixed at 14, 18 and 22 respectively.

artifacts is presented. In [96], the bound of synthesis error is derived for various configurations such as depth errors. Even without compression, the distortion would be introduced by the DIBR tools. Various DIBR algorithms have been proposed to reduce the synthesis error [92, 142], however it cannot be avoided. In this work, we are only interested in issues on compression that causes distortion. We directly investigate the relationship between the quality of virtual view and Q of texture map (Q^T) or depth map (Q^D).

Since the virtual view is projected from pixel value in the texture map, its quality will be affected by the quality of decoded texture map. In Fig. 5.11, the quality influence of texture map on virtual view is investigated by changing Q^T from 2 to 30 meanwhile fixing Q^D at 14, 18 and 22 respectively. The quality of virtual view (P^s) is measured in term of PSNR. As shown in Fig. 5.11, once Q^D is determined, the P^s-Q^D relationship can be approximated as linear. Similarly in Fig. 5.12, Q^D is changed from 2 to 30, while Q^D is fixed at 14, 18, 22 respectively. We can see that linearity also can be observed between P^s and Q^D. Therefore we have the P^s-Q^T relationship as

$$\frac{\partial P^s(Q^T, Q^D)}{\partial Q^T} = \beta(Q^D) \tag{5.9}$$

and the P^s-Q^D relationship as

$$\frac{\partial P^s(Q^T, Q^D)}{\partial Q^D} = \gamma(Q^T) \tag{5.10}$$

Moreover, we can observe that the values of $\beta(Q^D)$ or $\gamma(Q^T)$ change slowly with Q^D or Q^T. Table 5.4 shows the slopes of linear P^s-Q^T relation and linear P^s-Q^D relation in Fig. 5.11 and Fig. 5.12, when the corresponding Q^D or Q^T are fixed at 14, 18, 22. In Table 5.4, the derivatives of $\beta(Q^D)$ and $\gamma(Q^T)$ ($\Delta\beta(Q^D)/\Delta Q^D$ and $\Delta\gamma(Q^T)/\Delta Q^T$), which indicate the change rate with Q^D or Q^T, are very small. Therefore for simplicity, we approximate $\beta(Q^D)$ and $\gamma(Q^T)$ as constant and approximate (5.9) as

$$\frac{\partial P^s(Q^T, Q^D)}{\partial Q^T} = \beta \tag{5.11}$$

and (5.10) as

$$\frac{\partial P^s(Q^T, Q^D)}{\partial Q^D} = \gamma \tag{5.12}$$

where β and γ are considered as constants and their values depend on the video content.

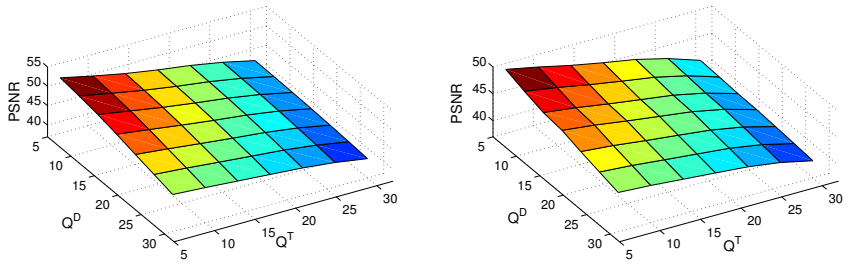
In Fig.5.13, the joint P^s-Q^T-Q^D relationship is illustrated by carrying out extensive experiments, where depth and texture maps are coded with Q from 6 to 30 respectively. The relation in Fig.5.13 can be approximated as 2D plane which indicates linear and decoupled relation between the P^s-Q^T and P^s-Q^D. For the completely decoupled linear relations, ideally the derivatives of $\beta(Q^D)$ and $\gamma(Q^T)$ should be 0. As shown in Table 5.4, the actual derivatives are very close to 0, which indicates the approximation is close to the ideal cases, thus the caused approximation error is neglected in the rest of the chapter.

Table 5.4: The slope of P^s-Q^T relation and P^s-Q^D relation

		Q=14	Q=18	Q=22	Der
Champagne	γ	-0.3076	-0.2853	-0.2636	0.0055
	β	-0.1214	-0.1088	-0.0942	0.0034
Kendo	γ	-0.0835	-0.0756	-0.0694	0.0018
	β	-0.2789	-0.2604	-0.2557	0.0029
Pantomime	γ	-0.1215	-0.1103	-0.0977	0.0030
	β	-0.3056	-0.2762	-0.2533	0.0065
Balloons	γ	-0.1293	-0.1215	-0.1095	0.0025
	β	-0.2319	-0.2180	-0.1847	0.0059

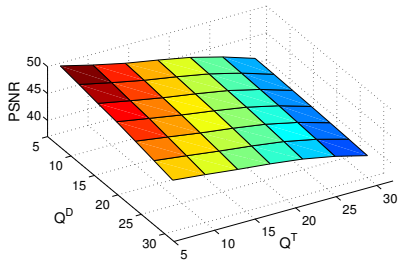
5.3.2 RDO Bit Allocation at Sequence Level

The reference relation between the coded views and the virtual views is illustrated in Fig. 5.14, where V1, V3 and V5 are coded views, and V2 and V4 are virtual views synthesized with the texture and depth maps of V1, V3 and V3, V5 respectively. Since both the coded views and virtual views will be presented for human, their

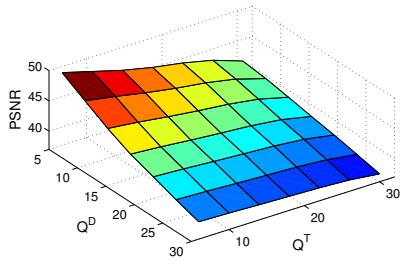


(a) *"Pantomime"*

(b) *"Balloons"*



(c) *"Kendo"*



(d) *"Champagne_tower"*

Figure 5.13: The joint relation between quality of synthesized view and Q^T and Q^D.

qualities are equally important. Due to the limitation of the transmission capacity or the storage space, the problem is how to allocate bit reasonably to optimize the overall quality performance. For convenience, in the rest of this chapter the superscripts T and D are used to indicate the texture map and depth map; the subscripts n, m and i, j stand for the view index. The optimization problem is formulated as

$$Max\left(\sum_{n\in C} P_n(Q_n^T) + \sum_{m\in S} P_m(\text{vec}Q^T, \text{vec}Q^D)\right) \tag{5.13}$$

where C is the set of the coded view index, e.g. $C = \{1, 3, 5\}$; S is the set of virtual view index, e.g. $S = \{2, 4\}$; $P_n(Q_n^T)$ and $P_m(\text{vec}Q^T, \text{vec}Q^D)$ are the quality of nth view and mth view. Since the quality of coded view is determined by the corresponding Q^T, P_n is the function of Q_n^T. For virtual view, its quality is

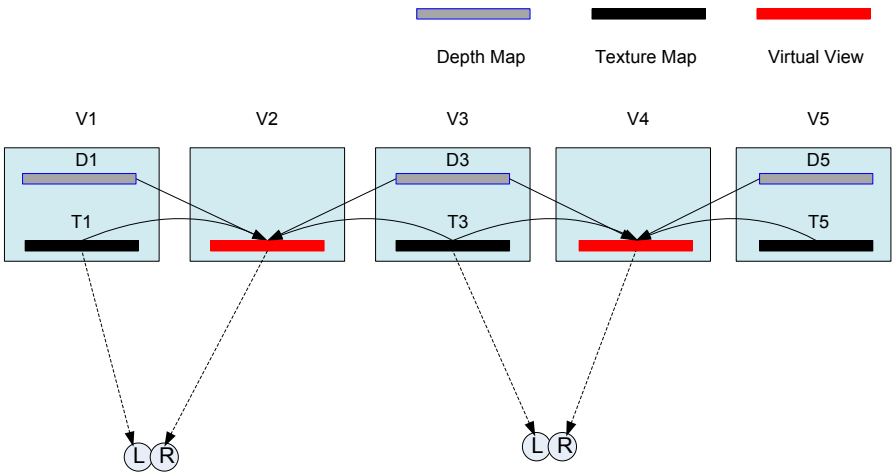


Figure 5.14: Illustration of the reference relationship of the virtual view and the coded view.

determined by both the Q^T and Q^D of the nearby coded views. For example, P_2 is determined by Q_1^T, Q_3^T and Q_1^D, Q_3^D as illustrated in Fig. 5.14. In general, P_m is function of vecQ^T and vecQ^D, where vec$Q^T = [Q_1^T, Q_3^T, \cdots]$ and vec$Q^D = [Q_1^D, Q_3^D, \cdots]$. The optimization problem is under the constraint that

$$\sum_{n \in C} \left(R_n^D(Q_n^T) + R_n^D(Q_n^D) \right) < R_{tot} \tag{5.14}$$

where R_n^T and R_n^D are the bits to code the nth texture and depth map respectively; R_{tot} is total target bits. It is obvious that the optimization problem is generally convex, since P of the coded view and virtual view have linear relationship with Q^T and Q^D as discussed in section 5.3. By applying the method of Lagrangian multiplier, we have

$$J = \left(\sum_{n \in C} P_n(Q_n^T) + \sum_{m \in S} P_m(\text{vec}Q^T, \text{vec}Q^D) \right) + \lambda \left(\sum_{n \in C} \left(R_n^T(Q_n^T) + R_n^D(Q_n^D) \right) - R_{tot} \right) \tag{5.15}$$

where λ is Lagrangian multiplier. The Q values of texture and depth map have different effects on the total quality. Q^D only affects the quality of virtual view while Q^T influences the quality of both coded views and virtual views. Therefore depending on the type of Q value (*i.e.*, Q^T or Q^D), we have different differential equations for (5.15). According to (5.8) and (5.11), the partial derivative of the first term of right side of (5.15) with respect to Q^T is derived as

$$\frac{\partial\left(\sum_{n\in C} P_n + \sum_{m\in S} P_m\right)}{\partial Q_i^T} = \alpha_i + \sum_{j\in K_i} \beta_{ij} \tag{5.16}$$

where α_i is the slope of linear P_i-Q_i^T relation in (5.8); β_{ij} is the slope of linear P_j-Q_i^T relation in (5.11); K_i is the set of virtual views whose qualities depend on ith coded view. For example in Fig. 5.14, $K_1 = \{2\}$ since the 1st view only affects the 2nd virtual view, while $K_3 = \{2, 4\}$ since the 3rd view affects the 2nd and 4th virtual views. Therefore Q^T in different positions have different effects on the total quality, and thus the right side of (5.16) varies for different views.

Meanwhile, by taking the partial derivative of the second term of right side of (5.15) with respective to Q^T, and combining with (5.5) we obtain

$$\frac{\partial R_i^T}{\partial Q_i^T} = \tau_i \cdot c_1 \cdot R_i^T \tag{5.17}$$

where R_i^T is bits for texture map. Finally, for the optimal solution, by differentiating (5.15) on both sides and replacing with (5.16) and (5.17), we get

$$0 = k_i^T + \lambda \cdot \tau_i^T \cdot c_1 \cdot R_i^{T*} \tag{5.18}$$

where $R_i^{T^*}$ is optimal bit for the texture map of ith view; k_i^T is a parameter of the texture map related to the view position that

$$k_i^T = \alpha_i + \sum_{j \in K_i} \beta_{ij}. \tag{5.19}$$

Similarly the partial derivative of the first term of right side of (5.15) with respective to Q_n^D is derived from (5.12) as

$$\frac{\partial\left(\sum_{n \in C} P_n + \sum_{m \in S} P_m\right)}{\partial Q_i^D} = \sum_{j \in K_i} \gamma_{ij} \tag{5.20}$$

where γ_{ij} is the slope of linear P_j-Q_i^D relation in (5.12). By taking the partial derivative of (5.15) with respect to Q_i^D and replacing with (5.20), we have

$$0 = k_i^D + \lambda \cdot \tau_i^D \cdot c_1 \cdot R_i^{D^*} \tag{5.21}$$

where R_i^T is the optimal bits for ith texture map; k_i^D is model parameter of the depth map in ith view that

$$k_i^D = \sum_{j \in K_i} \gamma_{ij}. \tag{5.22}$$

Therefore from (5.18) and (5.21) we can get the optimal bit allocation for both depth or texture map of ith view as

$$R_i^* = \frac{k_i/\tau_i}{\sum_{n \in C}(k_n^T/\tau_n^T + k_n^D/\tau_n^D)} R_{tot} \tag{5.23}$$

where τ_i and k_i can be the parameters for either texture map or depth map. Therefore the bit allocation scheme in (5.23) can be applied both for the texture

and depth map. In this way, optimal bit allocation among different views and among the texture and depth map are automatically achieved.

5.3.3 Model Parameter Estimation

In order to allocate bits according to (5.23), we have to access model parameter α_i, β_{ij}, γ_{ij} and τ_i before coding. Therefore the texture map is precoded at Q_A^T and Q_B^T and the depth map is precoded at Q_A^D and Q_B^D. Based on (5.8), (5.11) and (5.12), these parameters can be estimated as

$$\alpha_i = \frac{P_{iA} - P_{iB}}{Q_{iA}^T - Q_{iB}^T} \tag{5.24}$$

$$\beta_{ij} = \frac{\hat{P}_{jA} - \hat{P}_{jB}}{Q_{iA}^T - Q_{iB}^T} \tag{5.25}$$

$$\gamma_{ij} = \frac{\check{P}_{jA} - \check{P}_{jB}}{Q_{iA}^D - Q_{iB}^D} \tag{5.26}$$

where P_{iA} and P_{iB} are the PSNR of ith coded view precoded at Q_A^T and Q_B^T respectively; $\hat{P}_{j\,A}$ and $\hat{P}_{j\,B}$ are the PSNR of jth virtual view synthesized with the ith texture map precoded at Q_A^T and Q_B^T respectively; $\check{P}_{j\,A}$ and $\check{P}_{j\,B}$ are the PSNR of jth virtual view synthesized with the ith depth map precoded at Q_A^D and Q_B^D respectively.

In order to estimate these parameters, each view has to be precoded twice, which would involve heavy computation, especially when the view number is large. Usually the video contents of different views are highly similar. Thus we assume the R-D characteristics are similar in the same type of videos. To reduce the computational complexity, instead of precoding m views, only the texture and the

depth map of the first view are precoded. Then the model parameters α, β, γ are calculated according to (5.24), (5.25), (5.26) and they are used to predict the other similar parameters as

$$\alpha_i = \alpha, \quad \beta_{ij} = \beta, \quad \gamma_{ij} = \gamma \tag{5.27}$$

Meanwhile from (5.5), we can estimate τ_i for the texture map or the depth map as

$$\tau_i = \frac{ln(R_{iA}) - ln(R_{iB})}{ln(Q_{siA}) - ln(Q_{siB})} \tag{5.28}$$

where R_{iA} and R_{iB} refer to the output bits of texture map or depth map that are precoded at the corresponding Q_{siA} (Q_{iA}) and Q_{siB} (Q_{iB}). τ^T of the texture and τ^D of the depth map in the first view are used as estimation for those of other coded views.

On the other hand, the sequence complexity related parameter ρ_i needs to be estimated for the texture and depth map of each view. For the first view, ρ_1 can be estimated according to precoding result as

$$\rho_1 = \frac{R_{1A} - R_{1B}}{e^{\tau_1(c_1 Q_{1A} + c_2)} - e^{\tau_1(c_1 Q_{1A} + c_2)}} \tag{5.29}$$

where ρ_1 refers to model parameters for either the texture map or the depth map. Since R_A and R_B of other views are unavailable, a limited number of frames are encoded for the texture and depth map of each view, and the output bit rate is recorded as sample complexity r_i. ρ_i of other views is estimated as

$$\rho_i = \frac{r_i}{r_1}\rho_1 \tag{5.30}$$

In this way, we can estimate the model parameters with reduced computational complexity.

5.3.4 Frame Level Bit Regulation

Given the total bit rate constraint, the optimal target bit rate (R_i^*) for each texture or depth map of each view can be calculated according to (5.23). To achieve the target bit of the each sequence, two RC schemes can be applied. One is to apply RC at frame level (FL) to adjust Q dynamically along the sequence to achieve the target bit rate. The other is to adopt a constant Q (CQ) to code the entire sequence. Since the fluctuation in Q usually degrade the R-D performance, the CQ usually has better R-D performance than FL. On the other hand, FL has more accurate target bit rate achievement due to the adaptive adjustment of Q value.

In this work, we adopt both the FL and the CQ schemes to achieve the target bit. For the FL scheme, bit allocation algorithm in [69] is used to allocate target bit (R_t) at frame level and the corresponding Q_s for the coding frame is calculated based on (5.3) as

$$Q_s = \left(\frac{R_t}{\rho}\right)^{1/\tau} \tag{5.31}$$

Then the corresponding Q can be attained with the Q-Q_s relation.

For the CQ, with the target bit of the sequence, the Q can be calculated based on (5.5) as

$$Q = \frac{ln(R^*/\rho) - c_2\tau}{c_1\tau} \tag{5.32}$$

where R^* is the target bit; ρ and τ are the estimated model parameters. Since we have already accessed the R-D characteristics of the sequence and estimated these model parameters, the calculated Q lead to the achievement of the target bit.

Table 5.5: Model parameter value and the corresponding estimation

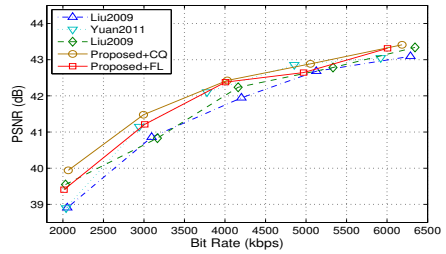
		Newspaper			Champagne_tower			Balloon			Mobile		
		Actual	**Est**	**E**	**Actual**	**Est.**	**E**	**Actual**	**Est.**	**E**	**Actual**	**Est.**	**E**
α	α_2	-0.515	-0.512	0.6	-0.406	-0.405	0.1	-0.413	-0.434	5.3	-0.739	-0.736	0.4
	α_3	-0.535	-0.512	4.3	-0.420	-0.405	3.5	-0.405	-0.434	7.2	-0.746	-0.736	1.3
β	β_{32}	-0.240	-0.242	0.8	-0.112	-0.104	7.0	-0.255	-0.258	1.4	-0.525	-0.514	2.1
	β_{34}	-0.250	-0.242	3.4	-0.098	-0.104	6.3	-0.252	-0.258	2.6	-0.519	-0.514	1.0
	β_{54}	-0.263	-0.242	8.0	-0.105	-0.104	1.1	-0.256	-0.258	0.9	-0.541	-0.514	5.0
γ	γ_{32}	-0.167	-0.151	9.5	-0.251	-0.256	2.1	-0.102	-0.107	5.4	-0.143	-0.155	8.5
	γ_{34}	-0.144	-0.151	4.9	-0.278	-0.256	7.8	-0.096	-0.107	11.5	-0.146	-0.155	5.6
	γ_{54}	-0.167	-0.151	9.7	-0.281	-0.256	9.0	-0.101	-0.107	6.9	-0.132	-0.155	17.1
τ^T	τ_2^T	-1.126	-1.073	4.7	-1.370	-1.314	4.1	-1.103	-1.045	5.3	-1.137	-1.093	3.9
	τ_3^T	-1.164	-1.073	7.8	-1.396	-1.314	5.9	-1.113	-1.045	6.2	-1.187	-1.093	7.9
τ^D	τ_2^D	-1.048	-1.010	3.6	-0.832	-0.922	10.8	-0.956	-0.993	3.8	-0.727	-0.721	0.9
	τ_3^D	-0.970	-1.010	4.1	-0.904	-0.922	1.9	-1.027	-0.993	3.3	-0.629	-0.721	14.6
ρ	ρ_2^T	19435	21041	8.3	50493	48792	3.4	23864	23637	1.0	6438	6618	2.8
	ρ_3^T	22032	21991	0.2	60480	57388	5.1	25290	25786	2.0	8040	7596	5.5
Average				**5.0**			**4.9**			**4.5**			**5.5**

5.3.5 Experimental Results

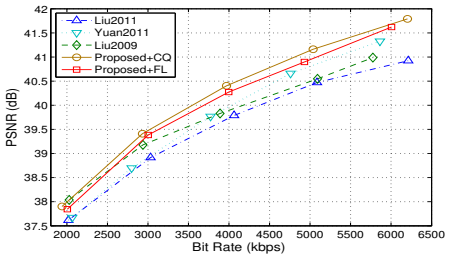
The experiments are conducted under 5-view scenario, where 3 views are coded, and 2 virtual views are synthesized. The testing sequences include *"Champagne_tower"* (1280×960), *"Balloons"* (1024×768) provided by Nagoya University [5], and *"Newspaper"* (1024×768) provided by Gwangju Institute of Science and Technology (GIST) [3]. The texture and depth map of three views are separately encoded with MVC encoder [9] as I-view, P-view and P-view respectively. For P-view, the interview prediction is only applied for key frames. View Synthesis Reference Software (VSRS) [10] is used to synthesize the virtual view. 201 frames are encoded for each view.

Verification of Parameter Estimation Scheme

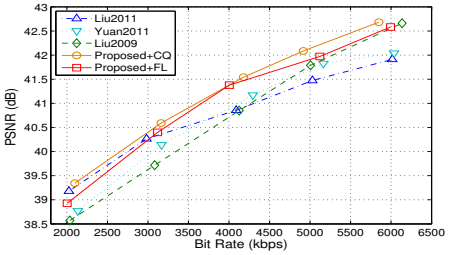
In this section, we verify the effectiveness of the proposed parameter estimation scheme in Section 5.3.3. In the experiments, α, β, γ are estimated according to (5.27), and ρ is estimated according to (5.30), and τ^T and τ^D are estimated based on (5.28).



(a) *"Balloons"*



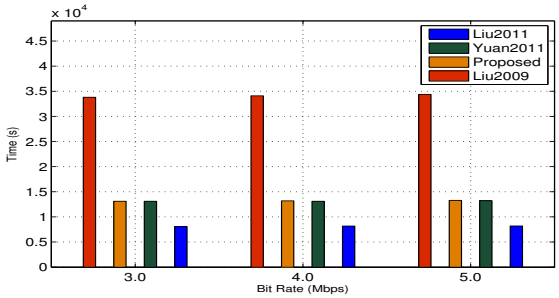
(b) *"Newspaper"*



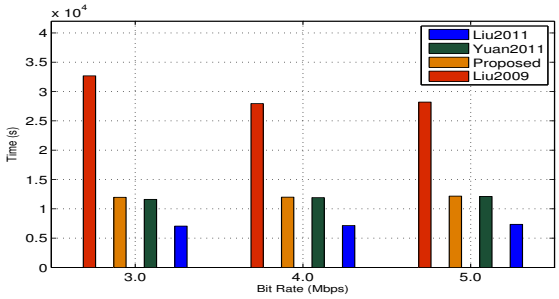
(c) *"Champagne_tower"*

Figure 5.15: The R-D curves. The autostereoscopic 3D video is set to 5-view scenario, where 3 views are coded views and 2 views are virtual views. The three coded views are coded with MVC codec as I-view, P-view, P-view respectively. Search range is set to 96 with GOP size 4. Target bits are set at 2.0 Mbps, 3.0 Mbps, 4.0 Mbps, 5.0 Mbps and 6.0 Mbps and the corresponding R-D points are depicted for each algorithm.

The results and the estimation errors are presented in Table 5.5. We can see that the estimation is accurate enough that the mismatch is less than 5.5% on overage, which indicates the proposed scheme can achieve accurate model parameter estimation.



(a) *"Balloons"*



(b) *"Newspaper"*

Figure 5.16: The consumed coding time. The autostereoscopic 3D video is set to 5-view scenario, where 3 views are coded views and 2 views are virtual views. The three coded views are coded with MVC codec as I-view, P-view, P-view respectively. Search range is set to 96 with GOP size 4. Target bits are set at 3.0 Mbps, 4.0 Mbps and 5.0 Mbps for each algorithm.

R-D Performance and Rate Accuracy

In order to evaluate the performance of the proposed RC algorithm, Liu2011 [80], Yuan2011 [138] and Liu2009 [79] are utilized for comparison. For the proposed algorithm, both the FL (proposed+FL) and the CQ (proposed+CQ) are employed to achieve the target bit for each sequence. Table 5.6, 5.7 and 5.8 summarize the output bits of the coded texture and the depth map of each view. Since the virtual views are generated with DIBR, for V2 and V4 in Table 5.6 and V3 and V5 in Table 5.7, V38 and V40 in Table 5.8, there are no output bits for the texture and

Table 5.6: Result summary of different RC algorithms on the sequence *"Balloons"*

T (Mbps)	V	Liu2011 Rate (kbps) T	D	P (dB)	E (%)	Yuan2011 Rate (kbps) T	D	P (dB)	E (%)	ΔP (dB)	Liu2009 Rate (kbps) T	D	P (dB)	E (%)	ΔP (dB)	Proposed+CQ Rate (kbps) T	D	P (dB)	E (%)	ΔP (dB)	Proposed+FL Rate (kbps)	P (dB)	E (%)	ΔP (dB)
	V1	832	211	41.13		718	265	41.31			642	412	40.79			725	129	41.36			759 127	41.16		
	V2			40.71				41.09					41.06					40.96				40.75		
3	**V3**	811	203	41.05	**3.1**	692	244	41.24	**2.0**	**0.29**	619	374	40.74	**5.5**	**-0.03**	1006	166	42.51	**0.2**	**0.62**	1065174	42.60	**0.3**	**0.36**
	V4			40.58				41.06					41.02					40.82				40.59		
	V5	828	210	40.82		712	309	41.05			635	484	40.54			865	103	41.74			759 125	40.96		
	V1	1126	281	42.27		908	353	42.30			913	472	42.33			1015	173	42.57			1013167	42.46		
	V2			41.74				42.06					42.34					41.88				41.71		
4	**V3**	1106	269	42.12	**5.1**	880	321	42.19	**5.6**	**0.16**	885	426	42.22	**4.0**	**0.29**	1311	232	43.33	**0.7**	**0.48**	1418231	43.36	**0.2**	**0.43**
	V4			41.62				42.00					42.28					41.72				41.57		
	V5	1143	278	41.98		903	413	41.98			909	557	42.02			1149	148	42.63			1011166	42.80		
	V1	1381	346	43.09		1164	450	43.09			1059708		42.81			1178	226	43.16			1265207	43.19		
	V2			42.49				42.81					43.04					42.56				41.62		
5	**V3**	1344	336	42.89	**2.5**	1136	406	42.93	**2.9**	**0.18**	1029633		42.67	**6.7**	**0.11**	1814	288	43.92	**1.1**	**0.21**	1778288	43.90	**0.6**	**-0.03**
	V4			42.31				42.76					42.95					42.33				41.71		
	V5	1371	349	42.62		1169	530	42.71			1060844		42.45			1366	182	42.45			1266166	42.80		
Average:					**3.6**				**3.5**	**0.21**				**5.4**	**0.12**				**0.7**	**0.43**			**0.4**	**0.25**

Table 5.7: Result summary of different RC algorithms on the sequence *"Newspaper"*

T (Mbps)	V	Liu2011 Rate (kbps) T	D	P (dB)	E (%)	Yuan2011 Rate (kbps) T	D	P (dB)	E (%)	ΔP (dB)	Liu2009 Rate (kbps) T	D	P (dB)	E (%)	ΔP (dB)	Proposed+CQ Rate (kbps) T	D	P (dB)	E (%)	ΔP (dB)	Proposed+FL Rate (kbps)	P (dB)	E (%)	ΔP (dB)
	V2	885	227	39.96		577	377	39.31			670	329	39.93			720	215	40.22			700 201	40.01		
	V3			37.64				37.92					38.11					38.12				37.54		
3	**V4**	798	209	39.56	**1.0**	540	400	38.95	**6.7**	**-0.22**	634	347	39.57	**2.1**	**0.26**	831	342	40.59	**2.3**	**0.50**	918 281	40.91	**0.1**	**0.47**
	V5			38.46				38.67					38.99					38.96				39.01		
	V6	722	191	38.96		550	354	38.64			648	310	39.29			623	201	39.16			702 201	39.45		
	V2	1152	304	40.90		741	538	40.37			748	569	40.42			963	309	41.30			934 267	41.13		
	V3			38.30				38.95					39.00					38.92				38.68		
4	**V4**	1088	279	40.54	**1.6**	702	575	40.00	**5.7**	**-0.02**	710	610	40.06	**2.7**	**0.04**	1094	488	41.57	**0.7**	**0.61**	1221374	41.76	**0.1**	**0.48**
	V5			39.26				39.78					39.88					39.95				39.47		
	V6	987	253	39.94		722	495	39.74			730	523	39.79			846	270	40.28			934 267	40.33		
	V2	1498	373	41.75		970	631	41.34			882	830	41.04			1134	375	41.94			1167334	41.97		
	V3			38.82				39.61					39.92					39.51				39.14		
5	**V4**	1347	347	41.32	**1.7**	932	680	40.98	**4.8**	**0.19**	844	905	40.67	**1.8**	**0.08**	1487	599	42.47	**0.8**	**0.69**	1527466	42.48	**1.4**	**0.43**
	V5			39.89				40.64					40.69					40.71				40.57		
	V6	1208	313	40.58		968	579	40.73			875	757	40.42			1099	345	41.18			1170267	40.33		
Average					**1.4**				**5.7**	**-0.02**				**2.2**	**0.13**				**1.3**	**0.60**			**0.5**	**0.46**

Table 5.8: Result summary of different RC algorithms on the sequence *"Champagne_tower"*

T (Mbps)	V	Liu2011 Rate (kbps) T	D	P (dB)	E (%)	Yuan2011 Rate (kbps) T	D	P (dB)	E (%)	ΔP (dB)	Liu2009 Rate (kbps) T	D	P (dB)	E (%)	ΔP (dB)	Proposed+CQ Rate (kbps) T	D	P (dB)	E (%)	ΔP (dB)	Proposed+FL Rate (kbps)	P (dB)	E (%)	ΔP (dB)
	38	891	240	41.48		567	603	40.29			521	720	39.96			628	331	40.78			617 292	40.60		
	39			39.05				40.35					39.79					39.90				39.47		
3	**40**	640	171	41.20	**0.6**	518	471	40.39	**5.6**	**-0.12**	474	388	40.05	**2.8**	**-0.55**	761	479	41.92	**5.4**	**0.33**	769 410	41.82	**3.9**	**0.13**
	41			38.60				39.82					39.31					39.87				39.44		
	42	823	217	40.99		544	464	39.85			525	455	39.48			645	318	40.48			620 410	40.66		
	38	1239	315	42.27		713	1035	41.22			646	1051	40.93			819	433	41.67			823 390	41.68		
	39			39.59				41.38					41.11					40.84				40.46		
4	**40**	896	233	41.83	**2.2**	649	571	41.37	**7.4**	**0.31**	590	534	41.09	**3.2**	**0.00**	1003	630	42.69	**4.5**	**0.69**	1020545	42.73	**0.1**	**0.52**
	41			39.12				41.04					40.54					41.08				40.53		
	42	1121	285	41.47		731	595	40.84			658	647	40.58			856	438	41.43			837 389	41.49		
	38	1494	397	42.98		832	1292	41.77			852	1182	41.89			936	547	42.16			1035487	42.33		
	39			40.10				42.13					41.83					41.44				41.14		
5	**40**	1107	292	42.52	**0.6**	759	636	41.93	**3.3**	**0.35**	777	590	42.06	**0.1**	**0.31**	1178	707	43.13	**1.7**	**0.61**	1303681	43.30	**2.4**	**0.50**
	41			39.56				41.84					41.55					41.73				41.06		
	42	1383	356	42.21		862	784	41.46			884	721	41.61			992	557	41.97			1033583	42.03		
Average					**1.1**				**5.4**	**0.18**				**2.0**	**-0.08**				**3.8**	**0.54**			**2.1**	**0.38**

depth map. The ratio of output bits of the texture and depth map is fixed close to 4:1 for Liu2011.

To evaluate the accuracy of the bit rate achievement, the following measurement is adopted

$$E = \frac{|R_{all} - R_{target}|}{R_{target}} \times 100\% \tag{5.33}$$

where R_{all} is the total bits used to encode the depth map and texture map of three views; R_{target} is the target bit rate. Table 5.6, 5.7 and 5.8 present the rate achievement accuracy of different algorithms. We can see the proposed+FL generally has the best performance that its mismatch is 0.4 %, 0.5 %, 2.1 % on average for different sequences. The proposed+CQ also achieves acceptable accuracy, i.e. 0.7 %, 1.3 % and 3.8 % on average.

The PSNR of both coded and virtual views are recorded for each view. The average PSNR is used to evaluate the overall quality performance of five views for different algorithms. The average PSNR of Liu2011 is set as the benchmark and the performances of other algorithms are measured as

$$\Delta P = P_i - \hat{P} \tag{5.34}$$

where $\hat{P}$ refers to the average PSNR of Liu2011 and P_i refers to the average PSNR of the rest algorithms. The results are presented in Table 5.6 5.7 and 5.8, where we can see the proposed+CQ achieves the best performance that the average PSNR gains are 0.43 dB, 0.60 dB and 0.54 respectively, while the proposed+FL has little degradation in R-D performance achieving 0.25 dB, 0.46 dB and 0.38 dB gain respectively.

For further illustration, typical R-D curves for different algorithms are shown in Fig. 5.15, where we can see the proposed+CQ demonstrates the best R-D efficiency among the different algorithms.

The computational complexity is compared for different algorithms.The computation complexity mainly comes from view coding and view synthesis process. For Liu2011, each view including the texture and depth map is coded with single pass, while for the proposed algorithm and Yuan2011, two additional iterations are required for the first view. For Liu2009, the texture maps of each view have to be coded for M times and the depth maps have to be code for three times, where M is the number of proper Q values which generate bit rate falling into the range $[1/2R_t, R_t]$. As for view synthesis, Liu2011 has to synthesize two virtual views, while for the proposed method and Yuan2011 three more virtual views need to be synthesized to assist model parameter calculation. For the Liu2009, $3M$ more virtual view synthesis are required to calculate the model parameters. Therefore, Liu2011 consumes the lowest computation complexity. Although the proposed algorithm requires additional computation in the precoding stage, with the increase of view number, the portion of the additional complexity in the total complexity will decrease. The actual consuming time for each algorithm is presented in Fig. 5.16, where we can see the proposed algorithm takes less time than Liu2009 and is comparable with Yuan2011.

5.4 Summary of 3D Video Coding

In this chapter, we proposed a RC scheme to achieve the best overall quality for 3DV. Based on power models for the R-Q_s and the MSE-Q_s relationship, we derived the exponential R-Q relationship and the linear $PSNR$-Q relationship.

Furthermore, a linear model is approximated for the quality dependency between the virtual view and the coded view. Based on the above R-D characteristics of both the coded view and the virtual view, a R-D optimized RC algorithm is derived. Experiments are conducted on different video sequences and the results demonstrate the effectiveness of the proposed algorithm.

Chapter 6

Conclusion and Future Work

6.1 Conclusion of the Research

Quality assessment and compression are two important parts of image and video processing system. This thesis includes researches on quality assessment and video coding techniques.

Different quality metrics are developed for images and videos. First, methods based on distortion grouping and content grouping are proposed. Since image and video quality is affected by both distortion types and contents, by proper grouping, the quality assessment problem is decoupled into simple problems where only single factor needs to be considered. Second, the chacteristics of human visual system are considered and PW-MSE is proposed for both image quality assessment and video quality assessment in chapter 4 and chapter 5 respectively. In PW-MSE, the masking effect as well as the low-passing filter characteristics of the initial process of HVS is explored. Since the perception on videos is quite different from images, the masking model and CSF in video is different from image and it is revised acccordingly in the proposed PW-MSE for videos. The experimental results on each steps verify the effectiveness of both masking and CSF models. PW-MSE for both images and videos outperforms other benchmark algorithms.

As for the video coding techniques, two different video formats are considered in this thesis, i.e., screen content and 3D video. For screen content videos, different edge modes are proposed to adapt to sharp edges in screen content. Significant

improvement has been achieved as comparing to the latest standard codec, i.e. HEVC. In 3D video coding, a novel bit allocation scheme is proposed to optimize the RD performance among different views and between texture and depth maps within a view. The experimental results show significant RD gains when it is evaluated with standard codec.

6.2 Future Work

The latest video coding standard, High Efficiency Video Coding (HEVC)[118], has improved the Rate-Distortion (RD) performance significantly as compared with the previous video coding standards such as MPEG-2 and H.264/AVC [133]. On the other hand, being similar to the development of previous standards, its performance optimization has not yet taken the characteristics of the Human Visual System (HVS) into account. Since the ultimate recipient of any video playback system is human being, it is desirable to develop video coding tools that optimize the perceptual quality under the bit rate constraint. Several HVS models and perceptual quality metrics have been developed by investigating different characteristics of the HVS such as contrast sensitivity, luminance adaptation , and perceptual masking. The integration of the HVS model and HEVC video coding tools is still an open problem and it demands further research efforts. For example, rate control is an important video coding tool that affects the overall video coding performance significantly. So far, rate control has not been designed by considering the perceptual RD optimization, and a perceptual RD optimized rate control is highly in demand.

In future, we will investigate a new Rate-Distortion Optimization (RDO) method, where the Mean Squared Error (MSE) distortion is replaced by a recently

developed perceptual distortion metric called the Perceptually Weighed MSE (PW-MSE), with an objective to optimize the perceptual RD performance of the HEVC encoder. A perceptual preprocessing method will be developed to improve the perceptual RD performance and a new perceptual rate control scheme will be proposed to allocate bits adaptively for best perceptual RD performance.

6.2.1 Perceptually Optimized Video Coding

To achieve high coding efficiency, HEVC employs the Lagrange method to compute the RD cost of different encoding parameter settings and selects the one with the minimum RD cost. The Lagrangian RD cost can be written in form of:

$$J = D + \lambda R \tag{6.1}$$

where D is the encoding distortion,R is the total number of bits used to encode the header, motion vector, quantized coefficients etc., and λ is the Lagrangian multiplier. The Lagrangian multiplier is defined in HEVC as

$$\lambda = \alpha \cdot 2^{\frac{Qp-12}{3}} \tag{6.2}$$

where Qp is the quantization parameter and ęÁ is a constant determined empirically by extensive experiments. Since the distortion is measured by the MSE, the selected optimal encoding parameter setting is not optimal in terms of perceptual quality. Moreover, although the Lagrangian multiplier λ is critical in the RDO optimization in HEVC, it is only adaptive with quantization parameter Qp, without considering the characteristics of input video content. To improve the above framework, we will develop an RDO scheme by introducing a new distortion metric and an adaptive Lagrangian multiplier in this task.

The new distortion metric PW-MSE developed in our previous work will be employed to replace the traditional MSE in the RDO process. The PW-MSE measure modifies the MSE measure by taking the spatial and temporal masking effects into account. It has been verified that the PW-MSE has a more accurate prediction of perceptual distortion in various image and video databases. The modified RD cost can be expressed as

$$J = PWMSE + \lambda \cdot R \tag{6.3}$$

Due to the masking effect, the same amount of distortion under different background may have different perceptual impact on human observers. For example, the complex texture region can mask more distortion than the smooth region. This implies that the Lagrangian multiplier should be adaptive to video content. To proceed along this line, we need to analyze the complexity of video content. Without introducing additional computational complexity, we plan to use AC coefficients obtained after DCT transform in HEVC to characterize the spatial complexity. Typically, larger AC coefficients implies content of high complexity. Besides, the temporal complexity can be analyzed using motion vectors in HEVC.

Based on video content characteristics, a Lagrangian multiplier can be designed. Intuitively, the Lagrangian multiplier should be larger in the complex texture region than in the smooth region. Because the distortion in the complex texture region is less important than that in the smooth region. We will find a quantitative model to compute the Lagrangian multiplier by taking the video content complexity into consideration.

The captured video content may contain some signal types that are of little value in preserving the fidelity of captured scenes. One type is the noise introduced

in the video acquisition process. Noise is not the target signal while its coding tends to increase the bit rate and decrease the perceptual quality. Another type is the high frequency signal imperceptible by human. Its removal will not reduce the perceptual quality but save encoding bits. Thus, it is desired to design proper filters to remove noise and imperceptible details. The contrast sensitivity function (CSF) that describes the frequency characteristics of the HVS can be used to design the filters. Furthermore, the frequency of a signal projected onto the retina can be affected by various factors such as the viewing distance, the display pixel density, etc. The filter design should consider all relevant factors to guarantee good perceptual quality at different viewing conditions.

6.2.2 Perceptual Rate Control

The goal of rate control is to regulate the encoded bit stream so as to achieve the best video quality without violating the constraints imposed on the encoder/decoder buffer size and the available channel bandwidth. To achieve R-D optimization, one has to build the rate-quantization (R-Q) model that characterizes the relationship between rate and Qp and the distortion-quantization (D-Q) model that characterizes the relationship between distortion and Qp. By replacing the traditional MSE distortion with the new PW-MSE distortion in the perceptually optimized encoder, the previous R-Q and D-Q models will not be accurate and, as a result, they will lead to a poor rate control performance in the new encoder. Thus, in the second task, we have to build up accurate R-Q and D-Q models first.

The traditional R-Q and D-Q models have been extensively studied in the literature before. In [16], a classical D-Q model was derived in form of

$$D = \chi Q_{step}^2 \tag{6.4}$$

where Q_{step} is the quantization step, D is measured in terms of MSE and parameter χ has a typical value of 1/12. A linear D-Q model was proposed in [58] to relate the PSNR quality measure and Qp as

$$PSNR = \rho Qp + \upsilon \tag{6.5}$$

where ρ and *upsilon* are two model parameters. For the new PW-MSE distortion measure, the exact relationship between D and the quantization parameter (or the quantization step size) is still unclear. To tackle it, we will conduct experiments so as to collect the distortion data and the corresponding quantization parameters under various test conditions. The D-Q relationship can be affected by various factors. Among them, the most influential ones are video content and the video coding method. In this study, we will analyze the impact of different video contents and coding settings on the D-Q relationship. In this way, proper model parameters of the D-Q model could be determined for the HEVC video coding method.

With the assumption that residual coefficients are Laplacian distribution, the following quadratic R-Q model was proposed in [29]:

$$R = \frac{a \cdot m}{Q_{step}} + \frac{b \cdot m}{Q_{step}} + c \tag{6.6}$$

where m is the Mean Absolute of Difference (MAD) of residuals between the original and predicted signals and a, b and c are model parameters. The classical quadratic model has been adopted as a non-normative RC tool in MPEG-4, H.264/AVC and HEVC. It is unclear whether such a relation would still remain in the proposed perceptual RDO encoder. Experiments based on the new perceptual video encoder will be conducted and a new R-Q relationship has to be derived accordingly.

Since the relationship between the perceptual quality and the bit rate of each block of a frame is highly nonlinear, it is different from image to image and block to block. Given a constraint on the total bit rate, equally distributing bits will not offer the best quality. In order to achieve optimal perceptual quality, the bit rate should be carefully allocated according to the RD characteristics. Using the new R-Q and D-Q models, we could obtain the best bit allocation for optimal perceptual RD performance. Based on the best bit allocation result, proper quantization parameters can be calculated using the R-Q model.

Bibliography

[1] Final report from the video quality experts group on the validation of objective models of video quality assessment, phase i.

[2] Final report from the video quality experts group on the validation of objective models of video quality assessment, phase i.

[3] Gist, ftp://203.253.128.142.

[4] Hm 7.0 software, available online `https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/`.

[5] http://www.tanimoto.nuee.nagoya-u.ac.jp/mpeg/mpeg_ftv.html.

[6] Toyoma database.

[7] Methodology for the subjective assessment of the quality of television pictures. In *ITU-R Recommendation BT.500-11*, 2002.

[8] Epfl-polimi video quality assessment database. In *P[Online]. Available: http://vqa.como.polimi.it/*, accessed in Oct. 2015.

[9] Joint multiview video coding jmvc 8.5 software package. In *CVS Server for the JMVC Software*, Mar. 2011.

[10] Report on experimental framework for 3d video coding. In *Doc. N11631, Guangzhou, China*, October 2010.

[11] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk. Frequency-tuned salient region detection. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1597–1604, June 2009.

[12] Peter GJ Barten. *Matrix Analysis*. Cambridge University Press, 1985.

[13] Peter GJ Barten. *Contrast sensitivity of the human eye and its effects on image quality*, volume 72. SPIE press, 1999.

[14] Debasish Basak, Srimanta Pal, and Dipak Chandra Patranabis. Support vector regression. *Neural Information Processing-Letters and Reviews*, 11(10):203–224, 2007.

[15] Philip Benzie, John Watson, Phil Surman, Ismo Rakkolainen, Klaus Hopf, Hakan Urey, Ventseslav Sainov, and Christoph Von Kopylow. A survey of 3dtv displays: techniques and technologies. *Circuits and Systems for Video Technology, IEEE Transactions on*, 17(11):1647–1658, 2007.

[16] T. Berger. *Rate-Distortion Theory: A Mathematical Basis for Data Compression*. NJ: Prentice-Hall, 1971.

[17] A. Bhat, S. Kannangara, Yafan Zhao, and I. Richardson. A full reference quality metric for compressed video based on mean squared error and video content. 22(2):165–173, Feb 2012.

[18] G. Bjontegaard. Calculation of average PSNR difference between RD-curves. *Document of Joint Collaborative Team on Video Coding*, pages VCEG–M033, 2001.

[19] Byron Boots, Geoffrey J. Gordon, and Sajid M. Siddiqi. A constraint generation approach to learning stable linear dynamical systems. *Advances in Neural Information Processing Systems 20*, pages 1329–1336, 2008.

[20] A. Borji and L. Itti. State-of-the-art in visual attention modeling. 35(1):185–207, Jan 2013.

[21] Andrew P Bradley. A wavelet visible difference predictor. 8(5):717–730, 1999.

[22] A. Bugeau and P. Perez. Detection and segmentation of moving objects in highly dynamic scenes. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pages 1–8, June 2007.

[23] Christina A. Burbeck and D. H. Kelly. Estimating just-noticeable distortion for video. *J. Opt. Soc. Am.*, 70:1121–1126, 1980.

[24] J. Xu C. Lan, X Peng and F. Wu. Intra transform skipping,. *Document of Joint Collaborative Team on Video Coding*, pages JCTVC–I0408, 2004.

[25] J. Xu C. Lan, X. Peng and F. Wu. Common test conditions and software reference configurations. *Document of Joint Collaborative Team on Video Coding*, pages JCTVC–H1100, Feb. 2012.

[26] Damon M Chandler and Sheila S Hemami. Effects of natural images on the detectability of simple and compound wavelet subband quantization distortions. 20(7):1164–1180, 2003.

[27] Damon M Chandler and Sheila S Hemami. VSNR: A wavelet-based visual signal-to-noise ratio for natural images. 16(9):2284–2298, 2007.

[28] Hui Cheng and Charles A Bouman. Document compression using rate-distortion optimized segmentation. *Journal of Electronic Imaging*, 10(2):460–474, 2001.

[29] Tihao Chiang and Ya-Qin Zhang. A new rate control scheme using quadratic rate distortion model. *Circuits and Systems for Video Technology, IEEE Transactions on*, 7(1):246–250, 1997.

[30] S. Chikkerur, V. Sundaram, M. Reisslein, and L.J. Karam. Objective video quality assessment methods: A classification, review, and performance comparison. 57(2):165–182, June 2011.

[31] Susana TL Chung, Gordon E Legge, and Bosco S Tjan. Spatial-frequency characteristics of letter identification in central and peripheral vision. *Vision Res.*, 42(18):2137–2152, 2002.

[32] D. Comaniciu, V. Ramesh, and P. Meer. Kernel-based object tracking. 25(5):564–577, May 2003.

[33] S. Daly. The visible difference predictor: an algorithm for the assessment of image fidelity. In *Proc. SPIE Human Vision, Visual Process. Digital Display III*, volume 1666, pages pp. 2ĺC15„ 1992.

[34] Scott J Daly. Visible differences predictor: an algorithm for the assessment of image fidelity. In *SPIE/IS&T 1992 Symposium on Electronic Imaging: Science and Technology*, pages 2–15. International Society for Optics and Photonics, 1992.

[35] Ismael Daribo, Christophe Tillier, and Béatrice Pesquet-Popescu. Motion vector sharing and bitrate allocation for 3d video-plus-depth coding. *EURASIP Journal on Applied Signal Processing*, 2009:3, 2009.

[36] Francesca De Simone, Lutz Goldmann, Vittorio Baroncini, and Touradj Ebrahimi. Subjective evaluation of jpeg xr image compression. In *SPIE Optical Engineering+ Applications*, pages 74430L–74430L. International Society for Optics and Photonics, 2009.

[37] Jianpeng Dong and Nam Ling. A context-adaptive prediction scheme for parameter estimation in h. 264/avc macroblock layer rate control. *Circuits and Systems for Video Technology, IEEE Transactions on*, 19(8):1108–1117, 2009.

[38] Karen Egiazarian, Jaakko Astola, Nikolay Ponomarenko, Vladimir Lukin, Federica Battisti, and Marco Carli. New full-reference quality metrics based on hvs. In *CD-ROM proceedings of the second international workshop on video processing and quality metrics, Scottsdale, USA*, volume 4, 2006.

[39] U. Engelke, H. Kaprykowsky, H. Zepernick, and P. Ndjiki-Nya. Visual attention in quality assessment. 28(6):50–59, Nov 2011.

[40] Christoph Fehn. Depth-image-based rendering (dibr), compression, and transmission for a new approach on 3d-tv. In *Electronic Imaging 2004*, pages 93–104. International Society for Optics and Photonics, 2004.

[41] John M Foley and Geoffrey M Boynton. New model of human luminance pattern vision mechanisms: analysis of the effects of pattern orientation, spatial phase, and temporal frequency. In *Computational Vision Based on Neurobiology*, pages 32–42. International Society for Optics and Photonics, 1994.

[42] Wilson S. Geisler and Jeffrey S. Perry. Real-time foveated multiresolution system for low-bandwidth video communication. *Proc. SPIE*, 3299:294–305, 1998.

[43] VIDEO QUALITY EXPERTS GROUP. Report on the validation of video quality models for high definition video content. 2010.

[44] Chenlei Guo, Qi Ma, and Liming Zhang. Spatio-temporal saliency detection using phase spectrum of quaternion fourier transform. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8, June 2008.

[45] L. Cormack H. R. Sheikh, Z. Wang and A. C. Bovik. Live image quality assessment database release 2. *Available: http://live.ece.utexas.edu/research/quality*, 2010.

[46] Patrick Haffner, Paul G Howard, Patrice Simard, Yoshua Bengio, Yann Lecun, et al. High quality document image compression with ąğdjvuąĺ. *Journal of Electronic Imaging*, 7(3):410–425, 1998.

[47] J Heemskerk, S DiNardo, and R Kostriken. Attentional resolution and the locus of visual awareness. *Nature*, 383:26, 1996.

[48] Sudeng Hu, Hanli Wang, Sam Kwong, and C-C Jay Kuo. Novel rate-quantization model-based rate control with adaptive initialization for spatial scalable video coding. *Industrial Electronics, IEEE Transactions on*, 59(3):1673–1684, 2012.

[49] Sudeng Hu, Hanli Wang, Sam Kwong, Tiesong Zhao, and C-C Jay Kuo. Rate control optimization for temporal-layer scalable video coding. *Circuits and Systems for Video Technology, IEEE Transactions on*, 21(8):1152–1162, 2011.

[50] Yi-Hsin Huang, Tao-Sheng Ou, Po-Yen Su, and Homer H Chen. Perceptual rate-distortion optimization using structural similarity index as quality metric. 20(11):1614–1624, 2010.

[51] Haruo Isono, Minoru Yasuda, and Hideaki Sasazawa. Autostereoscopic 3-d display using lcd-generated parallax barrier. *Electronics and Communications in Japan (Part II: Electronics)*, 76(7):77–84, 1993.

[52] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(11):1254–1259, Nov 1998.

[53] Information technology ITU. Digital compression and coding of continuous tone still images requirements and guidelines. *ech. Rep. T.81, ITU*, 1993.

[54] P ITU-T RECOMMENDATION. Subjective video quality assessment methods for multimedia applications. 1999.

[55] Yuting Jia, Weisi Lin, and A.A. Kassim. Estimating just-noticeable distortion for video. 16(7):820–829, July 2006.

[56] Lina Jin, Karen Egiazarian, and C-CJ Kuo. Jpeg-based perceptual image coding with block-based image quality metric. In *2012 19th IEEE International Conference on Image Processing (ICIP)*, pages 1053–1056. IEEE, 2012.

[57] T. Judd, K. Ehinger, F. Durand, and A. Torralba. Learning to predict where humans look. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 2106–2113, Sept 2009.

[58] Y. Takishima K. Takagi and A study on rate distortion optimization scheme for JVT coder Y. Nakajima. A study on rate distortion optimization scheme for jvt coder. In *Proc. SPIE*, volume 5150, Jul. 2003.

[59] Nejat Kamaci, Yucel Altunbasak, and Russell M Mersereau. Frame bit allocation for the h. 264/avc video coder via cauchy-density-based rate and distortion models. *Circuits and Systems for Video Technology, IEEE Transactions on*, 15(8):994–1006, 2005.

[60] Peter Kauff, Nicole Atzpadin, Christoph Fehn, Marcus Müller, Oliver Schreer, Aljoscha Smolic, and Ralf Tanger. Depth map creation and image-based rendering for advanced 3dtv services providing interoperability and scalability. *Signal Processing: Image Communication*, 22(2):217–234, 2007.

[61] Donald H Kelly. Motion and vision. ii. stabilized spatio-temporal threshold surface. *JOSA*, 69(10):1340–1349, 1979.

[62] Janusz Konrad and Michael Halle. 3-d displays and signal processing. *IEEE Signal Processing Magazine*, 6(24):97–111, 2007.

[63] Yung-Kai Lai and C-C Jay Kuo. A haar wavelet approach to compressed image quality measurement. *Journal of Visual Communication and Image Representation*, 11(1):17–40, 2000.

[64] Christian J Lambrecht and Murat Kunt. Characterization of human visual sensitivity for video imaging applications. *Signal Processing*, 67(3):255–269, 1998.

[65] Cuiling Lan, Guangming Shi, and Feng Wu. Compress compound images in h. 264/mpge-4 avc by exploiting spatial correlation. *Image Processing, IEEE Transactions on*, 19(4):946–957, 2010.

[66] Valero Laparra, Jordi Muñoz-Marí, and Jesús Malo. Divisive normalization image quality metric revisited. *JOSA A*, 27(4):852–864, 2010.

[67] Eric C Larson and Damon M Chandler. Most apparent distortion: full-reference image quality assessment and the role of strategy. *Journal of Electronic Imaging*, 19(1):011006–011006, 2010.

[68] Sanghoon Lee, M.S. Pattichis, and A.C. Bovik. Foveated video quality assessment. 4(1):129–132, Mar 2002.

[69] A. Leontaris and A. M. Tourapis. Rate control for the joint scalable 638 video model (jsvm). In *document JVT-W043*, 2007.

[70] Songnan Li, Lin Ma, and King Ngi Ngan. Full-reference video quality assessment by decoupling detail losses and additive impairments. 22(7):1100–1112, July 2012.

[71] ZG Li, Wen Gao, Feng Pan, SW Ma, Keng Pang Lim, GN Feng, Xiao Lin, Susanto Rahardja, HQ Lu, and Yan Lu. Adaptive rate control for h. 264. *Journal of Visual Communication and Image Representation*, 17(2):376–406, 2006.

[72] Joe Yuchieh Lin, Rui Song, Chi-Hao Wu, TsungJung Liu, Haiqiang Wang, and C-C Jay Kuo. Mcl-v: A streaming video quality assessment database. *Journal of Visual Communication and Image Representation*, 30:1–9, 2015.

[73] Anmin Liu, Weisi Lin, and Manish Narwaria. Image quality assessment based on gradient similarity. 21(4):1500–1512, 2012.

[74] H. Liu and I. Heynderickx. Visual attention in objective image quality assessment: Based on eye-tracking data. *Circuits and Systems for Video Technology, IEEE Transactions on*, 21(7):971–982, July 2011.

[75] Hantao Liu and Ingrid Heynderickx. Visual attention in objective image quality assessment: based on eye-tracking data. 21(7):971–982, 2011.

[76] Hantao Liu, Nick Klomp, and Ingrid Heynderickx. A no-reference metric for perceived ringing artifacts in images. *Circuits and Systems for Video Technology, IEEE Transactions on*, 20(4):529–539, 2010.

[77] Tsung-Jung Liu, Weisi Lin, and C-C Jay Kuo. Image quality assessment using multi-method fusion. *Image Processing, IEEE Transactions on*, 22(5):1793–1807, 2013.

[78] Tsung-Jung Liu, Weisi Lin, and CC Jay Kuo. A multi-metric fusion approach to visual quality assessment. In *Quality of Multimedia Experience (QoMEX), 2011 Third International Workshop on*, pages 72–77. IEEE, 2011.

[79] Yanwei Liu, Qingming Huang, Siwei Ma, Debin Zhao, and Wen Gao. Joint video/depth rate allocation for 3d video coding based on view synthesis distortion model. *Signal Processing: Image Communication*, 24(8):666–681, 2009.

[80] Yanwei Liu, Qingming Huang, Siwei Ma, Debin Zhao, Wen Gao, Song Ci, and Hui Tang. A novel rate control technique for multiview video plus depth based 3d video coding. *Broadcasting, IEEE Transactions on*, 57(2):562–571, 2011.

[81] Yucheng Liu and Jan P Allebach. A computational texture masking model for natural images based on adjacent visual channel inhibition. In *IS&T/SPIE Electronic Imaging*, pages 90160D–90160D. International Society for Optics and Photonics, 2014.

[82] Jeffrey Lubin. A human vision system model for objective picture quality measurements. In *Broadcasting Convention, 1997. International*, pages 498–503. IET, 1997.

[83] Siwei Ma, Wen Gao, and Yan Lu. Rate-distortion analysis for h. 264/avc video coding and its application to rate control. *Circuits and Systems for Video Technology, IEEE Transactions on*, 15(12):1533–1544, 2005.

[84] Jesús Malo, Irene Epifanio, Rafael Navarro, and Eero P Simoncelli. Nonlinear image representation for efficient perceptual coding. *Image Processing, IEEE Transactions on*, 15(1):68–80, 2006.

[85] J. Mannos and D.J. Sakrison. The effects of a visual fidelity criterion of the encoding of images. *Information Theory, IEEE Transactions on*, 20(4):525–536, Jul 1974.

[86] Pina Marziliano, Frederic Dufaux, Stefan Winkler, and Touradj Ebrahimi. Perceptual blur and ringing metrics: application to jpeg2000. *Signal processing: Image communication*, 19(2):163–172, 2004.

[87] M. Masry, S.S. Hemami, and Y. Sermadevi. A scalable wavelet-based video distortion metric and applications. 16(2):260–273, Feb 2006.

[88] Philipp Merkle, Yannick Morvan, Aljoscha Smolic, Dirk Farin, Karsten Mueller, PHN de With, and Thomas Wiegand. The effects of multiview depth video compression on multiview rendering. *Signal Processing: Image Communication*, 24(1):73–88, 2009.

[89] Philipp Merkle, Aljoscha Smolić, Karsten Müller, and Thomas Wiegand. Efficient prediction structures for multiview video coding. *Circuits and Systems for Video Technology, IEEE Transactions on*, 17(11):1461–1473, 2007.

[90] Philipp Merkle, Aljoscha Smolic, Karsten Müller, and Thomas Wiegand. Multi-view video plus depth representation and coding. In *Image Processing, 2007. ICIP 2007. IEEE International Conference on*, volume 1, pages I–201. IEEE, 2007.

[91] Marta Mrak and Ji-Zheng Xu. Improving screen content coding in hevc by transform skipping. In *Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European*, pages 1209–1213. IEEE, 2012.

[92] Karsten Mueller, Aljoscha Smolic, Kristina Dix, Philipp Merkle, Peter Kauff, and Thomas Wiegand. View synthesis for advanced 3d video systems. *EURASIP Journal on Image and Video Processing*, 2008(1):1–11, 2008.

[93] Karsten Müller, Philipp Merkle, and Thomas Wiegand. 3-d video representation using depth maps. *Proceedings of the IEEE*, 99(4):643–656, 2011.

[94] Junghak Nam, Donggyu Sim, and Ivan V Bajić. Hevc-based adaptive quantization for screen content videos. In *Broadband Multimedia Systems and Broadcasting (BMSB), 2012 IEEE International Symposium on*, pages 1–4. IEEE, 2012.

[95] King N Ngan, Kin S Leong, and H Singh. Adaptive cosine transform coding of images in perceptual domain. 37(11):1743–1750, 1989.

[96] Ha Thai Nguyen and Minh N Do. Error analysis for image-based rendering with depth information. *Image Processing, IEEE Transactions on*, 18(4):703–716, 2009.

[97] N Nill. A visual model weighted cosine transform for image compression and quality assessment. 33(6):551–557, 1985.

[98] King N.Ngan, K.S. Leong, and H. Singh. Adaptive cosine transform coding of images in perceptual domain. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 37(11):1743–1750, Nov 1989.

[99] Feng Pan, Xiao Lin, Susanto Rahardja, Weisi Lin, E Ong, Susu Yao, Zhongkang Lu, and Xiaokang Yang. A locally adaptive algorithm for measuring blocking artifacts in images and videos. *Signal Processing: Image Communication*, 19(6):499–506, 2004.

[100] Subok Park, Eric Clarkson, Matthew A Kupinski, and Harrison H Barrett. Efficiency of the human observer detecting random signals in random backgrounds. *JOSA A*, 22(1):3–16, 2005.

[101] M. Pinson and S. Wolf. Comparing subjective video quality testing methodologies. *SPIE Video Communications and Image Processing Conference*, pages 8–11, 2003.

[102] M.H. Pinson and S. Wolf. A new standardized method for objectively measuring video quality. *Broadcasting, IEEE Transactions on*, 50(3):312–322, Sept 2004.

[103] Nikolay Ponomarenko, Oleg Ieremeiev, Vladimir Lukin, Karen Egiazarian, Lukui Jin, Jaakko Astola, Benoit Vozel, Kacem Chehdi, Marco Carli, Federica Battisti, et al. Color image database tid2013: Peculiarities and preliminary results. In *Visual Information Processing (EUVIP), 2013 4th European Workshop on*, pages 106–111. IEEE, 2013.

[104] Nikolay Ponomarenko, Vladimir Lukin, Alexander Zelensky, Karen Egiazarian, M Carli, and F Battisti. Tid2008-a database for evaluation of full-reference visual quality assessment metrics. *Advances of Modern Radioelectronics*, 10(4):30–45, 2009.

[105] Nikolay Ponomarenko, Flavia Silvestri, Karen Egiazarian, Marco Carli, Jaakko Astola, and Vladimir Lukin. On between-coefficient contrast masking of dct basis functions. In *Proceedings of the Third International Workshop on Video Processing and Quality Metrics*, volume 4, 2007.

[106] Mahesh Ramasubramanian, Sumanta N Pattanaik, and Donald P Greenberg. A perceptually based physical error metric for realistic image synthesis. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 73–82. ACM Press/Addison-Wesley Publishing Co., 1999.

[107] Cesar Ramirez and Andrew Watson. A standard observer for spatial vision. *Threshold*, 10(20):30.

[108] Iain E Richardson. *H. 264 and MPEG-4 video compression: video coding for next-generation multimedia*. John Wiley & Sons, 2004.

[109] K. Seshadrinathan and A.C. Bovik. Motion tuned spatio-temporal quality assessment of natural videos. *Image Processing, IEEE Transactions on*, 19(2):335–350, Feb 2010.

[110] Feng Shao, Gangyi Jiang, Weisi Lin, Mei Yu, and Qionghai Dai. Joint bit allocation and rate control for coding multi-view video plus depth based 3d video. *Multimedia, IEEE Transactions on*, 15(8):1843–1854, 2013.

[111] Hamid R Sheikh, Alan C Bovik, and Gustavo De Veciana. An information fidelity criterion for image quality assessment using natural scene statistics. 14(12):2117–2128, 2005.

[112] Hamid Rahim Sheikh and Alan C Bovik. Image information and visual quality. *Image Processing, IEEE Transactions on*, 15(2):430–444, 2006.

[113] Hamid Rahim Sheikh, Muhammad Farooq Sabir, and Alan Conrad Bovik. A statistical evaluation of recent full reference image quality assessment algorithms. *Image Processing, IEEE Transactions on*, 15(11):3440–3451, 2006.

[114] Thomas Sikora and Bela Makai. Shape-adaptive dct for generic coding of video. *Circuits and Systems for Video Technology, IEEE Transactions on*, 5(1):59–62, 1995.

[115] Athanassios Skodras, Charilaos Christopoulos, and Touradj Ebrahimi. The jpeg 2000 still image compression standard. 18(5):36–58, 2001.

[116] Aljoscha Smolic. 3d video and free viewpoint videoąxfrom capture to display. *Pattern recognition*, 44(9):1958–1968, 2011.

[117] Jean-Luc Starck, Emmanuel J Candès, and David L Donoho. The curvelet transform for image denoising. *Image Processing, IEEE Transactions on*, 11(6):670–684, 2002.

[118] G.J. Sullivan, J. Ohm, Woo-Jin Han, and T. Wiegand. Overview of the high efficiency video coding (hevc) standard. *Circuits and Systems for Video Technology, IEEE Transactions on*, 22(12):1649–1668, Dec 2012.

[119] Patrick C Teo and David J Heeger. Perceptual image distortion. In *IS&T/SPIE 1994 International Symposium on Electronic Imaging: Science and Technology*, pages 127–141. International Society for Optics and Photonics, 1994.

[120] Phong V Vu, Cuong T Vu, and Damon M Chandler. A spatiotemporal most-apparent-distortion model for video quality assessment. In *Image Processing (ICIP), 2011 18th IEEE International Conference on*, pages 2505–2508. IEEE, 2011.

[121] Shiqi Wang, Abdul Rehman, Zhou Wang, Siwei Ma, and Wen Gao. Ssim-motivated rate-distortion optimization for video coding. 22(4):516–529, 2012.

[122] Zhou Wang, Alan C Bovik, and Ligang Lu. Wavelet-based foveated image quality measurement for region of interest image coding. In *Image Processing, 2001. Proceedings. 2001 International Conference on*, volume 2, pages 89–92. IEEE, 2001.

[123] Zhou Wang, Alan C. Bovik, Ligang Lu, and Jack L. Kouloheris. Foveated wavelet image quality index. *Proc. SPIE*, 4472:42–52, 2001.

[124] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. 13(4):600–612, 2004.

[125] Zhou Wang and Qiang Li. Information content weighting for perceptual image quality assessment. 20(5):1185–1198, 2011.

[126] Zhou Wang, Eero P Simoncelli, and Alan C Bovik. Multiscale structural similarity for image quality assessment. In *Signals, Systems and Computers, 2004. Conference Record of the Thirty-Seventh Asilomar Conference on*, volume 2, pages 1398–1402. Ieee, 2003.

[127] Andrew B Watson and Albert J Ahumada. A standard model for foveal detection of spatial contrast. *J. Vis.*, 5(9):6, 2005.

[128] Andrew B Watson, Robert Borthwick, and Mathias Taylor. Image quality and entropy masking. In *Electronic Imaging'97*, pages 2–12. International Society for Optics and Photonics, 1997.

[129] Andrew B Watson and Joshua A Solomon. Model of visual contrast gain control and pattern masking. *JOSA A*, 14(9):2379–2391, 1997.

[130] Andrew B Watson, Gloria Y Yang, Joshua A Solomon, and John Villasenor. Visibility of wavelet quantization noise. 6(8):1164–1175, 1997.

[131] Zhenyu Wei and K.N. Ngan. Spatio-temporal just noticeable distortion profile for grey scale image/video in dct domain. 19(3):337–346, March 2009.

[132] Zhenyu Wei and K.N. Ngan. Spatio-temporal just noticeable distortion profile for grey scale image/video in dct domain. *Circuits and Systems for Video Technology, IEEE Transactions on*, 19(3):337–346, March 2009.

[133] T. Wiegand, G.J. Sullivan, G. Bjontegaard, and A. Luthra. Overview of the h.264/avc video coding standard. *Circuits and Systems for Video Technology, IEEE Transactions on*, 13(7):560–576, July 2003.

[134] S. Wolf and M. Pinson. Video quality measurement techniques. *Inst. Telecommun. Sci., Boulder, CO, NTIA*, 1(21):12–32, Jun 2002.

[135] Long Xu, Songnan Li, King Ngi Ngan, and Lin Ma. Consistent visual quality control in video coding. 23(6):975–989, June 2013.

[136] Wufeng Xue, Lei Zhang, Xuanqin Mou, and Alan C Bovik. Gradient magnitude similarity deviation: a highly efficient perceptual image quality index. *Image Processing, IEEE Transactions on*, 23(2):684–695, 2014.

[137] J You, T Ebrahimi, and A Perkis. Attention driven foveated video quality assessment. 23(1):200, Dec. 2014.

[138] Hui Yuan, Yilin Chang, Junyan Huo, Fuzheng Yang, and Zhaoyang Lu. Model-based joint bit allocation between texture videos and depth maps for 3-d video coding. *Circuits and Systems for Video Technology, IEEE Transactions on*, 21(4):485–497, 2011.

[139] Lin Zhang, Ying Shen, and Hongyu Li. Vsi: a visual saliency-induced index for perceptual image quality assessment. *Image Processing, IEEE Transactions on*, 23(10):4270–4281, 2014.

[140] Lin Zhang, D Zhang, and Xuanqin Mou. Fsim: a feature similarity index for image quality assessment. 20(8):2378–2386, 2011.

[141] Jieying Zhu and Nengchao Wang. Image quality assessment by visual gradient similarity. 21(3):919–933, 2012.

[142] C Lawrence Zitnick, Sing Bing Kang, Matthew Uyttendaele, Simon Winder, and Richard Szeliski. High-quality video view interpolation using a layered representation. In *ACM Transactions on Graphics (TOG)*, volume 23, pages 600–608. ACM, 2004.