Perceptual Video Coding using Steerable Pyramids

Von der Fakultät für Elektrotechnik und Informationstechnik
der Rheinisch-Westfälischen Technischen Hochschule Aachen
zur Erlangung des akademischen Grades eines Doktors der
Ingenieurwissenschaften genehmigte Dissertation

vorgelegt von
M.Sc
Uday Singh Thakur
aus Kullu

Berichter:
Univ.-Prof. Dr.-Ing. Jens-Rainer Ohm
Univ.-Prof. Dr. David Bull

Tag der mündlichen Prüfung: 04.09.2018

Diese Dissertation ist auf den Internetseiten
der Universitätsbibliothek online verfügbar.
Don't be dead serious about your life – it's just a play.

-Sadhguru
Vorwort

Firstly, I would like express my sincere gratitude to my advisor Prof. Jens-Rainer Ohm for providing his continuous support, encouragement and expertise on my PhD study and related research. His guidance helped me in all the time of research and writing of this thesis. Besides my advisor, I am also grateful to Prof. David Bull for his insightful comments on the thesis and for providing me an opportunity to join his team for my secondment at the University of Bristol. My sincere thanks also goes to Dr.-Ing Mathias Wien for providing continuous support and expertise with my research and also helping me with my everyday challenges at the IENT. Besides my advisor and Prof. David Bull, I would also like to thank the rest of my thesis examination committee: Prof. Anke Schmeink and Prof. Rainer Waser.

I also thank all my fellow labmates for their support and help during my stay at the IENT and for all the fun we had in the last 4 years. Special thanks to Bappadiya Ray and Madhukar Bhat for their support and help.

Last but not the least, I would like to thank my family: my parents and my sisters (Navdeep, Sara and Isha) and my friend: Shiva for supporting me throughout writing this thesis and my life in general.

Kullu, July 2019
## Contents

1 Introduction .................................................. 1
   1.1 Main Contributions of this Thesis ..................... 2
   1.2 Outline .................................................. 2

2 Fundamentals ................................................. 5
   2.1 Role of Phase and Human Vision ....................... 5
   2.2 Steerable Filters ........................................ 6
      2.2.1 Filter Constraints .................................. 6
      2.2.2 Angular Decomposition ............................... 7
      2.2.3 Radial Decomposition ................................. 7
      2.2.4 Implementation ...................................... 8
   2.3 Video Coding Challenges and Proposed Solution ....... 11

3 Static Textures: Perceptual Coding using Texture Synthesis 15
   3.1 Static Textures ......................................... 15
   3.2 Parameter Coding for Texture Synthesis ................ 16
   3.3 Texture Analysis and Parameter Types .................. 16
      3.3.1 Marginal Statistics ................................ 16
      3.3.2 Coefficient Correlation ............................. 17
      3.3.3 Magnitude Correlation ............................... 17
      3.3.4 Cross-Scale Phase Statistics ......................... 17
   3.4 Compression Scheme ...................................... 18
      3.4.1 Compression of Auto-correlation Function (ACF) .... 18
      3.4.2 Candidates for Auto-correlation Function (ACF) Prediction 19
      3.4.3 DCT property for Auto-correlation Function (ACF) .... 23
      3.4.4 Significant Coefficient Scan .......................... 25
      3.4.5 Compression for Intra-Scale Cross-correlation ....... 27
      3.4.6 Compression for Inter-scale Cross-correlation ........ 30
      3.4.7 Compression of Marginal Statistics .................. 30
   3.5 Texture Synthesis ........................................ 30
   3.6 Experimental Results .................................... 31
      3.6.1 Subjective Evaluations ............................... 31
      3.6.2 Effects of Quantizing Parameters on Texture Synthesis 32
      3.6.3 Limitations of Texture Synthesis .................... 32
   3.7 Conclusion ............................................... 33

4 Reduced Resolution Coding with Detail Reconstruction in B-Pictures 37
   4.1 Introduction ............................................. 37
   4.2 Dynamic Texture and Coding Challenges ................ 37
      4.2.1 Reduced Resolution Update ........................... 39
## Contents

4.3 Reduced Resolution Coding for B-Pictures ................................................. 40
   4.3.1 Coding of Low-pass Signal ................................................................. 40
   4.3.2 Reconstruction of the RR picture at the Decoder ................................. 41

4.4 Post-processing for the Reduced Resolution Picture .................................. 43
   4.4.1 Motion Estimation using Low-pass Picture ........................................... 43
   4.4.2 Motion Estimation using magnitude of the orientated BP ..................... 44
   4.4.3 Motion Estimation using Phase of the orientated BP ............................ 46
   4.4.4 Spatial Domain Upsampling of RR Coded Pictures ............................... 47
   4.4.5 Error Correction and Hole-filling ....................................................... 47

4.5 Grand Challenge Model ................................................................. 49
   4.5.1 Motion Estimation using Band-pass Coefficients ................................ 50
   4.5.2 Error Correction ................................................................. 50

5 Conclusion ........................................................................ 53

A Annex: Additional Results on Static Texture Synthesis ............................ 55

B Annex: Results of the RR coding ............................................................. 71
   B.1 Experimental Set-up and Results ........................................................ 71
   B.2 Experimental Set-up ........................................................................ 71
      B.2.1 Encoder configuration ................................................................. 71
      B.2.2 Content Specifications and Rate Points ....................................... 72
      B.2.3 Subjective Test Conditions .......................................................... 73
      B.2.4 Subjective Test Methodology ....................................................... 73
      B.2.5 Subjective Results .................................................................. 75
      B.2.6 Objective Results ................................................................. 76

Bibliography ........................................................................ 91
Notation

$B_k(\omega)$  band-pass oriented filter
$H_0(\omega)$  high-pass filter
$L_0(\omega)$  low-pass filter
$X(f)$  Continuous Time Fourier Transform of $x(t)$
$|X(f)|$  magnitude spectrum of $X(f)$
$Z(f)$  analytic spectrum of signal $z(t)$
$z_r(t)$  real part of the analytic signal
$z_i(t)$  imaginary part of the analytic signal
$HT\{\cdot\}$  Hilbert transform
$arg(\cdot)$  argument of a function
$B_{s,k}(r, \theta)$  oriented filter in Fourier domain
$A_s(r)$  radial component of oriented filter
$G_k(\theta)$  angular component of oriented filter
$P(\cdot)$  Probability
$f(m,n)$  auto-correlation matrix of size $M \times N$
$F(p,q)$  DCT of the auto-correlation matrix $f(m,n)$
$CC(i,j)$  cross-correlation between sub-bands $i$ and $j$
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACF</td>
<td>Auto-correlation function</td>
</tr>
<tr>
<td>BP</td>
<td>Band-pass</td>
</tr>
<tr>
<td>CABAC</td>
<td>Context-based adaptive binary arithmetic coding</td>
</tr>
<tr>
<td>CTFT</td>
<td>Continuous Time Fourier Transform</td>
</tr>
<tr>
<td>CVS</td>
<td>Coded video sequence</td>
</tr>
<tr>
<td>DCT</td>
<td>Discrete Cosine Transform</td>
</tr>
<tr>
<td>EG</td>
<td>Exp-Golomb code</td>
</tr>
<tr>
<td>FFT</td>
<td>Fast Fourier Transform</td>
</tr>
<tr>
<td>GOP</td>
<td>Group of pictures</td>
</tr>
<tr>
<td>HEVC</td>
<td>High efficiency video coding</td>
</tr>
<tr>
<td>HP</td>
<td>High-pass</td>
</tr>
<tr>
<td>HVS</td>
<td>Human visual system</td>
</tr>
<tr>
<td>ICIP</td>
<td>International conference on Image Processing</td>
</tr>
<tr>
<td>IFFT</td>
<td>Inverse Fast Fourier Transform</td>
</tr>
<tr>
<td>JEM</td>
<td>Joint exploration model</td>
</tr>
<tr>
<td>JVET</td>
<td>Joint video exploration team</td>
</tr>
<tr>
<td>LP</td>
<td>Low-pass</td>
</tr>
<tr>
<td>MC</td>
<td>Motion-compensation</td>
</tr>
<tr>
<td>ME</td>
<td>Motion-estimation</td>
</tr>
<tr>
<td>MV</td>
<td>Motion-vector</td>
</tr>
<tr>
<td>NN</td>
<td>Nearest neighbour</td>
</tr>
<tr>
<td>PCA</td>
<td>Principal component analysis</td>
</tr>
<tr>
<td>POC</td>
<td>Picture order count</td>
</tr>
<tr>
<td>PSNR</td>
<td>Peak signal to noise ratio</td>
</tr>
<tr>
<td>RA</td>
<td>Random access</td>
</tr>
<tr>
<td>RR</td>
<td>Reduced resolution</td>
</tr>
<tr>
<td>SAD</td>
<td>Sum of absolute difference</td>
</tr>
<tr>
<td>SHVC</td>
<td>Scalable high efficiency video coding</td>
</tr>
<tr>
<td>SLP</td>
<td>Shift corrected low-pass</td>
</tr>
<tr>
<td>SP</td>
<td>Steerable pyramid</td>
</tr>
<tr>
<td>SRC</td>
<td>Source video sequence</td>
</tr>
<tr>
<td>SSIM</td>
<td>Structural similarity index</td>
</tr>
<tr>
<td>TSB</td>
<td>Transform sub-block</td>
</tr>
<tr>
<td>TU</td>
<td>Truncated Unary binarization</td>
</tr>
<tr>
<td>UHD</td>
<td>Ultra high definition</td>
</tr>
<tr>
<td>ULP</td>
<td>Upsampled low-pass</td>
</tr>
</tbody>
</table>
1 Introduction

The biological vision has many interesting properties that can be exploited for improving the efficiency of today’s video compression. For example, human eye is relatively more sensitive to certain spatial and temporal frequencies which can be exploited for content adaptive coding. However, a few perceptual optimizations have already been very well exploited in today’s compression world such as, discarding high-frequencies which correspond to the fine details and are perceptually of little importance. This is primarily achieved by adaptively quantizing transform coefficients corresponding to the high-frequencies \cite{Wie14}. Such a compression scheme holds well for data with low spatial variance due to decent energy compaction achieved in the transform domain. In the case of content with high local variance the energy compaction is limited and therefore, discarding any high-frequency coefficients introduces higher perceptual distortions. In addition to this, if the motion is characterized with rapid and non-linear change over time (temporal high-frequencies) then it may limit the performance of motion compensation. In such a case the residual will have a high energy and leading to a peak in the bit-rate requirements.

In the context of this thesis we are mainly focused on the content that remains challenging to code for current generation video codecs \cite{Afo+16} and in a broad sense characterized with either only high spatial frequencies or both high spatial and temporal frequencies. Former refers to static textures which is purely spatial i.e. corresponds to an object in a scene and exhibits spatial homogeneity such as fabric, surface of stone, walls, etc. \cite{BSO11}. The latter refers to dynamic textures that are video sequences with changing texture showing a certain degree of temporal stationarity. Based on the type of motion, dynamic textures may exhibit continuous appearance (e.g., water surfaces, flames from a burning candle, etc.) or discernible appearance (e.g., leaves fluttering in the wind) \cite{Dor+03}. For static textures that specifically exhibit near stochastic appearance, it is hard for the human eye to perceive the fine details inside such a content as it is characterized with near random phase. Distortion measures like PSNR and SSIM \cite{Wan+04} are unreliable when evaluating the visual quality of such content. Texture synthesis \cite{PS00a} is a technology that synthesizes artificial textures using modeled parameters from the amplitude and phase separately of a multiscale and multi-orientation sub-band decomposition and, reconstructs decent quality perceptual results at much lower PSNR. Chapter 3 of the thesis explores the potential of texture synthesis as a perceptual coding tool from the perspective of efficiently coding static textures.

For dynamic textures on the other hand, often characterized with complex motion patterns and as a consequence, a relatively high percentage of the bit-rate is spent in coding the B-pictures corresponding to the two highest temporal $id$s. At lower rates, such pictures are highly prone to the blurring and blocking artefacts and also sometime introduces motion jerkiness in the video thus, leading to an overall decrease in the perceptual quality. Chapter 4 of the thesis introduces an adaptive spatial resolution scheme designed for B-pictures($temporal_{id} > 2$) with post-processing (detail reconstruction) at the decoder side, using motion compensated band-pass signal from the nearest full resolution pictures. The
scheme overall tends to minimize the artefacts originating from these pictures, thus leading to a significant improvement in the overall visual quality for certain class of dynamic textures. Further, the performance of this scheme is also evaluated for other content beyond dynamic textures.

1.1 Main Contributions of this Thesis

As mentioned above, the thesis deals with static and dynamic texture coding. In the context of former, texture synthesis based on steerable pyramid band-pass statistics is used as an alternative tool for coding. The main contribution in this aspect lies in the design of a novel compression scheme for coding of the synthesis parameters. This involves a dedicated compression scheme for each parameter type which includes steps like transform, quantization, prediction and adaptive binarization. The thesis also includes investigation on how parameter quantization affects the visual quality of synthesized texture. Further, pairwise subjective comparisons are provided to judge the performance of the proposed method, showing clear benefits against the HEVC [Sul+12] coded texture at similar rate, provided that the area of homogeneous texture is sufficiently large.

For dynamic textures, the main contribution is made by introducing a post-processing step for spatio-temporal synthesis of the high-frequency component in B-pictures of Joint Exploration Model (JEM) [JEM16]. This is based on the observation that otherwise, in a typical random access scenario under limited bit-rate conditions, such pictures tend to be most susceptible to be carrying visual artefacts in the case of dynamic textures. An adaptive spatial resolution scheme is introduced, where at the encoder side these B-pictures are coded in a down-sampled resolution. At the decoder, the down-sampled pictures are decoded which is followed by their up-sampling using steerable pyramids (as a post-processing step). The up-sampling method reconstructs high frequencies using motion compensated band-pass signal from the nearest full resolution key-picture. Visual artefacts originating from possibly incorrect reconstructed high frequencies are separately handled and suppressed. The performance of the proposed method is evaluated subjectively as well as objectively against the state-of-the-art JEM reference codec, and it is shown to provide quality improvements at same rate for some sequences.

1.2 Outline

The thesis deals with coding of two different classes of textures i.e., static and dynamic. In Chapter 2 the necessary theoretic foundation related to steerable pyramids and concepts of video coding which are relevant to this thesis are introduced. A description of synthesis parameters and a detailed compression scheme for each parameter type is described in Chapter 3 which also includes the results of the proposed method, discussion on the effects of parameter quantization and limitations of the synthesis coding model. In Chapter 4 we introduce the coding challenges with dynamic textures and describe our proposed reduced resolution B-picture coding model with decoder side post-processing. An evaluation of the model discussed in Chapter 4 is provided in annex B. Final conclusions for the both the developed models are drawn in Chapter 5.
Additional results for the Chapter 3 are also provided in the annex A, which is followed by references.
2 Fundamentals

This thesis deals with the perceptually optimized video coding for static and dynamic texture content. To understand the methods which are developed in this thesis, some prior understanding of basic concepts and methods are necessary. These fundamentals are provided in this chapter.

2.1 Role of Phase and Human Vision

The importance of Fourier phase and amplitude for retrieval of structural image features has been debated in several previous works [OL81, LMS97, NH07]. The basic notion is that phase bears topological information about position of structures in an image, whereas amplitude encodes image intensity. The experimental evidence in [GE15] with phase only reconstruction of image clearly indicates that Fourier phase is more significant for retrieval of cognitive image features (such as edges) when compared to the amplitude only reconstruction. Any distortion of phase affects topology of edge structures and may makes the image unrecognizable. Pioneering works of [HW62; CR68; BC69; TBK69] tell us that different groups of neurons in the visual cortex act as spatially organized filters that extract particular image features (i.e., spatial frequency and orientation) within a certain range (bandwidth) of their sensitivity. Further studies were carried on modeling of visual receptive fields and analysis of their amplitude- and phase-transfer functions [ODF90; AB85; Hee92; Fle94]. Physiological evidence in [MB88] indicates that the human visual system responds strongly to points in an image where the phase information is highly ordered also termed as phase congruency [Kov00]. It is an important characteristic of phase and used for detecting corners and edges [Kov03].

However, when it comes to textures perception theory suggests that some of the main texture characteristics are contained in the Fourier magnitude [Jul62; WK07]. Under such assumption, its perception should not vary when the texture phase is randomized. This concept holds well for micro textures (stochastic textures) which are generally characterized with high degree of randomness in their phase [GGM11]. The texture synthesis model described in [GGM11] works very well for such random phase textures. Another synthesis model for micro-textures based on sub-band statistics was experimented for coding in [BSO11] and provided better visual quality compared to the reference codec. This shows the potential of such models when exploited for specifically coding micro-textures. The synthesis model proposed in [PS00a] uses statistics of both the Fourier magnitude and phase and synthesizes relatively wider range of textures with decent perceptual quality thus showing the relevance of phase for modeling textures. Therefore, in our work we investigate the texture synthesis model given in [PS00a] for coding static textures. It is evident from the results in this thesis that the model has a strong potential to provide a compact representation for coding a decent range of textures with better visual quality when compared to HEVC’s intra prediction [Sul+12].
2.2 Steerable Filters

Steerable filter is a class of filters in which a filter of any arbitrary orientation is constructed as a linear combination of a set of basis filters \([\text{FA91}]\). Such filters are extensively used in image and video processing applications e.g. image de-noising \([\text{Por+03}]\), image extrapolation \([\text{Su+05}]\), image fusion \([\text{Liu+01}]\), texture retrieval \([\text{DV02}; \text{TBT05}]\), texture analysis and synthesis \([\text{PS00b}]\), sparsity-constrained image reconstruction \([\text{DDM}]\), image data compression \([\text{TM01}; \text{TC15}]\), motion analysis and processing \([\text{Mey+15}; \text{TC15}]\), image enhancement, etc. Steerable pyramid decomposition \([\text{Sim+92}; \text{KS96}; \text{Gre+94}; \text{FA91}; \text{SF95}]\) provides an over-complete representation of an image into sub-bands which are localized in both scale and orientation. The scale tuning of the filters is constrained by a recursive system see Figure 2.1. The orientation tuning is constrained by the property of steerability. According to \([\text{FA91}]\), "A set of filters form a steerable basis (1) if they are rotated copies of each other, and (2) a copy of the filter at any orientation may be computed as a linear combination of the basis filters." Overall this provides a method of steering the basis functions in the direction of maximal response. Further, the transform is designed to be aliasing-free for \(\beta=1\) as shown in Figure 2.6 (\(\beta\) is the roll-off for the raised-cosine filter).

\[ \hat{X}(\omega) = \left[ |H_0(\omega)|^2 + |L_0(\omega)|^2 \left( |L_1(\omega)|^2 + \sum_{k=0}^{K} |B_k(\omega)|^2 \right) \right] X(\omega) + a.t. \]  

(2.1)
2.2 Steerable Filters

Figure 2.2 Steerable pyramid representation for single stage with first derivative i.e. $K = 1$. It is to be noted that number of orientation will be $K + 1$.

where $a.t.$ is the aliasing term. $L_1$ filter has a zero response for frequencies greater than $\pi/2$ along both $\omega_x$ and $\omega_y$ which eliminates the aliasing terms. The transfer function of the system is equal to one. The constraints on the steerable filters are described in the subsequent sub-sections.

2.2.2 Angular Decomposition

The angular constraint for the band-pass filters $B_k(\vec{\omega})$ is derived from the condition of steerability [FA91] and can be expressed as:

$$B_k(\vec{\omega}) = B(\vec{\omega}) \left[ -j \cos(\theta - \theta_k) \right]^K$$  \hspace{1cm} (2.2)

where $\theta = \text{arg}(\vec{\omega})$, $\theta_k = \pi k / (K + 1)$ for $k \in \{0, 1, \ldots, K\}$, and

$$B(\vec{\omega}) = \sqrt{\sum_{k=0}^{K} |B_k(\vec{\omega})|^2}$$  \hspace{1cm} (2.3)

Equation 2.2 states that $B_k(\vec{\omega})$ is the $K$th order directional derivative, in the direction $\theta_k$, of the function $B(\vec{\omega})/|\vec{\omega}|^2$.

2.2.3 Radial Decomposition

The constraints on the filters given in the diagram Figure 2.1 are as follows:

1. Band-limiting to prevent aliasing in sub-sampling operation:

$$L_1(\vec{\omega}) = 0 \text{ for } |\vec{\omega}| > \pi/2$$  \hspace{1cm} (2.4)
2 Fundamentals

2. Flat System Response:

\[ |H_0(\vec{\omega})|^2 + |L_0(\vec{\omega})|^2 \left[ |L_1(\vec{\omega})|^2 + \sum_{k=0}^{K} |B_k(\vec{\omega})|^2 \right] = 1 \]  

(2.5)

3. Recursion:

\[ |L_1(\vec{\omega})/2|^2 = |L_1(\vec{\omega})/2|^2 \left[ |L_1(\vec{\omega})|^2 + \sum_{k=0}^{K} |B_k(\vec{\omega})|^2 \right] \]  

(2.6)

During recursion, \( L_1(\vec{\omega}/2) \) plays the role of the initialization filter \( L_0(\vec{\omega}) \). This is enforced by setting \( L_0(\vec{\omega}) = L_1(\vec{\omega}/2) \).

2.2.4 Implementation

The filters are designed using weighted least squares techniques in the Fourier domain to approximately fit the constraints detailed above. Higher derivatives filters are used to obtain high angular resolution e.g. case \( K=3 \) as shown in Figure 2.5. In this thesis, complex analytic filter type is used which is an extended version of steerable pyramid representation i.e. the real and imaginary parts correspond to a pair of even- and odd-symmetric filters (analogous to a Hilbert Transform pair in one dimension) [PS00a]. This allows us to utilize measures of local phase and energy as texture descriptors. A short description on how to create analytic signals from real signals is given next, which will be helpful in understanding the complex output of the steerable pyramids coefficients.

Analytic signals

The subsequent paragraph including Figure 2.3 is replicated (with permission of the author) from section 1.8 (p.32) of [MV17]:

"Fourier Transform of a real-valued signal is complex-symmetric. It implies that the content at negative frequencies are redundant with respect to the positive frequencies. Gabor [Gab46] aimed to create an analytic signal by removing redundant negative frequency content resulting from the Fourier transform. The analytic signal is complex-valued but its spectrum will be one-sided (only positive frequencies) that preserved the spectral content of the original real-valued signal.

Let \( x(t) \) be a real-valued non-bandlimited finite energy signal, for which we wish to construct a corresponding analytic signal \( z(t) \). The Continuous Time Fourier Transform (CTFT) of \( x(t) \) is given by

\[ X(f) = \int_{-\infty}^{+\infty} x(t)e^{-j2\pi ft} dt \]  

(2.7)

Magnitude spectrum of \( X(f) \) is shown in Figure 2.3a. We note that the signal \( x(t) \) is a real-valued and its magnitude spectrum \( |X(f)| \) is symmetric and extends infinitely in the frequency domain. In frequency domain, the spectral content \( Z(f) \) of the analytic signal
2.2 Steerable Filters

(a) CTFT of real-valued non-bandlimited finite energy signal $x(t)$. (b) One sided magnitude spectrum of the corresponding analytic signal $z(t)$.

Figure 2.3 (a) shows the spectrum of continuous signal $x(t)$ and (b) shows the spectrum of analytic signal $z(t)$.

$z(t)$ is given by

$$Z(f) = \begin{cases} 
X(0) & \text{for } f = 0 \\
2X(f) & \text{for } f > 0 \\
0 & \text{for } f < 0 
\end{cases} \quad (2.8)$$

Since the spectrum of the analytic signal is one-sided, the analytic signal will be complex valued in the time domain, hence the analytic signal can be represented in terms of real and imaginary components as $z(t) = z_r(t) + jz_i(t)$. Since the spectral content is preserved in an analytic signal, it turns out that the real part of the analytic signal in time domain is essentially the original real-valued signal itself $z_r(t) = x(t)$. Hilbert transform [Liu12] can be used to find a companion function (imaginary part in the equation above) to a real-valued signal such that the real signal can be analytically extended from the real axis to the upper half of the complex plane. Denoting Hilbert transform as $HT\{\cdot\}$, the analytic signal is given by

$$z(t) = z_r(t) + jz_i(t) = x(t) + jHT\{x(t)\} \quad (2.9)$$

From the above discussion it can be concluded that analytic signal $z(t)$ for a real-valued signal $x(t)$, can be constructed using two approaches. (1) Frequency domain approach: The one-sided spectrum of $z(t)$ is formed from the two-sided spectrum of the real-valued signal $x(t)$ by applying [Equation 2.8]. (2) Time domain approach: Using Hilbert transform approach as given in [Equation 2.9].

In the case of steerable pyramids frequency domain approach is used in the generation of complex coefficients as shown in Figure 2.5b and Figure 2.4b.

The oriented filters are polar-separable in the Fourier domain, and are written as (for details see [PS00a]):

$$B_{s,k}(r, \theta) = A_s(r) G_k(\theta), \quad s \in [0,S], \ k \in [0,K] \quad (2.10)$$

with radial and angular parts

$$A_s(r) = \begin{cases} 
\cos\left(\frac{\pi}{2} \log_2\left(\frac{r}{\pi}\right)\right), & \frac{r}{\pi} \in \left[\frac{1}{2}, 2\right] \\
0, & \text{otherwise} 
\end{cases} \quad (2.11)$$
2 Fundamentals

Figure 2.4 Steerable filter mask of a particular sub-band is shown for both complex and real output.

(a) Filter mask for real output
(b) Filter mask for complex output

Figure 2.5 Idealized illustration of the spectral decomposition performed by a steerable pyramid with 3 scales and 4 orientations. Frequency axes range from $-\pi$ to $\pi$. Shaded region indicates the spectral component of a single sub-band (vertically-oriented) at level 2 for both real and complex valued pyramid decomposition.

\[
G_k(\theta) = \begin{cases} 
\cos\left(\theta - \frac{\pi k}{K}\right)^{(K-1)}, & \left|\theta - \frac{\pi k}{K}\right| < \frac{\pi}{2} \\
0, & \text{otherwise}
\end{cases} \tag{2.12}
\]

where $r$, $\theta$ are polar frequency coordinates with $s$th pyramid level, $k$th orientation, $S$ is the pyramid levels and $K$ is derivative order. Sub-bands are sub-sampled by a factor of $2^s$ along both axes. The high- and low-pass bands are defined as:

\[
H(r) = \begin{cases} 
\cos\left(\frac{\pi}{2} \log_2\left(\frac{r}{\pi}\right)\right), & r \in \left[\frac{\pi}{2}, \pi\right] \\
0, & \text{otherwise}
\end{cases} \tag{2.13}
\]

\[
L(r) = \begin{cases} 
\cos\left(\frac{\pi}{2} \log_2\left(\frac{2^{s+1}r}{\pi}\right)\right), & r \in \left[\frac{\pi}{2^{s+1}}, \frac{\pi}{2^s}\right] \\
1, & r < \frac{\pi}{2^s+1} \\
0, & r > \frac{\pi}{2^s} \tag{2.14}
\end{cases}
\]

In the context of static textures i.e. Chapter 3 we have used fixed configuration of steerable
pyramids with 3 scales and 3 orientations and in the context of reduced resolution coding
i.e. Chapter 4 we have used fixed configuration of steerable pyramids with 1 scale and 2
orientations.

![Graph](image)

Figure 2.6 Transfer function of raised-cosine filter with various roll-off ($\beta$) factors in case of 1 and 2
dimensional signal as used in Steerable Pyramid low-pass. In this thesis the value of $\beta$ is kept 1.

### 2.3 Video Coding Challenges and Proposed Solution

For necessary fundamentals on video coding refer to [Wie14]. The fundamental concept
behind most video coding standards is the hybrid coding scheme i.e. a merger of temporal
prediction between pictures with transform coding for the residual signal. Generally, intra- or
inter-prediction is performed on the video signal that is input into the encoder and a residual
signal is generated by subtracting the prediction from the input signal. Subsequently, the
residual signal is transformed, quantized, binarized and encoded into the bitstream together
with parameters required for decoding the bitstream.

In the context of this thesis we are mainly focused on two different types of content that
are highly challenging to code for the state-of-the-art video coding technologies like HEVC
and JEM [Wie14; JEM16]. Static textures is one such category that is challenging to code
(in the context of intra coding). Such a content comprises of high amount of spatial variations
which are randomly distributed over the textured region (see hay texture Figure 2.7a).
During encoding process intra-coding tools available within HEVC are employed for the pre-
diction of samples but due to high variations between the neighboring samples of such a
content, it is likely that residual has relatively high energy as compared to the content with
less spatial variations. The residual is further transformed to remove any further correlation
if present within the residual signal and potentially represent the signal with a few trans-
form coefficients. It turns out that due to the very nature of the residual (high energy) the
number of significant coefficients are very large which eventually requires large rate for cod-
ing (ideally, there should be few significant coefficients which concentrate towards the low
frequency bases in the transform block and at higher frequency positions may only contain
zero coefficients). Often at very low bit-rates, heavy blocking and blurring artefacts tend to
occur over such regions which downgrades the perceptual quality (see Figure 2.7b).

Two interesting visual properties of static textures that motivates the work in this thesis are
(1) Human Visual System can only perceive limited amount of details inside a textured region
and (2) textures tend to occur in a continuous region and homogeneously spread over a large
2 Fundamentals

Figure 2.7 Blocking and blurring artefacts observed at lower bit-rate when using HEVC’s compression on a hay texture.

area. In Figure 2.7 the HEVC coded image is at an approximately 5 dB higher PSNR than the synthesized image but, perceptually the synthesized image is visually better. State-of-the-art video coding algorithms do not take into account these properties when coding such content. In our approach for coding static textures, a parametric texture synthesis model described in [PS00a] is employed as an alternative coding tool with HEVC. The synthesis model exploits the two previously mentioned visual properties of textures by statistically modeling the content. The model parameters are then coded into the bitstream and later reconstructed at the decoder side for synthesis. Overall from the results provided in this thesis, it can be judged subjectively that a better visual quality is achievable at a very low bit-rate when compared to HEVC at the same rate, provided the area to be synthesized is large enough (e.g. see Figure 2.7c). However due to limited performance of the objective quality measures such as PSNR, SSIM etc. it is not possible to employ such objective metrics
2.3 Video Coding Challenges and Proposed Solution

The second category of the content that we discuss in this thesis is the dynamic textures. Dynamic textures are mostly characterized with complex motion patterns and based on their appearance they may be continuous (e.g. water waves Figure 4.1a) or discrete (e.g. leaves swirling in the wind Figure 4.2a). Performance of the state-of-the-art video compression technologies like HEVC is very limited when coding dynamic texture content. During encoding, continuous dynamic textures are often characterized with intra mode selection indicating motion compensations failure to find a good predictor as shown in Figure 4.1. In case of discrete dynamic textures there is high percentage of small partitions (4×4) which is mostly due to the presence of random motions at the granular level as shown in Figure 4.2. HEVC coding analysis in \cite{Afo16} points to the fact that majority of the bits for coding dynamic textures (both continuous and discrete) are spent on the residual. Further, coding statistics given in \cite{Tha17} substantiate the fact that B-pictures relatively require much higher rate for dynamic textures than it would be needed to code content with simple linear motions. Under low bit-rate conditions, heavy blocking and blurring artefacts are often observed in B-pictures corresponding to the two highest temporal ids. Presence of such artefacts in the B-pictures tends to introduce jerkiness due to highly dynamic nature of the content, this significantly downgrades the overall perceptual quality of the scene.

Perceiving motion in case of dynamic texture is limited for Human Visual System due to high amount of random temporal variation at the very granular level. In our proposed solution for the dynamic textures (considering random access configuration) we target those pictures that are most susceptible to artefacts (i.e. two highest temporal ids) and code these pictures in a reduced resolution format with a lower QP offset. In this way a decent quality is reconstructed in the reduced resolution. At the decoder side these pictures are up-sampled together with high-frequency synthesis using information from the neighboring full resolution frames. The subjective results provided in Appendix B is a concrete evidence that our model outperforms JEM at the same rate as our overall perceptual quality is better in majority cases when viewing dynamic textures. On the downside the proposed model has a very high complexity due to motion estimation being carried out at the decoder side for high-frequency synthesis.

![Figure 2.8](image) Blocking artefacts observed in one of the pictures corresponding to the highest temporal ids at low bit-rate coding when using JEM’s compression on dynamic texture region. In the proposed coding method the relatively less blocking artefact is observed at the same rate.
2 Fundamentals

The results for standard content sequences are also provided in the Appendix B and it is evident that the performance of the proposed model is generally better when compared to the JEM at low rate points.
3 Static Textures: Perceptual Coding using Texture Synthesis

In general, the word texture refers to surface characteristics and appearance of an object given by the size, shape, density, arrangement, proportion of its elementary parts. Although textures have been studied and researched over the last few decades in image processing and computer vision, there is still no precise definition of the notion texture, mainly due to its wide range of visual patterns. In the context of this thesis, textures are divided into two main classes: static or dynamic. The former type is purely spatial and refers to an object in a scene which exhibits spatial homogeneity such as a fabric, surface of a stone etc. The latter is a time-varying visual pattern that may or may not exhibit certain temporal stationarity, e.g., water waves, flames from a burning material etc. The purpose of the investigation is to find alternative methods besides the existing ones for specifically coding textures, that also exploit human perception and altogether benefit state-of-the-art video compression technology.

3.1 Static Textures

Static texture is a very general term used to describe a wide range (regular → stochastic) of textured content with contradicting properties. Haralick considers a texture as an organised area phenomenon which can be decomposed into primitives having specific spatial distributions, also known as structural approach. For instance, Figure 3.1a is composed of particular texture elements, i.e. bricks which are organised in a particular spatial structure indicating certain underlying placement rules. On the other hand, Cross and Jain suggested texture to be a stochastic, possibly periodic, two-dimensional image field. This definition describes a texture as a Markov random field. As already discussed in Section 2.3 that textures are challenging to code even for state-of-the-art video compression technologies like HEVC and JEM. As a potential solution for coding textures we have explored texture synthesis as a coding tool. It has shown remarkable results in the past on a wide range of textured content by synthesizing textures with a high degree of perceptual
equivalence. It can be either parametric [Bal12; PS00a] or non-parametric [EF01; WL00]. In this thesis we have only used parametric texture synthesis model given in [PS00a] and compared its results with the intra coding of HEVC.

3.2 Parameter Coding for Texture Synthesis

In this section a coding scheme for texture synthesis parameters is proposed with reference to the texture synthesis method proposed in [PS00a]. Different synthesis parameter types and a compression scheme for each type are described here. Depending on the parameter type, a transform may or may not be applied which is followed by prediction from available candidates and quantization of residual. The parameters are later reconstructed at the decoder from the bitstream and used for synthesis of the analysed textured content. A very general texture analysis and synthesis framework is shown below in Figure 3.2. It is to be noted that synthesis is performed on a wide range of homogeneous texture patches mostly of size 256 × 256. Identifying textured regions in an image is beyond the scope of this thesis.

3.3 Texture Analysis and Parameter Types

The input texture image (RGB) signal is analysed in the complex wavelet domain using the steerable pyramid decomposition described in [Chapter 2]. The statistical descriptors are estimated on the pairs of wavelet coefficients from steerable pyramid decomposition (separately for each color channel), at adjacent spatial locations, orientations, and scales. In particular, the expected product of the raw coefficient pairs (i.e., correlation), and the expected product of their magnitudes is measured. The parameters also include a few marginal statistics of the input signal and low-pass coefficients at different scales. A brief explanation of statistical descriptors and the parameters associated with them are as follows.

3.3.1 Marginal Statistics

Marginal statistics include scalar values such as mean, variance, skewness, kurtosis and range of intensity i.e maximum and minimum value of each color component of the input RGB signal. Skewness, kurtosis and range of intensity of the Principal Component Analysis (PCA) computed between pixels of the same color channel. Skewness and kurtosis of the partial reconstruction of the low-pass signal at each scale. Mean of each sub-band (magnitude), variance of high-pass signal are also computed.
3.3 Texture Analysis and Parameter Types

3.3.2 Coefficient Correlation

Covariances of sub-band coefficients can arise from spectral peaks (i.e., periodicity) or ridges (i.e., globally oriented structure) in a texture \cite{FMP93}. In order to represent these spectral features, central part of the 2D auto-correlation function (ACF) of all sub-band (real part of the coefficient) are used as a texture descriptor. The local auto-correlation of the low-pass images computed at each level of the pyramid decomposition are also used as descriptors. This set of parameter provides high spectral resolution in the low frequencies and low spectral resolution in high frequencies, which is a natural solution for a scale-invariant modelling of images \cite{TPN99}. Such parameters tend to represent periodic structures and long-range correlations.

3.3.3 Magnitude Correlation

Edges or strong features give rise to large coefficients in local spatial neighborhoods, as well as at adjacent scales and orientations. Correlation of the magnitude of pairs of coefficients at adjacent positions, orientations and scales is computed as a descriptor for features see Figure 3.3.

3.3.4 Cross-Scale Phase Statistics

It is an important statistics to represent the phase of the responses to local features, such as edges and lines. Phase across scales is coherent at the edges. In order to capture the local phase behaviour, the relative phase of coefficients of sub-bands at adjacent scales is measured. In general, the local phase varies linearly with distance from features, but the rate at which it changes for fine-scale coefficients is twice that of those at the coarser scale \cite{PS00a}. To compensate for this, the complex phase of the coarse-scale coefficients is doubled to compute the cross-correlation of these modified coefficients with the fine-scale coefficients (To compute the inter-scale statistics, the coarser sub-band is upsampled to match the dimensions of the fine sub-band).

Figure 3.3 The figure shows the inter- and intra-scale cross-correlation of a single sub-band shown in darker grey shade with all other sub-bands marked in the lighter grey shade (3-scale and 3-orientation decomposition) for a single color channel image.
3.4 Compression Scheme

In this section we provide a detailed description of our compression scheme for coding each parameter type described in the previous section.

3.4.1 Compression of Auto-correlation Function (ACF)

Auto-correlation function (ACF) is monotonically decreasing (except for texture having some inherent periodicity) from the central value \( \rho_{ac}(0,0) \) as shown in Figure 3.15. Depending on the rate at which the decrease happens, central samples of ACF may have a significant low frequency component. Therefore, discrete cosine transform-2 (DCT-2) shown in Figure 3.5 followed by a uniform quantization is applied to each matrix (Figure 3.16). Further, quantized DCT values are then appropriately scaled such that coefficient indices can be represented as integers which are suitable for CABAC encoding of coefficients (Figure 3.18). An example of texture grass type-1 Figure 3.36 is shown for the sub-band magnitude ACF parameters from scale 2, orientation 1 after transform and quantization of DCT values under variable step sizes in Figure 3.7 and Figure 3.8, also for orientation 2 in Figure 3.10 and Figure 3.11 and orientation 3 in Figure 3.13 and Figure 3.14. A significant number of coefficients are zero in each of the cases, thus making DCT suitable for the ACF compression.

![Overall coding chain for all the ACF parameters](image)

Sub-band magnitudes and partial reconstruction at each level of a multi-scale and multi-orientation representation of an image may have different amount of energy. Therefore, all the ACF matrices are normalized by their central value before applying transform and quantization to it. It is demonstrated that salient features in images give rise to large coefficients

\[
\begin{bmatrix}
0.37796 & 0.37796 & 0.37796 & 0.37796 & 0.37796 & 0.37796 \\
0.52112 & 0.41791 & 0.23192 & 0 & -0.23192 & -0.41791 & -0.52112 \\
0.48159 & 0.11894 & -0.33327 & -0.53452 & -0.33327 & 0.11894 & 0.48159 \\
0.41791 & -0.23192 & -0.52112 & 0 & 0.52112 & 0.23192 & -0.41791 \\
0.33327 & -0.48159 & -0.11894 & 0.53452 & -0.11894 & -0.48159 & 0.33327 \\
0.23192 & -0.52112 & 0.41791 & 0 & -0.41791 & 0.52112 & -0.23192 \\
0.11894 & -0.33327 & 0.48159 & -0.53452 & 0.48159 & -0.33327 & 0.11894
\end{bmatrix}
\]

![DCT-2 matrix 7x7 is used for the compression of ACF](image)
3.4 Compression Scheme

\[
ACF = \begin{bmatrix}
0.161556 & 0.216552 & 0.263615 & 0.282271 & 0.262580 & 0.214545 & 0.158612 \\
0.243902 & 0.327539 & 0.399744 & 0.427800 & 0.394777 & 0.317896 & 0.230539 \\
0.386285 & 0.544275 & 0.691515 & 0.752808 & 0.685595 & 0.533314 & 0.372309 \\
0.478520 & 0.694021 & 0.905555 & 1.000000 & 0.905555 & 0.694021 & 0.478520 \\
0.372309 & 0.533314 & 0.685595 & 0.752808 & 0.691515 & 0.544275 & 0.386285 \\
0.230539 & 0.317896 & 0.394777 & 0.427800 & 0.399744 & 0.327539 & 0.243902 \\
0.158612 & 0.214545 & 0.262580 & 0.282271 & 0.263615 & 0.216552 & 0.161556 \\
\end{bmatrix}
\]

Figure 3.6 Normalized central samples of ACF, sub-band scale-2 and orientation-1 (R-channel)

\[
DCT_{Quantized} = \begin{bmatrix}
30 & 0 & -7 & 0 & -1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-13 & 0 & 4 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
2 & 0 & -1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\]

Figure 3.7 Quantized DCT-2 coefficients obtained by using step size 0.1 for the ACF in Figure 3.6

in local spatial neighbourhoods, both adjacent scales and orientations [PS00a]. This implies that normalized matrices at adjacent scale, orientation or color could be similar. Therefore, the ACF matrices can be predicted from the other color components or adjacent scales or other orientations thus exploiting any redundancy between inter-color/scale/orientation of parameters for a better compression:

3.4.2 Candidates for Auto-correlation Function (ACF) Prediction

In our approach, identical transform (DCT-2) is applied to all normalized matrices and therefore the prediction is directly carried out on the DCT-2 coefficient which is followed by the quantization of the residual. A matrix could be predicted from neighbouring scales or orientations of same color, or from other color components (see Figure 3.17). Therefore, for each matrix, a candidate list is created using matrices from (already coded) neighbouring color, orientations and scale which is also available at decoder side. For each candidate, the resid-

\[
DCT_{Quantized} = \begin{bmatrix}
5 & 0 & -1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-2 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\]

Figure 3.8 Quantized DCT-2 coefficients obtained by using step size 0.6 for ACF in Figure 3.6
3 Static Textures: Perceptual Coding using Texture Synthesis

\[
ACF = \begin{bmatrix}
0.198902 & 0.297059 & 0.363714 & 0.332579 & 0.244085 & 0.172793 & 0.128287 \\
0.257364 & 0.424252 & 0.584311 & 0.558279 & 0.385720 & 0.246943 & 0.182453 \\
0.281766 & 0.488432 & 0.782045 & 0.846340 & 0.591788 & 0.340865 & 0.231807 \\
0.269108 & 0.442704 & 0.781947 & 1.000000 & 0.781947 & 0.442704 & 0.269108 \\
0.231807 & 0.340865 & 0.591788 & 0.846340 & 0.782045 & 0.424252 & 0.257364 \\
0.182453 & 0.246943 & 0.385720 & 0.558279 & 0.584311 & 0.424252 & 0.257364 \\
0.128287 & 0.172793 & 0.244085 & 0.332579 & 0.363714 & 0.246943 & 0.182453 \\
\end{bmatrix}
\]

Figure 3.9 Normalized central samples of the ACF, sub-band scale-2 and orientation-2 (R-channel).

\[
DCT_{Quantized} = \begin{bmatrix}
28 & 0 & -11 & 0 & 1 & 0 & 0 \\
0 & 3 & 0 & -2 & 0 & 0 & 0 \\
-8 & 0 & 5 & 0 & -1 & 0 & 0 \\
0 & -1 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\]

Figure 3.10 Quantized DCT-2 coefficients obtained by using step size 0.1 for ACF in Figure 3.9

\[
DCT_{Quantized} = \begin{bmatrix}
5 & 0 & -2 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 \\
-1 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\]

Figure 3.11 Quantized DCT-2 coefficients obtained by using step size 0.6 for ACF in Figure 3.9

\[
ACF = \begin{bmatrix}
0.106264 & 0.134494 & 0.209109 & 0.302829 & 0.333460 & 0.270911 & 0.180181 \\
0.143347 & 0.204435 & 0.352936 & 0.533771 & 0.560107 & 0.400500 & 0.235340 \\
0.190898 & 0.306012 & 0.568885 & 0.836641 & 0.768540 & 0.466367 & 0.255726 \\
0.235958 & 0.416805 & 0.769261 & 1.000000 & 0.769261 & 0.416805 & 0.235958 \\
0.255726 & 0.466367 & 0.768540 & 0.836641 & 0.568885 & 0.306012 & 0.190898 \\
0.235340 & 0.400500 & 0.560107 & 0.533771 & 0.352936 & 0.204435 & 0.143347 \\
0.180181 & 0.270911 & 0.333460 & 0.302829 & 0.209109 & 0.134494 & 0.106264 \\
\end{bmatrix}
\]

Figure 3.12 Central samples of ACF, sub-band scale-2 and orientation-3 (R-channel).
3.4 Compression Scheme

\[ DCT_{Quantized} = \begin{bmatrix}
27 & 0 & -11 & 0 & 1 & 0 & 0 \\
0 & -3 & 0 & 2 & 0 & 0 & 0 \\
-9 & 0 & 5 & 0 & -16 & 0 & 0 \\
0 & 1 & 0 & -1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix} \]

Figure 3.13 Quantized DCT-2 coefficients obtained by using step size 0.1 for ACF in Figure 3.12

\[ DCT_{Quantized} = \begin{bmatrix}
4 & 0 & -2 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & 0 & 0 & 0 \\
-1 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix} \]

Figure 3.14 Quantized DCT-2 coefficients obtained by using step size 0.6 for ACF in Figure 3.12

\[ ACF = \begin{bmatrix}
0.414372 & 0.500282 & 0.567430 & 0.623602 & 0.566782 & 0.495318 & 0.408604 \\
0.450702 & 0.542504 & 0.619013 & 0.680763 & 0.619248 & 0.539785 & 0.447220 \\
0.483600 & 0.582631 & 0.676742 & 0.770685 & 0.66886 & 0.583243 & 0.482301 \\
0.509576 & 0.613551 & 0.726453 & 1.000000 & 0.726453 & 0.613551 & 0.509576 \\
0.482301 & 0.583243 & 0.666886 & 0.770685 & 0.676742 & 0.582631 & 0.483600 \\
0.447220 & 0.539785 & 0.619248 & 0.680763 & 0.619013 & 0.542504 & 0.450702 \\
0.408604 & 0.495318 & 0.566782 & 0.623602 & 0.567430 & 0.500282 & 0.414372
\end{bmatrix} \]

Figure 3.15 Central part of the ACF 7 × 7 dimension (R-channel PCA).

\[ DCT = \begin{bmatrix}
4.020274 & -0.000000 & -0.649225 & 0.000000 & 0.042052 & 0.000000 & -0.107550 \\
0.000000 & 0.009128 & -0.000000 & -0.00313 & -0.000000 & 0.000587 & -0.000000 \\
-0.366517 & 0.000000 & 0.134161 & 0.000000 & -0.075191 & -0.000000 & 0.065844 \\
0.000000 & 0.000207 & -0.000000 & 0.004970 & 0.000000 & -0.005429 & -0.000000 \\
0.049088 & -0.000000 & -0.066850 & 0.000000 & 0.056678 & 0.000000 & -0.049699 \\
0.000000 & 0.01670 & 0.000000 & -0.004280 & -0.000000 & 0.004245 & -0.000000 \\
-0.064364 & 0.000000 & 0.054501 & 0.000000 & -0.044942 & 0.000000 & 0.037683
\end{bmatrix} \]

Figure 3.16 DCT-2 of the ACF given in Figure 3.15
ual is calculated by taking the difference between the candidate and original coefficients and subsequently the energy of the residual is calculated. The optimum candidate is selected as the candidate corresponding to the minimum energy. The complete list of the candidates as used in the prediction scheme is described in Table 3.2.

The selected candidate's information needs to be signalled to the decoder informing it which among multiple predictor is selected. In other words, the index of the predictor needs to be signalled to the decoder. Based on our observations, a few candidates have very high probability for being selected for prediction due to high correlation. Therefore an optimized variable length binarization scheme is designed to efficiently code the index of the candidate i.e. use less bins to binarize most probable candidate and more bins to binarize less probable candidates. This is achieved by creating a specific candidate order list. An example is shown from the Table 3.2, where matrix corresponding to scale-2, orientation-3, color-3 has 5 candidates. The candidate's probability of being preferred for prediction over the other candidates defines the candidate order which is TU binarized as shown in Table 3.1. The scheme is adapted such that the most likely chosen candidate has a shortest code word length i.e. cand-1, while the least probable candidate has longest code word length i.e. cand-5.

<table>
<thead>
<tr>
<th>Index</th>
<th>Candidate's Name</th>
<th>TU bins</th>
</tr>
</thead>
<tbody>
<tr>
<td>cand-1</td>
<td>color-2</td>
<td>0</td>
</tr>
<tr>
<td>cand-2</td>
<td>color-1</td>
<td>10</td>
</tr>
<tr>
<td>cand-3</td>
<td>orientation-2</td>
<td>110</td>
</tr>
<tr>
<td>cand-4</td>
<td>orientation-1</td>
<td>1110</td>
</tr>
<tr>
<td>cand-5</td>
<td>scale-1 (ori-3, col-3)</td>
<td>11110</td>
</tr>
</tbody>
</table>

Table 3.1 Candidate order list with TU binarization for coding the index of candidates in accordance with their probability of being selected for prediction. The candidate list for predicting scale-2, orientation-3 and color-3 is shown above.

In order to estimate the probability for each candidate lets consider a case for scales=3, orientation=3, color = 3, in this case we get 27 sub-band ACF matrices( 9 at each scale)+12 low-pass ACF(3 from each level including the low-pass) and 3 color channel PCA
ACF \[\text{PS00a}\]. This totals 42 ACF matrices and among these Matrices, cand-1, cand-2, cand-3, cand-4 and cand-5 is selected respectively N1, N2, N3, N4, and N5 times \((N1 + N2 + N3 + N4 + N5 = 42)\). Therefore, we can estimate the probability for each candidate, \(P(\text{cand-1}) = \frac{N1}{42}, P(\text{cand-2}) = \frac{N2}{42} \ldots\). In our case the highest preference is given to the candidate color as RGB channels carry a highly correlated signal.

<table>
<thead>
<tr>
<th>Scale</th>
<th>Orientation</th>
<th>Color</th>
<th>Num. of Candidates</th>
<th>Candidates(Sca., Ori., Col.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>PCA-color1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>(1,1,1) and PCA-color2</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>(1,1,2), (1,1,1) and PCA-color3</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>(1,1,1) and PCA-color1</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>(1,2,1), (1,1,2) and PCA-color2</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>(1,2,2), (1,2,1), (1,1,3) and PCA-color3</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>(1,2,1), (1,1,1) and PCA-color1</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>(1,3,1), (1,2,2), (1,1,2) and PCA-color2</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>3</td>
<td>5</td>
<td>(1,3,2), (1,3,1), (1,2,3), (1,1,3) and PCA-col3</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>(1,1,1)</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>(2,1,1) and (1,1,2)</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>(2,1,2), (2,1,1) and (1,1,3)</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>(1,2,1) and (1,2,1)</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>(2,2,1), (1,2,1) and (1,2,2)</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>(2,2,2), (2,2,1), (1,2,1) and (1,2,3)</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>(2,2,1), (2,1,1) and (1,3,1)</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>(2,3,1), (2,2,2), (2,1,2) and (1,3,2)</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>3</td>
<td>5</td>
<td>(2,3,2), (2,3,1), (2,2,3), (2,1,3) and (1,3,3)</td>
</tr>
</tbody>
</table>

Table 3.2 Candidate list at each point of prediction is shown for a two level pyramid with 3 orientations and 3 color channels. The candidate list is updated each time for prediction as shown above, so as to place the candidates in the right order of the candidate list. Highest preference is given to the color, followed by adjacent orientations and scale. The index of the most probable candidate has the shortest code word length and is coded with TU binarization.

### 3.4.3 DCT property for Auto-correlation Function (ACF)

The DCT when applied to the ACF as shown in Figure 3.18 leads to zero valued alternate diagonals. These diagonals can be skipped during coefficient scanning as shown in Figure 3.19, thus reducing the overhead to be coded. A mathematical proof is derived to show why the alternate diagonals are zero valued.

The symmetry property of ACF Matrix is given by:

\[
\rho(i, j) = \rho(-i, -j) \quad (3.1)
\]
3 Static Textures: Perceptual Coding using Texture Synthesis

We define an ACF matrix \( f(m, n) \) of size \( M \times N \), where \( M \) and \( N \) are both odd. Extracting only the central part of auto-correlation, for symmetry the following holds:

\[
f(m, n) = f(M - 1 - m, N - 1 - n)
\]

(3.2)

Now, DCT of the ACF matrix is given by:

\[
F(p, q) = \alpha_p \alpha_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) \cos \left( \frac{\pi(2m + 1)p}{2M} \right) \cos \left( \frac{\pi(2n + 1)q}{2N} \right)
\]

(3.3)

where

\[
\alpha_p = \begin{cases} \frac{1}{\sqrt{M}}, & p = 0 \\ \frac{1}{\sqrt{2M}}, & 1 \leq p \leq M - 1 \end{cases}
\]

(3.4)

and

\[
\alpha_q = \begin{cases} \frac{1}{\sqrt{N}}, & q = 0 \\ \frac{1}{\sqrt{2N}}, & 1 \leq q \leq N - 1 \end{cases}
\]

(3.5)

Computation of \( F(p, q) \) can be divided into two parts: Contribution from centre term and non-center terms. Contribution from any non-center term \((m, n)\) and its diagonally opposite term \((M - 1 - m, N - 1 - n)\):

\[
\Rightarrow \alpha_p \alpha_q f(m, n) \left[ \cos \left( \frac{\pi(2m + 1)p}{2M} \right) \cos \left( \frac{\pi(2n + 1)q}{2N} \right) \right. \\
+ \cos \left( \frac{\pi(2(M - 1 - m) + 1)p}{2M} \right) \cos \left( \frac{\pi(2(N - 1 - n) + 1)q}{2N} \right) \\
\text{taking, } \frac{\pi(2m + 1)p}{2M} = \alpha, \frac{\pi(2n + 1)q}{2N} = \beta
\]

\[
\Rightarrow \alpha_p \alpha_q f(m, n) \left[ \cos \alpha \cos \beta + \cos(\pi p - \alpha) \cos(\pi q - \beta) \right]
\]

(3.6)

if \((p+q)\) is odd,

\[
\Rightarrow \alpha_p \alpha_q f(m, n) \left[ \cos \alpha \cos \beta + (-\cos \alpha \cos \beta) \right]
\]

(3.7)

Every diagonal pair contributes to zero, for \((p+q)\) is odd.

Now, for center point \(((M - 1)/2, (N - 1)/2)\):

\[
\Rightarrow \alpha_p \alpha_q f(m, n) \left[ \cos \left( \frac{\pi(2(M - 1)/2 + 1)p}{2M} \right) \cos \left( \frac{\pi(2(N - 1)/2 + 1)q}{2N} \right) \right]
\]

(3.8)

\[
\Rightarrow \alpha_p \alpha_q f(m, n) \cos \frac{\pi p}{2} \cos \frac{\pi q}{2}
\]

(3.9)

Also if \((p+q)\) is odd, the center term is always zero. Therefore, DCT of ACF parameters have zeros in alternate diagonal columns.
3.4 Compression Scheme

\[ DCT_{\text{Quantized}} = \begin{bmatrix}
40 & 0 & -6 & 0 & 0 & 0 & -1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-4 & 0 & 1 & 0 & -1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -1 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-1 & 0 & 1 & 0 & 0 & 0 & 0 \\
\end{bmatrix} \]

**Figure 3.18** Quantized DCT-2 coefficients obtained from the raw DCT-2 coefficients in Figure 3.16 by using step size 0.1 and subsequently rounding off to the nearest integer.

**Figure 3.19** Scanning order starting from the last significant coefficient up to DC value. Only alternate diagonals are scanned as the remaining are zero valued and therefore they are skipped.

3.4.4 Significant Coefficient Scan

Diagonal scanning of the residual (for optimum candidate) is done (starting from last non-zero coefficient and back to DC coefficient) as shown in Figure 3.19, which is similar to the coefficient scan order used in HEVC [Sul+12]. The 1d vector formed after scan is shown in Figure 3.20. The position of the last significant coefficient is coded first using fixed length binarization. We also send a coefficient significant flag that informs the decoder which location have non-zero values. Further, for each non-zero coefficient in the 1d array, its absolute value-1 is coded using CABAC Figure 3.22 and its sign coded separately. For the magnitude part, Exponential Golomb binarization scheme of adaptive order setting is used based on the location of a coefficient. As shown in Figure 3.18 the magnitude range is highly varying between top left and bottom right. Therefore, for the top left location (0,0) i.e., the DC value is always coded using order 2 (i.e. encode larger numbers in fewer bits Golomb coding), its neighbour coefficient at location (0, 2) and (2, 0) are coded with order 1 (mid-range values) and remaining coefficients with order 0 (small values).

\[ DCT_{\text{Quantized scan}} = [1 \ 0 \ 1 \ 0 \ 1 \ -1 \ 0 \ -1 \ 0 \ 0 \ -1 \ 0 \ 0 \ -4 \ 0 \ -6 \ 40] \]

**Figure 3.20** DCT coefficients given in Figure 3.18 are arranged in a 1d array after diagonal scanning, starting from the last significant coefficient in the left→right order until last DC coefficient.

It is to be noted that due to symmetry of ACF for each matrix nearly half of the coefficients are zero in DCT domain Section 3.4.3 in particular alternate diagonal columns are
3 Static Textures: Perceptual Coding using Texture Synthesis

Coefficient 1 0 1 0 1 −1 0 −1 0 −1 0 0 1 0 0 −4 0 −6 40
Sign. flag − 0 1 0 1 1 0 1 0 1 0 1 0 0 1 0 0 1 0 1 1
Sign flag 0 − 0 − 0 1 − 1 − 1 − 1 − 0 − − 0 − − 1 − 1 0
Rem. level − − 0 − 0 0 − 0 − 0 − 0 − 0 − 0 − 0 − 0 − 3 − 5 39

Figure 3.21 Coefficient scan of the ACF

\[ DCT_{coeff \ remain\ level} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 3 & 5 & 39 \end{bmatrix} \]

Figure 3.22 Absolute DCT coefficients in 1d after subtracting 1 from absolute value at each significant location including the LSC coefficient.

Figure 3.23 ACF compression gain achieved using our prediction at various quantization levels as compared to the coding without using prediction candidates.
zero. Therefore, every alternate diagonal column can be skipped while doing the scan. This method further reduces the number of coefficients to be coded.

The coding order of the ACF matrices starts with PCA of color channel which are coded followed by partial reconstruction at each scale (including low-pass band). This is followed by sub-band magnitude matrices. Overall, the prediction strategy (prediction from neighbouring scale or orientation or color) provides significant bit-rate savings for auto-correlation parameters as shown [Figure 3.23].

### 3.4.5 Compression for Intra-Scale Cross-correlation

Intra-scale cross correlation is computed on sub-band magnitude and on real part of sub-band coefficients. Intra-scale correlation has an inherent symmetry:

\[ CC(i, j) = CC(j, i) \]  \hspace{1cm} (3.11)

where \( CC(i, j) \) is the correlation between sub-bands \( i \) and \( j \) (see [Figure 3.24]). The normalized cross-correlated coefficients are derived from actual coefficients by:

\[ CC(i, j)_{\text{normalized}} = \frac{CC(i, j)}{\sqrt{CC(i, i) \cdot CC(j, j)}} \]  \hspace{1cm} (3.12)

\( CC(i, j)_{\text{normalized}} \) is considerably higher when sub-bands \( i \) and \( j \) are of same color and orientation as shown in [Figure 3.25]. Therefore, given a set of \( CC(i, j)_{\text{normalized}} \) values, we have ordered the values in a specific way to form a 1d vector in order to reduce the high frequency. Firstly, \( CC(i, j)_{\text{normalized}} \) values of same color (different orientation) sub-bands are arranged as shown in [Figure 3.26] and [Figure 3.27]. Priority is given to values where relative difference of orientations between two bands is less. This is followed by correlation values of same orientations (different color) sub-bands as shown in [Figure 3.28] and [Figure 3.29]. In the end, \( CC(i, j)_{\text{normalized}} \) values of different orientations and different color sub-bands are arranged as shown in [Figure 3.30] and [Figure 3.31]. The 1d vector is transformed using DCT-2 operation and quantized as shown in [Figure 3.32]. The main reason for this ordering is to rearrange the cross-correlation values in descending order. This reduces high frequency components which tends to provide an efficient energy compaction. The position for the correlation values is not coded as this coding order is kept fixed for all the textures. The coefficients in the coarser scale are further predicted from adjacent fine scale coefficients as they are already coded. Thus exploiting any similarity between inter scale coefficients if present. The absolute value of the DCT-2 coefficients in [Figure 3.32] is coded using exponential Golomb coding with order zero and sign is coded separately.
### Figure 3.24 Intra-scale cross-correlation between sub-bands magnitude at scale-1

<table>
<thead>
<tr>
<th></th>
<th>R-Ori-1</th>
<th>R-Ori-2</th>
<th>R-Ori-3</th>
<th>G-Ori-1</th>
<th>G-Ori-2</th>
<th>G-Ori-3</th>
<th>B-Ori-1</th>
<th>B-Ori-2</th>
<th>B-Ori-3</th>
</tr>
</thead>
<tbody>
<tr>
<td>R-Ori-1</td>
<td>1.000000</td>
<td>0.296848</td>
<td>0.293028</td>
<td>0.269815</td>
<td>0.243344</td>
<td>0.341710</td>
<td>0.181436</td>
<td>0.122541</td>
<td>0.115670</td>
</tr>
<tr>
<td>R-Ori-2</td>
<td>0.296848</td>
<td>1.000000</td>
<td>0.503601</td>
<td>0.68105</td>
<td>0.219334</td>
<td>0.341710</td>
<td>0.181436</td>
<td>0.122541</td>
<td>0.115670</td>
</tr>
<tr>
<td>R-Ori-3</td>
<td>0.293028</td>
<td>0.503601</td>
<td>1.000000</td>
<td>0.688556</td>
<td>0.094333</td>
<td>0.231956</td>
<td>0.031327</td>
<td>0.037422</td>
<td>0.136594</td>
</tr>
<tr>
<td>G-Ori-1</td>
<td>0.269815</td>
<td>0.68105</td>
<td>0.688556</td>
<td>1.000000</td>
<td>0.269815</td>
<td>0.243344</td>
<td>0.228840</td>
<td>0.073414</td>
<td>0.081854</td>
</tr>
<tr>
<td>G-Ori-2</td>
<td>0.243344</td>
<td>0.219334</td>
<td>0.094333</td>
<td>0.269815</td>
<td>1.000000</td>
<td>0.341710</td>
<td>0.075170</td>
<td>0.153602</td>
<td>0.097752</td>
</tr>
<tr>
<td>G-Ori-3</td>
<td>0.341710</td>
<td>0.341710</td>
<td>0.341710</td>
<td>0.269815</td>
<td>0.243344</td>
<td>1.000000</td>
<td>0.075170</td>
<td>0.153602</td>
<td>0.097752</td>
</tr>
<tr>
<td>B-Ori-1</td>
<td>0.181436</td>
<td>0.181436</td>
<td>0.181436</td>
<td>0.269815</td>
<td>0.243344</td>
<td>0.341710</td>
<td>1.000000</td>
<td>0.075170</td>
<td>0.153602</td>
</tr>
<tr>
<td>B-Ori-2</td>
<td>0.122541</td>
<td>0.122541</td>
<td>0.122541</td>
<td>0.269815</td>
<td>0.243344</td>
<td>0.341710</td>
<td>0.075170</td>
<td>1.000000</td>
<td>0.075170</td>
</tr>
<tr>
<td>B-Ori-3</td>
<td>0.115670</td>
<td>0.115670</td>
<td>0.115670</td>
<td>0.269815</td>
<td>0.243344</td>
<td>0.341710</td>
<td>0.075170</td>
<td>0.075170</td>
<td>1.000000</td>
</tr>
</tbody>
</table>

Figure 3.25 Normalized cross-correlation obtained from Figure 3.24 using Equation 3.12

### Figure 3.26 Cross-correlation between same color and different orientation (values inside rectangular blocks).

<table>
<thead>
<tr>
<th></th>
<th>R-Ori-1</th>
<th>R-Ori-2</th>
<th>R-Ori-3</th>
<th>G-Ori-1</th>
<th>G-Ori-2</th>
<th>G-Ori-3</th>
<th>B-Ori-1</th>
<th>B-Ori-2</th>
<th>B-Ori-3</th>
</tr>
</thead>
<tbody>
<tr>
<td>R-Ori-1</td>
<td>0.296848</td>
<td>0.293028</td>
<td>0.269815</td>
<td>0.341710</td>
<td>0.243344</td>
<td>0.181436</td>
<td>0.122541</td>
<td>0.115670</td>
<td></td>
</tr>
<tr>
<td>R-Ori-2</td>
<td>0.293028</td>
<td>0.503601</td>
<td>0.68105</td>
<td>0.219334</td>
<td>0.104964</td>
<td>0.014301</td>
<td>0.125675</td>
<td>0.021171</td>
<td></td>
</tr>
<tr>
<td>R-Ori-3</td>
<td>0.269815</td>
<td>0.68105</td>
<td>1.000000</td>
<td>0.269815</td>
<td>0.104964</td>
<td>0.014301</td>
<td>0.125675</td>
<td>0.021171</td>
<td></td>
</tr>
<tr>
<td>G-Ori-1</td>
<td>0.341710</td>
<td>0.219334</td>
<td>0.094333</td>
<td>0.269815</td>
<td>0.104964</td>
<td>0.014301</td>
<td>0.125675</td>
<td>0.021171</td>
<td></td>
</tr>
<tr>
<td>G-Ori-2</td>
<td>0.243344</td>
<td>0.104964</td>
<td>0.269815</td>
<td>0.269815</td>
<td>0.104964</td>
<td>0.014301</td>
<td>0.125675</td>
<td>0.021171</td>
<td></td>
</tr>
<tr>
<td>G-Ori-3</td>
<td>0.181436</td>
<td>0.014301</td>
<td>0.269815</td>
<td>0.269815</td>
<td>0.104964</td>
<td>0.014301</td>
<td>0.125675</td>
<td>0.021171</td>
<td></td>
</tr>
<tr>
<td>B-Ori-1</td>
<td>0.122541</td>
<td>0.122541</td>
<td>0.122541</td>
<td>0.269815</td>
<td>0.269815</td>
<td>0.269815</td>
<td>0.181436</td>
<td>0.115670</td>
<td></td>
</tr>
<tr>
<td>B-Ori-2</td>
<td>0.115670</td>
<td>0.115670</td>
<td>0.115670</td>
<td>0.269815</td>
<td>0.269815</td>
<td>0.269815</td>
<td>0.181436</td>
<td>0.115670</td>
<td></td>
</tr>
<tr>
<td>B-Ori-3</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td>0.000000</td>
<td></td>
</tr>
</tbody>
</table>

Figure 3.27 Cross-correlation between same color and different orientation organized in a 1D format.

\[
\begin{bmatrix}
0.296848 & 0.503601 & 0.293028 & 0.269815 & 0.341710 & 0.243344 & 0.181436 & 0.122541 & 0.115670 \\
\end{bmatrix}
\]
3.4 Compression Scheme

\[
\begin{bmatrix}
\text{R-Ori-1} & \text{R-Ori-2} & \text{R-Ori-3} & \text{G-Ori-1} & \text{G-Ori-2} & \text{G-Ori-3} & \text{B-Ori-1} & \text{B-Ori-2} & \text{B-Ori-3} \\
\text{R-Ori-1} & 1.000000 & 0.296848 & 0.293028 & 0.236150 & 0.104120 & 0.110491 & 0.148641 & 0.045443 & 0.020182 \\
\text{R-Ori-2} & 0.296848 & 1.000000 & 0.503601 & 0.068105 & 0.219334 & 0.104964 & 0.014301 & 0.125675 & 0.021171 \\
\text{R-Ori-3} & 0.293028 & 0.503601 & 1.000000 & 0.068556 & 0.094333 & 0.231956 & 0.013327 & 0.037422 & 0.136594 \\
\text{G-Ori-1} & 0.236150 & 0.068105 & 0.068556 & 1.000000 & 0.269815 & 0.243344 & 0.228840 & 0.073414 & 0.081854 \\
\text{G-Ori-2} & 0.104120 & 0.219334 & 0.094333 & 0.269815 & 1.000000 & 0.341710 & 0.075170 & 0.153602 & 0.097752 \\
\text{G-Ori-3} & 0.110491 & 0.104964 & 0.231956 & 0.341710 & 1.000000 & 0.074856 & 0.097082 & 0.187183 \\
\text{B-Ori-1} & 0.148641 & 0.013327 & 0.013327 & 0.075170 & 0.074856 & 1.000000 & 0.181436 & 0.115670 \\
\text{B-Ori-2} & 0.045443 & 0.125675 & 0.037422 & 0.075170 & 0.153602 & 0.097082 & 1.000000 & 0.122541 \\
\text{B-Ori-3} & 0.020182 & 0.021171 & 0.136594 & 0.097752 & 0.187183 & 0.115670 & 0.122541 & 1.000000 \\
\end{bmatrix}
\]

Figure 3.28 Cross-correlation between different color and same orientation (values inside rectangular blocks).

\[
[0.236150 \ 0.219334 \ 0.231956 \ 0.228840 \ 0.153602 \ 0.187183 \ 0.148641 \ 0.125675 \ 0.136594]
\]

Figure 3.29 Cross-correlation between different color and same orientation organized in a 1D format.

\[
\begin{bmatrix}
\text{R-Ori-1} & \text{R-Ori-2} & \text{R-Ori-3} & \text{G-Ori-1} & \text{G-Ori-2} & \text{G-Ori-3} & \text{B-Ori-1} & \text{B-Ori-2} & \text{B-Ori-3} \\
\text{R-Ori-1} & 1.000000 & 0.296848 & 0.293028 & 0.236150 & 0.104120 & 0.110491 & 0.148641 & 0.045443 & 0.020182 \\
\text{R-Ori-2} & 0.296848 & 1.000000 & 0.503601 & 0.068105 & 0.219334 & 0.104964 & 0.014301 & 0.125675 & 0.021171 \\
\text{R-Ori-3} & 0.293028 & 0.503601 & 1.000000 & 0.068556 & 0.094333 & 0.231956 & 0.013327 & 0.037422 & 0.136594 \\
\text{G-Ori-1} & 0.236150 & 0.068105 & 0.068556 & 1.000000 & 0.269815 & 0.243344 & 0.228840 & 0.073414 & 0.081854 \\
\text{G-Ori-2} & 0.104120 & 0.219334 & 0.094333 & 0.269815 & 1.000000 & 0.341710 & 0.075170 & 0.153602 & 0.097752 \\
\text{G-Ori-3} & 0.110491 & 0.104964 & 0.231956 & 0.341710 & 1.000000 & 0.074856 & 0.097082 & 0.187183 \\
\text{B-Ori-1} & 0.148641 & 0.013327 & 0.013327 & 0.075170 & 0.074856 & 1.000000 & 0.181436 & 0.115670 \\
\text{B-Ori-2} & 0.045443 & 0.125675 & 0.037422 & 0.075170 & 0.153602 & 0.097082 & 1.000000 & 0.122541 \\
\text{B-Ori-3} & 0.020182 & 0.021171 & 0.136594 & 0.097752 & 0.187183 & 0.115670 & 0.122541 & 1.000000 \\
\end{bmatrix}
\]

Figure 3.30 Cross-correlation between different color and different orientation (values inside rectangular blocks).

\[
[0.104120 \ 0.104964 \ 0.068105 \ 0.094333 \ 0.110491 \ 0.068556 \ 0.073414 \ 0.097752 \ 0.075170 \\
0.097082 \ 0.081854 \ 0.074856 \ 0.045443 \ 0.021171 \ 0.014301 \ 0.037422 \ 0.020182 \ 0.013327]
\]

Figure 3.31 Cross-correlation between different color and different orientation organized in a 1D format.

\[
[1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]
\]

Figure 3.32 Quantized DCT coefficients (step-size=0.1) in a 1d vector corresponding to Figure 3.25.
3.4.6 Compression for Inter-scale Cross-correlation

For inter-scale cross-correlation, similar symmetry to the one in Section 3.4.5 does not hold as sub-bands \( i \) and \( j \) are from different scales. Here, the correlation values \( CC(i, j) \) are normalized by dividing all the values by maximum value. The correlation values are arranged in the 2d form. It is transformed using DCT operation. The coefficients in the coarser scale are further predicted from adjacent fine scale coefficients exploiting any possible similarity between coefficients at adjacent scales.

3.4.7 Compression of Marginal Statistics

Marginal statistics are scalar values. \( N \)-significant bits are used for representing scalars (where \( N \) can be set empirically, in our case \( N=8 \)). Consequently, a scalar value \( V \) can be represented by its significant digit \( D \) (10 bits) and exponent \( E \) (base 10)(3 bits) in the following manner:

\[
V = D \times 10^E
\]  

(3.13)

The absolute value and sign of both \( D \) and \( E \) is coded separately. Some parameters are inherently positive (input image’s mean, variance, range of intensity value, mean of sub-band magnitude, variance of high pass etc.). For such terms, sign is not coded, as it is unsigned. Also exponent of some parameters can be predicted from other color component or other orientations, as the values are in similar range.

3.5 Texture Synthesis

In this section a brief summary is given on the texture synthesis process (for complete details on texture synthesis algorithm and parameter adjustments refer to [PS99; PS00a]). The synthesis approach used here is based upon the concept of alternating projections on to the constraint surfaces described in [PS00a]. The process starts with an image of Gaussian white noise. The image is decomposed into a complex steerable pyramid and statistical constraints are forced by sequentially and repeatedly projecting onto the set of images satisfying each constraint. A recursive coarse-to-fine process imposes the statistical constraints on the low-pass and band-pass sub-bands as shown in Figure 3.33. In our case 25 iterations of synthesis are used in order to synthesize a decent visual quality image.

![Figure 3.33](image)

Figure 3.33 Block diagram for texture synthesis algorithm. The above figure is reproduced from [PS00a].
3.6 Experimental Results

Perceptual coding is best evaluated perceptually. Therefore, in order to validate the quality of texture synthesis over conventional intra prediction as used in HEVC, a pairwise comparison of our visual results with HEVC intra prediction is provided at the same rates. It is left to the reader to judge the quality of the synthesis. Textures need relatively larger bit-rate to code and often show blocking and blurring at lower rates therefore, QPs in the range of 44 - 51 were selected to judge the perceptual performance at very low rate points. HM 16.2 was used as a reference software [Jo]. Objective assessment was also performed but due to limited performance of today’s quality assessment methods like PSNR, SSIM and VIF [SBV05, Wan+04], it is not possible to evaluate reliably the visual quality of synthesized texture. An example of the objective assessment is shown in Figure 3.34. The quality of synthesis is measured at four different rate points using the PSNR, SSIM and VIF as quality metrics. From the objective assessment it can be concluded that quality of synthesis is not comparable with HEVC at all the given rate points. Also, a pairwise comparison of the perceptual quality is provided in Figure 3.35. In our opinion the subjective quality of the texture synthesized at the same rate is better than the HEVC coded.

It is to be noted that a fixed patch-size of 256 × 256 is analysed and synthesized. In general, texture synthesis scheme can only benefit under the assumption that textures are homogeneous over a region of certain size in a picture otherwise, cost of parameter coding will exceed the cost of coding the residual after intra prediction. However, for certain textures results for patch-size of 128 × 128 and 64 × 64 are also provided for visual assessment see Figure A.12 and Figure A.16. Patch-size smaller then 64 × 64 is not possible with our scheme due to limited compression achieved on synthesis parameters.

![Figure 3.34 Objective comparison for the grass type-1 texture patch (256×256 pixels).](image)

3.6.1 Subjective Evaluations

The comparisons are made at very low bit-rate (QP=44 and beyond). Perceptual results at two different rate points are provided for most textures. To match the bit-rates between the synthesis coding and HEVC variable quantization using step-sizes ranging between 0.01 - 0.63 is applied on the synthesis parameters. It is easily perceivable from Figure 3.35 (also see annex: A for more results) that at lower rates HEVC coded texture show blocking and blurring artefacts. It is an interesting fact that fine details inside textures if not represented with
pixel accuracy but, if it still holds a similar statistical characteristics as described in [PS00a], it is perceived as equivalent by human eye. This simple concept provides a compact representation of textures than actually reproducing textures with pixel fidelity for patch-size of 256 × 256 and 128 × 128. Blocking and blurring artefacts are completely absent in synthesis based coding which makes the perceptual quality significantly better over HEVC at the same rate. For patch-size of 64 × 64 visual quality is very limited as shown in Figure A.15 and Figure A.16. This is due to the very heavy quantization applied to all the parameters in order to match the HEVC's bit-rate.

3.6.2 Effects of Quantizing Parameters on Texture Synthesis

Compression of synthesis parameters using step sizes ranging between 0.01 - 0.63 produce varying visual artefacts. The type of artefact appearing depend on the texture type i.e having random or regular structures at the granular level. To keep the experiment settings uniform fixed quantization is used for all the parameters belonging to patch-size 256 × 256. For smaller patch sizes 128 × 128 and 64 × 64 a separate quantization is applied on the ACF and cross-correlation parameters in order to match the bits/pixels of the HEVC coded texture. In case of textures that constitute of random granular structures an example for grass type-1 texture synthesis result is shown in Figure 3.36 at four different step-sizes. At step-size= 0.01 the synthesized texture has no strong artefacts and is very similar to the synthesis without quantization. With increase in the step size, strength of visual artefacts also starts to increase such as strong noticeable contouring pointing in arbitrary orientation as shown in Figure 3.36e. For more visual results see annex A.

In case of textures that constitute of regular granular structures (usually such structure is distinguishable when seen from a short distance) an example for carpet type-3 is shown in Figure 3.37. At step-size= 0.01 the synthesized texture has no strong artefacts (except the ones originating from synthesis itself) and the granular structure of texture stays intact as shown in Figure 3.37c similar to result with no quantization. With increase in the step size, strength of noticeable artefacts also starts to increase such as, loss of regularity in granular structure and arbitrary oriented contours are seen as shown in Figure 3.37e. The regular granular structure inside the texture is generally more sensitive to the human eye (details are perceptually distinguishable when seen from short distance) and therefore any distortion at the granular level becomes annoying. Therefore, we have provided limited results for such type of content. For more such visual results see pebbles Figure A.8 and denim Figure A.9

3.6.3 Limitations of Texture Synthesis

It is a very interesting fact that texture synthesis performs best on content that is the most challenging for intra-prediction i.e. textures with random granular structure. The scheme described in [PS00a] works decently for textures with random granular structures as shown in Figure 3.36, A.1, A.2, A.3, A.5, A.6, A.7, A.10 and A.11. In such cases perceptually indiscernible variations are randomly distributed over the patch and Human Visual System (HVS) is well exploited using texture synthesis. Visual quality of synthesized textures starts to have limited performance in the case of textures with discernible granularity and periodicity as shown in Figure 3.37, A.4 and A.9. In such textures the HVS is able to perceive the synthesis
3.7 Conclusion

We have provided an investigation into the compression of synthesis parameters required for the texture synthesis coding at the decoder side. The results provided in this thesis clearly prove that a much better visual quality can be achieved for certain class of textures at very low bit-rates when compared to the state-of-the-art codec HEVC provided, the area to be synthesized is large enough. Parameter quantization with varying step-sizes and its effects on the texture synthesis quality are also part of the investigation. Further improvement in this work are necessary to make it practical i.e., e.g. making the synthesis feasible for small block-sizes like 16×16 which can be achieved by discarding any unnecessary parameters from the complete set of parameters and adaptively quantizing each parameter type. This would require a detailed analysis on texture properties and descriptors (parameters) which would also help us in limiting the number of scales and orientation depending on the texture. Shifting the synthesis model to YUV domain rather then RGB as used in this thesis, may also give some improvement in compression.

Overall, integration of texture synthesis with conventional video coding still remains a challenge due to various identified issues as already discussed in Section 3.6.3. The parametric texture synthesis approach based on sub-band statistics works decently only for a limited range of content and therefore, its application in a real world coding scenario may not be highly beneficial. With recent improvements in the texture modeling using neural networks [GEB15] the quality of synthesis is significantly improved, such models may be better suited for future video coding as it can synthesize a wide range of textured content. In the near future with improvements in quality metrics that will better model the human vision, there is a possibility that such a technology can be integrated in to a video codec. A final conclusion is drawn in Chapter 5.

artefacts and as a result the visual quality appears to be relatively low as compared to the others.

Overall, the synthesis algorithm used here provides decent visual quality results only for a limited range of content and this restricts the use of such a technology in a video coding scenario beyond research purposes. In a practical video coding scenario there may be a much wider range of content (textures) and developing a fully integrated synthesis model (for a limited texture range) inside a video coding framework may have a very little room for potential gains. Also, due to large number of parameters it is not feasible to use synthesis for small block-sizes like 8×8 or 16×16 or 32×32 as used in video coding algorithms. Evaluating the quality of the synthesized texture using quality metrics is another fundamental issue from the perspective of video coding. Fitting a texture synthesis as an optional prediction mode into a video codec will certainly require quality assessment for rate-distortion optimization, without which it cannot become a practical model.
Figure 3.35 A subjective comparison is shown at a similar bits/pixel (bpp) for the grass type-1 texture patch (256×256 pixels).
Figure 3.36 Effects of quantization on the synthesized grass type-1 texture.
Figure 3.37 Effects of quantization on the synthesized carpet type-3 texture.
4 Reduced Resolution Coding with Detail Reconstruction in B-Pictures

4.1 Introduction

Distribution of ultra high definition (UHD) or 4K video requires transmission of extremely large volumes of data. State-of-the-art video compression technology has reduced the size of bitstreams sufficiently thus making distribution of UHD feasible. With increase in the computation power, complex and high performance video codecs have become a reality like H.265, VP9 etc.; at the same time there is a significant increase in the volume of data due to 4 and 8K resolutions together with high frame rates (60-120Hz). This increase in the spatial and temporal resolution poses challenges for low bit-rate coding when the bandwidth is very limited due to significant increase in the volume of data to be transmitted. The challenge is even larger when the content constitutes of high spatial and temporal frequencies e.g. dynamic textures or only high temporal frequencies e.g. scenes consisting of rapid motion etc.

4.2 Dynamic Texture and Coding Challenges

Handling complex motions has always remained challenging for video compression \[Bal12, ZB11\]. In this chapter, we primarily focus on dynamic textures but not limited to. In general, dynamic texture is a time-varying visual pattern that may or may not exhibit certain temporal stationarity. Perceptually, dynamic textures can be further categorized as discrete and continuous \[PFH10\]. Discrete dynamic textures have a discernible appearance in motion e.g. leaves fluttering in the wind, on the other hand continuous one is characterized with continuous deformable media or are practically indiscernible e.g. water waves, flames from a burning material, waving field of grass seen from considerable distance etc. Performance of the state-of-the-art compression technologies like HEVC and JEM is very limited when coding dynamic textures. The encoding analysis performed on homogeneous dynamic texture patches in \[Afo+16\] points to the fact that the majority of bits are spent on residual coding. This implies that the bits used for coding additional information such as motion vectors and mode signalling have only minor impact to the final bit-rate of the encoded sequence.

Continuous dynamic textures are often coded using intra-mode selection as shown in Figure 4.1 whereas, discrete dynamic textures are characterized with a majority of small block partitions as shown in Figure 4.2 but both have high energy in residual \[Afo+16, TNW16\]. All these facts lead us to the conclusion that conventional approach of motion-compensation is not well suited for such signals. In a random access coding scenario when coding a standard video sequence, the pictures corresponding to the two highest \(t_{\text{q}}\)'s usually account for a 2-7% of the total bit-rate needed to code the entire sequence. This may slightly vary depending upon the nature of the content and presence of motion. Such low bit-rate needed
to code these pictures indicates an effective compression scheme. In case of dynamic textures B-pictures corresponding to the two highest \( t_{id} \)'s may account for 20-40% of the total bit-rate, which indicates us that the performance of the codec is very limited.

To show that B-pictures are relatively expensive to code in dynamic textures we have used sequences from the Homtex database \cite{Afo+16}. This database consists of both continuous and discrete homogeneous textured sequences with spatial resolution of 256 \( \times \) 256. A test is performed using random access scenario of JEM, encoding a temporal resolution of 120 frames for majority of the sequences from Homtex at four different rate points (QPs 22, 27, 32 and 37), this gives us the standard encoding bit-rates for full resolution. Further, following the same test conditions (same QPs) we encode the same sequences but with the down-sampled versions for the two highest \( t_{id} \)'s as described in Section 4.3. In Figure 4.3 savings in the bit-rate are estimated by computing difference in the size of bitstreams at variable rate points against the conventional bit-rates. This clearly gives us an insight that the rate required to code B-pictures is relatively much higher in the case of dynamic textures (in standard video sequences Figure B.2 the rate required is much less). In the past, down-sampling
4.2 Dynamic Texture and Coding Challenges

Figure 4.3 Estimated difference in the bit-rate computed on the Homtex database after down-sampling the two highest t_id’s by a factor of 2 along both height and width against the conventional coding.

operations were performed on the residual of P-picture in order to save bits when complex scenes appeared and up-sampling was performed at the decoder side [H263]. The method is described in brief in the next subsection, it will throw a greater insight in understanding our proposed work.

4.2.1 Reduced Resolution Update

The annex Q of the ITU-T Recommendation H.263, video coding for low bit-rate communication [H263] described an optional Reduced-Resolution Update (RRU) mode. The mode could be used when encoding a highly active scene and provided the opportunity to reduce the coded picture rate at the same time maintaining sufficient subjective quality. This is achieved by encoding an image at essentially a reduced resolution (RR), the reference picture used for prediction was of normal resolution while transform and quantization of the residual is performed at a lower resolution. At the decoder, after inverse quantization and transform, the residual needs to be upsampled prior to adding it to the motion compensated image, essentially allowing the final image to be reconstructed at full resolution as shown in Figure 4.4. This mode allowed an encoder to maintain high frame rate (and thus improved temporal resolution) while also maintaining high resolution and decent quality in stationary areas. Syntax of a bitstream encoded in this mode was essentially identical to a bitstream coded in full resolution, the main difference was on how all modes within the bitstream were interpreted, and how the residual information was considered and added after motion compensation. More specifically, an image in this mode had 1/4 the number of macroblocks (MBs) compared to a full resolution coded picture, while motion vector data was associated with block sizes of 32 × 32 and 16 × 16 of the full resolution picture, instead of 16 × 16 and 8 × 8, respectively. On the other hand, DCT and texture data are associated with 8×8 blocks of a reduced resolution image, while an upsampling process is required in order to generate the final full image representation. For complete details, please refer to [H263]. Further, this mode was proposed for H.264 [TB05; PE02].
4.3 Reduced Resolution Coding for B-Pictures

In this section, we introduce our proposed scheme used within the state-of-the-art reference software package, provided by the Joint Video Exploration Team (JVET) [JEM16]. Our scheme codes low-pass (LP) signal obtained after steerable pyramid (SP) decomposition [SF95] i.e a down-sampled picture scaled by a factor of 2 along both height and width. At the encoder side our proposed scheme employs coding the LP for B-pictures associated with temporal identifier ($t_{id}$)$>2$, instead of sending full resolution pictures as done in joint exploration model (JEM). It is to be noted that in the RRU method as described in the previous section the residual of P-picture is down-sampled, whereas in our model the B-picture’s are down-sampled. At the decoder side, up-sampling is performed in these pictures using an inverse SP operation to reconstruct fine details or high frequencies for these low resolution pictures using motion compensated band-pass (BP) and high-pass (HP) signal from its nearest key-picture. In this thesis, pictures associated with $t_{id}$<3 are termed as key-pictures. Perceptual artefacts resulting from the proposed up-sampling are identified and corrected locally by synthesizing BP and HP signals. Further, in the annex B subjective evaluation of the test model is shown to prove its significance in the state-of-the-art video coding technology.

4.3.1 Coding of Low-pass Signal

In a typical video compression configuration of HEVC [Wie14] the random access prediction structure employs hierarchical B-picture following the coding order as indicated in Figure 4.5. The coding order specifies the order in which pictures are reconstructed at the decoder and thereby defines which pictures may be used as a reference. Set of pictures comprising the coding structure is termed as group of pictures (GOP). Each picture within a GOP
4.3 Reduced Resolution Coding for B-Pictures

Figure 4.5 Hierarchical-B coding structure with 5 temporal layers. Key-pictures are indicated in dark grey shade. Light grey shade indicates pictures with \( t_{id} > 2 \) which are coded in low resolution. POC below each picture shows the display order of each picture.

has a \( t_{id} \) associated to it. There is no dependency of pictures with lower \( t_{id} \) on picture with higher \( t_{id} \) and therefore, it gives us the freedom to manipulate the pictures with higher \( t_{id} \) without affecting the key-pictures. In our experiment, GOP size of 16 with 5 temporal layers is used.

As briefly discussed in Section 4.2 that depending upon the nature of content e.g. dynamic texture, B-pictures even with higher \( t_{id} \)’s may get relatively expensive to code. Employing RR scheme can significantly benefit the coding of such content, provided the perceptual quality is kept equivalent at the decoder. In our work, an SP decomposition based RR coding scheme is employed for B-pictures with \( t_{id} > 2 \). RR refers to the SP decomposition LP signal which is coded, instead of its full resolution picture, see Figure 4.6b. SP’s perform down-sampling operation in the FFT domain and therefore normalization is performed each time to normalize the LP values in time domain to the 10 bit range. Any values after normalization less than 0 or greater than 1023 are clipped. In our case normalization factor is 4 as the original picture is scaled down by a factor of 4 i.e 1/2 height and 1/2 width. In general, normalization factor is the down-sampling factor of the picture data.

Our model is only proposed for the random access configuration and is implemented in two main stages. In the first stage, after encoding the key-pictures, the LP image of each key-picture’s reconstruction signal is stored in the picture buffer. In the next stage this signal is used for creating additional reference picture list when predicting the LP of the pictures with \( t_{id} > 2 \). It is to noted that LP images of the key-pictures are not coded into the bitstream and therefore there is no RR intra-picture. It is also to be noted that in our implementation we have not made any changes for temporal motion vector prediction.

4.3.2 Reconstruction of the RR picture at the Decoder

To decode the above bitstream, certain modifications are needed at the decoder. Anchors are conventionally decoded using the standard reference picture list as shown in Figure 4.7. We create an additional reference picture list RR with LP of each reconstructed key-picture as shown in Figure 4.8. We switch between the standard and RR reference picture list based
4 Reduced Resolution Coding with Detail Reconstruction in B-Pictures

Figure 4.6 Low-pass signal of steerable pyramid down-sampled by a factor of 2 both height and width of the original, to be coded for pictures with $t_{id} > 2$

on the $t_{id}$ of the picture order count (POC). If the $t_{id} > 2$ RR reference picture list is called for decoding.

![Decoded reference picture list in full resolution.](image)

Figure 4.7 Decoded reference picture list in full resolution.

Post-processing is applied on the RR picture after its reconstruction at the decoder. The post-processed picture for $t_{id} = 3$ i.e. e.g. POC2 cannot be placed in the RR reference picture list. This is due to the fact that picture with $t_{id} = 3$ may act as a reference for decoding a picture with $t_{id} = 4$ i.e. e.g. POC1 and POC2. Therefore, an additional buffer is created for holding the post-processed $t_{id} = 3$ picture until its adjacent POCs with $t_{id} = 4$ are fully decoded (see Figure 4.9). No additional buffers are needed for holding post-processed pictures with POCs $t_{id} = 4$, due to the fact that the pictures in this temporal layer are not used as a reference for any other layer.

![RR reference picture list derived from the standard full resolution reference pictures](image)

Figure 4.8 RR reference picture list derived from the standard full resolution reference pictures
4.4 Post-processing for the Reduced Resolution Picture

Post-processing is initiated right after the reconstruction of the RR picture at the decoder. The main goal of the post-processing algorithm is to synthesize high frequencies i.e. up-sample the picture with fine details that were discarded during the SP down-sampling operation at the encoder. In order to reconstruct these details, our algorithm targets to synthesize the BP and HP component for the decoded LP picture. Since SP decomposition is invertible i.e. it can reconstruct back the original signal given its decomposed components a decent reconstruction quality is achievable in full resolution, provided the signal in the bands is synthesized to a decent level of accuracy.

Consecutive pictures generally have high similarity in a video and therefore, the probability of finding a close approximation of both BP and HP signal is maximum in the corresponding BP and HP of the nearest key-picture. In order to search the closely matching signal for RR picture sub-bands e.g. POC 1, it can be best predicted from POC 0 similarly, for POC 3 it can be best predicted from POC 4. For POC 2 we use the causal signal in display order i.e. POC 0 as it lies centered between two key-pictures as shown in Figure 4.5.

4.4.1 Motion Estimation using Low-pass Picture

ME is done between LP of the key-picture with its nearest RR coded picture. Separate ME is done for the luma and chroma. These pictures are readily available at the decoder side after the decoding. Dynamic range of the LP of a key-picture is different from the dynamic range of a RR coded picture. Therefore the RR coded picture’s range is multiplied with the normalization factor in order to bring the LP into same value range. Bi-directional block
matching operation is performed between the closest key-picture’s LP block \( f(x, y) \) and RR picture block \( g(x, y) \). Minimum sum of the absolute difference (SAD) is used to find the MV’s. This gives us an integer pixel displacement \((\Delta x, \Delta y)\). If \((\Delta x, \Delta y)\) is the true displacement, then \((\Delta x, \Delta y)\) determined using block matching algorithm should be a good integer estimate of \((\Delta x, \Delta y)\). The block size i.e \((4 \times 4)\) is kept uniform throughout the entire process for luma component and \((2 \times 2)\) for the chroma components. For finding the best match in the RR picture we have used log search which is one of the fast search algorithm with search range of 40 for \( t_{id}=3 \) and 32 for \( t_{id}=4 \). Varying such range is due to the fact that \( t_{id}=3 \) is farther away from its nearest key-picture and therefore, larger search range may be required for large motions.

The second step of the algorithm is to use Taylor series approximation to refine the search \([\text{CVN10}]\). Since the shifted block \( f(x + \delta x, y + \delta y) \) differs from the true image by only \((\delta x, \delta y)\), where \(|\delta x| < 1 \) and \(|\delta y| < 1 \) the Taylor series approximation is approximately valid. Therefore, the overall displacement can be determined as

\[
\Delta x = \overline{\Delta x} + \delta x \quad \text{and} \quad \Delta y = \overline{\Delta y} + \delta y
\]  

(4.1)

It is observed that some of the MVs may be false due to complex motions. These incorrect MVs normally have low spatial coherence in the MV fields. Therefore a motion vector smoothing is applied to the entire MV field in order to suppress random MVs by using 2D median filtering with window size \(3 \times 3\) see \[\text{Figure 4.12}\]. Before MC MVs are scaled by the scaling factor i.e. 2, the resolution of the sub-band to be compensated is 2 times the height and width of the picture on which the motion is estimated. Further, the smoothed MV field is up-sampled to the size of the picture to be compensated i.e. each pixel has its own MVs during compensation, such operations tends to minimize blocking artefacts \([\text{TL11}]\) as shown in \[\text{Figure 4.12c}\].

\[\text{Figure 4.11} \quad \text{ME using a combination of block matching algorithm and Taylor Approximation.}\]

\[\text{4.4.2 Motion Estimation using magnitude of the orientated BP}\]

Besides using LP for ME, scale-2 BP is also investigated for ME. In general, for a multiple orientation SP sub-band decomposition, we perform a separate ME at scale 2 between each orientation (magnitude) of the key-picture’s LP with its corresponding orientation from the RR coded picture as shown in \[\text{Figure 4.13}\]. The result is an over-complete MV field by a factor of \( K \), where \( K \) is number of orientations. The MV pair for a specific block location corresponding to the minimum SAD in all the orientations is selected for compensating all
4.4 Post-processing for the Reduced Resolution Picture

Figure 4.12 MV refinement and its effect on the compensated signal. It is clear that block based shifting produces blocking artefacts where as pixel based shifting minimizes such block boundary discontinuities. It is also observed that MV smoothing is must to preserve the important structures from distorting.

The oriented sub-bands. In some cases, blocks in a specific orientation may not have presence of any structures and this may lead to wrong selection of MV due to very low SAD score. Therefore, MVs corresponding to only those blocks whose energy is more than a minimum threshold $E$ can be selected for compensating. We have empirically defined the value of $E=1000$ which is kept constant for all sequences.

$$MV_{minSAD} = \min_{SAD} \begin{cases} MV_1 \\ MV_2 \\ MV_k \end{cases}$$

(4.2)

All further steps in motion refinement are the same as mentioned in Section 4.4.1. It is a well known fact that the ME is most accurate along the direction of maximum image gradient, at edges this direction lies perpendicular to an edge. Isolation of edge structures using SP decomposition i.e separating e.g. horizontal and vertical component of a structure into separate sub-bands and estimating motion using these isolated bands provides a relatively higher probability of finding a better candidate match in one of the orientations that leads to a better MC for all the orientations provided, the block size is kept small like $4\times4$ in our method. In the case when motion is estimated from the LP image, such isolation of oriented structures is not possible and therefore the estimate may not reflect the isolated motion for a specific edge inside a block or in case of complex textures may have a completely wrong match. This isolation of the oriented structures tend to perform better over the textured regions consisting of multiple oriented structures.
Figure 4.13 Block matching is performed between corresponding orientations only. This gives us an over-complete MV field. The MV pair with minimum SAD among ME1 and ME2 is selected for MC of all orientations.

Figure 4.14 MC performance is limited over spatial high frequency content when motion is estimated using the LP given in Section 4.4.1 as shown in the left image. MC performs better over spatial high frequency content when estimated using method proposed in Section 4.4.2 as shown in the right image.

4.4.3 Motion Estimation using Phase of the oriented BP

A comparison of our proposed method in Section 4.4.2 is made with a similar model based on phase of the oriented BP for ME. Amplitude of the calculated phase can take any value in the range $[0, 2\pi]$ that is returned by the arctangent function. In cases, where the phase exceeds this range of values, it will be wrapped so that it stays within the normal range $[0, 2\pi]$. In such cases, the wrapped phase will contain one, or more $2\pi$ jumps. Therefore, during block matching check for the phase shift is introduced that only selects the smallest phase shift as the true shift between the two corresponding coefficients inside blocks being matched see Equation 4.3, $\theta_1$ and $\theta_2$ corresponds to the phase angles of the coefficients inside the blocks being matched and $\theta_1 > \theta_2$.

$$\theta_{diff} = \min \left\{ \frac{\theta_1 - \theta_2}{\theta_2 + 2\pi - \theta_1}, \theta_2 + 2\pi - \theta_1 \right\} \quad \text{where} \quad \theta_1 > \theta_2$$

(4.3)
4.4 Post-processing for the Reduced Resolution Picture

The $\theta_{diff}$ provides us the difference for SAD which is the matching criterion. Remaining settings are similar to those mentioned in Section 4.4.2. The result of the phase based method is compared against the proposed method as shown in Figure 4.15. It can be clearly seen that phase based ME is not robust in textured areas.

Figure 4.15 MC performance is limited over spatial high frequency content when motion is estimated using the phase of the Band-pass given in Section 4.4.3 as shown in the left image. MC performs better over spatial high frequency content when estimated using method proposed in Section 4.4.2 as shown in the right image.

4.4.4 Spatial Domain Upsampling of RR Coded Pictures

Up-sampling with conventional methods like nearest neighbour, bilinear and Lanczos [Pad14] makes the picture look blurred when compared to our method proposed in Section 4.4.2 as shown in Figure 4.16. Also, in general the performance of these up-sampling methods is very limited especially in video coding due to quantization of the residual that introduces distortions in the reconstructed down-sampled picture itself. This type of reconstructed signal is not appropriate for a decent quality up-sampling. Pumping of the reconstruction quality is likely to be induced over time when upsampled with these methods. The reason for pumping is due to the fact that non-key-pictures are at much lower perceptual quality than the key-pictures. Therefore, it is necessary to synthesize the band-pass signal for reconstruction that tunes well in sharpness with the key-pictures.

4.4.5 Error Correction and Hole-filling

Due to the motion of an object or camera over time, regions will be occluded or new areas will be exposed (holes) in the RR coded picture, which happens mostly at the object boundaries. Exposed areas or holes refer to the new areas that were actually occluded in the key-picture and become visible in the RR coded picture, due to local motion of the objects or camera panning. During block matching, if the block location corresponds to a hole in the RR coded picture, any compensated block at this location is not an appropriate match and may result
Figure 4.16 Subjective comparison, low-pass coded at QP=27 and upsampled. (a) shows the reference (c) and (d) shows the upsampled low-pass using conventional methods. (b) shows the upsampled signal using method proposed in Section 4.4.2.
in an artefact. Artefacts also appear when compensating challenging areas consisting of spatially high frequencies [Gir93], due to limited performance of ME over such regions.

In this subsection, we propose the identification of such error blocks. During MC of BP we also compensate the up-sampled LP of the key-picture. This compensated signal has a reference which is up-sampled LP of the RR coded picture. The reason to use this is that errors are robustly measured in the pixel domain on lower frequencies which are perceptually sensitive and it gives us a decent estimate about the errors in higher frequencies. Individual block level PSNR values gives us an estimate on the local quality of the region. A threshold (δ) peak signal-to-noise ratio (PSNR) is used to decide if the block needs any correction. Any compensated block in the RR coded up-sampled LP whose PSNR is below (δ) is used as an error indicator and no compensation is done in the oriented BP corresponding to these blocks. The identified blocks below (δ) are not compensated and a zero value is filled into the erroneous locations. Our proposed model has a significant advantage, as the granularity in motion comes from the RR coded picture and the compensated BP only complements to its sharpness. Therefore, instead of viewing visually annoying artefact at the locations of holes or complex motions due to MC, we simply discard the high frequency band over that region. This gives an overall decent picture reconstruction without any perceptual distortions when viewing the content, see Figure 4.17.

4.5 Grand Challenge Model

A Grand Challenge (GC) model is derived from the scheme described in Section 4.3 for participation in a global event on video compression technology held in ICIP, Beijing [GC]. The goal of the challenge is to identify technology that improves compression beyond the current state-of-the-art in video compression, the most recent standard HEVC [Sul+12]. Participants were asked to deliver bitstreams with pre-defined maximum target rates for the provided set of sequences, and a decoder executable for reconstructing
the decoded videos. A subjective test evaluation was carried out to judge the performance of our model compared to the anchor bitstreams.

In this sub-section we describe the necessary changes made during the post-processing stage in our model. The model was in an intermediate stage of development during the competition time and therefore, introduced more artefacts compared to the one proposed in Section 4.4.2.

### 4.5.1 Motion Estimation using Band-pass Coefficients

Performing IFFT operation on the FFT of the pyramid’s LP gives us the LP in time domain. In our observation it is found that the LP image has an inherent global shift introduced as a result of frequency domain down-sampling. This shifted signal when used for ME may introduce artefacts such as double contouring on edges due to the shift also being carried into the high resolution compensated sub-bands. The shift becomes even more prominent due to scaling and upsampling of MV field by a factor of 2 both in height and width as the sub-band to be compensated is 4 times the LP resolution. To fix this shift a reconstruction using inverse SP is done with BP and HP valued to zeros. This reconstructs an up-sampled LP picture in time domain with full resolution. The problem of the shifted LP is fixed in this up-sampled LP which is further down-sampled by a factor of 2 along height and width in the time domain using bilinear interpolation to speed up the ME process (see Figure 4.18).

In the GC model, separate ME is performed on real and imaginary part of the BP coefficients. To reduce the computation complexity due to ME only single sub-band is used for up-sampling the RR coded picture. Although, the reconstruction with only one sub-band is not perfect, however due to the low rate points of the anchor bitstreams it was empirically observed that the quality was comparable perceptually. Finally, there is a significant change in the error correction and hole-filling section as compared to one explained in Section 4.4.5.

### 4.5.2 Error Correction

The BP and HP approximation using ME for \( t_{\text{id}} > 2 \) pictures show ringing artefact after reconstruction. Such artefacts are noticeable perceptually, if the video is viewed in slow motion. One underlying reason for such distortions is limited performance of MC in certain regions e.g. uncovered regions (holes), regions with high spatial frequencies, edges separating bright and dark regions etc. In our observation it was observed that pictures with \( t_{\text{id}}=3 \) perceived relatively better after MC than \( t_{\text{id}}=4 \), due to the reason that they are adjacent to the nearest key-picture. In order to have an acceptable perceptual quality after reconstruction, low quality compensated motion blocks are identified and corrected. Figure 4.19 shows the improvement in reconstruction quality after block level error correction. In our method PSNR is used as a distortion measure to validate the quality of compensated signal. It is applied on the block level to identify the perceptual distortions as described in Section 4.4.5.
Figure 4.19 The figure shows the cropped section from the sequence LampLeaves where a comparison is made with- and without error correction. On the left side result is shown without any error correction. On the right side the result is shown after error correction.

The selection of $\delta$ for discarding the relevant MVs related to large distortions is empirically selected based on the trials conducted on a range of $\delta$ values and is kept constant for all the sequences at all rate points. The BP and HP signal for these identified blocks are replaced with the aliased component which is present in the RR coded picture (due to the nature of LP filter shown in Figure 2.6c) since it is also part of the band-pass signal. The corrected BP and HP are finally used to reconstruct the upsampled picture.

Subjective and objective evaluations for the GC model are provided in Appendix B.
5 Conclusion

In this thesis separate investigations are provided for perceptual coding of static and dynamic textures but not limited to, by using an over-complete steerable pyramid representation. The over-complete and complex output of the steerable pyramid decomposition provides measure of local phase and energy as visual descriptors for static textures. Such visual descriptors provide a compact representation for static textures as compared to conventional intra coding of HEVC, provided the area of the homogeneous texture is sufficiently large. The main contribution of this thesis with regard to the static textures lies in the design of compression schemes for efficient coding of the synthesis parameters (descriptors). A dedicated compression scheme ( steps involving transform, prediction, quantization and binarization) is designed to efficiently code each parameter type. Effect of parameter quantization on the visual quality of synthesized texture is also part of the investigation. Finally, pairwise subjective comparisons are provided to judge the performance of the texture synthesis coding showing clear benefit against the HEVC intra coded texture at similar rate for a fixed patch size.

When integrating such a scheme with the rate distortion mechanism of a codec an open question arises on how to evaluate the quality of synthesized texture. State-of-the-art quality assessment tools like PSNR, SSIM, VIF etc. are unable to effectively evaluate the quality of synthesized texture. Beside quality assessment, texture synthesis is computationally expensive process for the decoder side as it is iterative (25 iteration are used in our case) with only a decent visual quality achievable. In order to improve the existing model further studies can be carried out on analysing which parameters captures most of the visual quality of textures in addition to this what is the right number of scales and orientations needed for decent texture analysis and synthesis. These investigation will be highly content specific but will have a tremendous impact on the compression of parameters. This will lead to comparable synthesis quality for smaller blocks and reduced computational complexity too at the decoder. However, synthesized textures blocks will require blending with neighbouring blocks so as to give the texture a homogeneous appearance. As already mentioned in the Section 3.6.3 the synthesis algorithm used here provides decent visual quality results only for a small range of content and this may restrict the use of such a technology in a practical video coding scenario beyond research purposes as the potential room for the coding gains may be low. In such cases the synthesis approach that can be trained for wider range of textures is crucial and, approaches based on neural networks may be a better option [GEB15].

In the context of dynamic textures, our investigation reveals that B-pictures ( \( t_{id} > 2 \)) for dynamic textures are expensive to code and therefore a reduced resolution scheme with decoder side high-frequency synthesis is proposed as a decent solution. In this method at the encoder side B-pictures corresponding to the two highest \( t_{id}'s \) are coded in lower resolution. At the decoder side, the low resolution pictures are decoded which is followed by their up-sampling using steerable pyramids (as a post-processing step). The up-sampling method reconstructs high frequencies using motion compensated band-pass signal from the
5 Conclusion

The performance of the proposed method is evaluated subjectively as well as objectively against the state-of-the-art JEM reference software, and it is shown to provide subjective quality improvements at the same rate (Section B.2.5).

The thesis provides an efficient way of coding dynamic texture content but not limited to. Compensating high-frequencies from neighboring key-pictures provides a high-quality reconstruction in these pictures and thus improves the overall perceptual quality of the synthesized video. One huge downside of this model is that it is computationally expensive due to decoder side motion estimation involved. In a real world scenario only a few blocks may correspond to the dynamic texture that are expensive to code. In such cases using steerable pyramid as a global tool for down-sampling and post-processing is not an efficient method. Future work will focus towards adapting the synthesis model to a block level synthesis tool i.e. a locally adaptive spatial resolution scheme. In this scheme the blocks that correspond to dynamic textures will be coded in down-sampled format and up-sampled at the decoder side. Such adaptive spatial resolution scheme requires design of appropriate filters with finite impulse response that can be made compatible with the small block sizes as used in HEVC and JEM softwares. Further, common problem with both the static and dynamic texture is the quality evaluation of the synthesis. As already discussed that, state-of-the-art quality assessment tools are unable to properly evaluate the synthesized video quality and without a proper quality metrics it is not practically feasible to use the developed coding tools in a realistic video coding scenario.
A Annex: Additional Results on Static Texture Synthesis
Figure A.1 A subjective comparison is shown at a similar bits/pixel (bpp) for the grass type-2 texture patch (256×256 pixels).
Figure A.2 A subjective comparison is shown at a similar bits/pixel (bpp) for the carpet type-1 texture patch (256×256 pixels).
Figure A.3 A subjective comparison is shown at a similar bits/pixel (bpp) for the carpet type-2 texture patch (256×256 pixels).
Figure A.4 A subjective comparison is shown at a similar bits/pixel (bpp) for the carpet type-3 texture patch (256×256 pixels).
Figure A.5 A subjective comparison is shown at a similar bits/pixel (bpp) for the carpet type-3 texture patch (256×256 pixels).
Figure A.6 A subjective comparison is shown at a similar bits/pixel (bpp) for the sand texture patch (256×256 pixels).
A subjective comparison is shown at a similar bits/pixel (bpp) for the texture patch (256×256 pixels).

Figure A.7
Figure A.8 A subjective comparison is shown at a similar bits/pixel (bpp) for the pebbles texture patch (256×256 pixels).
A subjective comparison is shown at a similar bits/pixel (bpp) for the denim texture patch (256×256 pixels).

**Figure A.9**
Figure A.10 A subjective comparison is shown at a similar bits/pixel (bpp) for the carpet type texture patch (256×256 pixels).
A subjective comparison is shown at a similar bits/pixel (bpp) for the dry grass texture patch (256×256 pixels).
Figure A.12 A subjective comparison is shown at a similar bits/pixel (bpp) for the grass type-1 texture patch (128×128 pixels). An adaptive quantization is applied separately on the ACF ie 0.08, 0.15 and for the cross-correlation 0.6 in order to match the rates.
Figure A.13 A subjective comparison is shown at a similar bits/pixel (bpp) for the grass type-2 texture patch (128×128 pixels). An adaptive quantization is applied separately on the ACF ie 0.01, 0.07 and for the cross-correlation 0.6 in order to match the rates.
Figure A.14 A subjective comparison is shown at a similar bits/pixel (bpp) for the dry grass texture patch (128×128 pixels). An adaptive quantization is applied separately on the ACF ie 0.013 and for the cross-correlation 0.6 in order to match the rates.

Figure A.15 A subjective comparison is shown at a similar bits/pixel (bpp) for the grass type-2 texture patch (64×64 pixels). An adaptive quantization is applied separately on the ACF ie 0.63 and for the cross-correlation 0.6 in order to match the rates.
Figure A.16 A subjective comparison is shown at a similar bits/pixel (bpp) for the dry grass texture patch (64×64 pixels). An adaptive quantization is applied separately on the ACF ie 0.63 and for the cross-correlation 0.6 in order to match the rates.
B Annex: Results of the RR coding

B.1 Experimental Set-up and Results

In this chapter, a brief description is given on the experimental set-up which is followed by subjective testing methodology used for evaluation of the proposed model as described in Chapter 4. All the testing procedures were handled independently by the Grand Challenge Organizers [GC]. The primary goal of the challenge was to identify technology that improves compression beyond the current state-of-the-art. The author chose the competition as a platform for subjective evaluation of his proposed model against the JEM anchors. Participants were asked to deliver bitstreams with pre-defined maximum target rates Table B.2 and a decoder executable for reconstructing the decoded videos. A subjective as well as objective evaluation was carried out to judge the performance of the proposed GC model. Our proposed model was developed on JEM 6.0 [JEM16] reference software base which was the state-of-the-art at the time of competition.

B.2 Experimental Set-up

The Grand Challenge Organizing Committee provided the test sequences and certain rules to be followed that are explained in the next subsection.

B.2.1 Encoder configuration

Random Access configuration was used for evaluation, following the JVET common test conditions and software reference configurations [SL16]. The intra refresh period should be dependent on the frame rate of the source and the GOP size in use: a value of 48 should be used for 50fps, a value of 64 should be used for sequences with 60fps. A static QP setting was applied for generation of the anchors. A one-time change of the quantization parameter from value QP to value QP+1 may be applied in order to meet the defined target bit rates. The quantization parameter settings applied for the anchors was to be reported. If specific mechanisms of rate control, and multi-pass/lookahead decisions are used during

Figure B.1 Dynamic texture test sequences from the BVI database [Pap+15]
B Annex: Results of the RR coding

![Image](144x713 to 253x782)
![Image](257x713 to 366x782)
![Image](369x713 to 479x782)
![Image](200x624 to 310x693)
![Image](313x624 to 422x693)

(a) CatRobot  (b) DaylighRoad  (c) FoodmarketPopcorn
(d) FoodmarketSteam  (e) ParkRunning2

Figure B.2 Standard test sequences

B.2.2 Content Specifications and Rate Points

The sequences consisted of both dynamic texture and non-dynamic texture content. In total 9 sequences were provided of which 4 consisted of dynamic textures (2 continuous i.e. CalmingWater and DropsOnWater and 2 discrete type i.e. LampLeaves and TreeWills as shown in [Figure B.1]). Remaining 5 were cropped UHD (2500×1600) video sequences that consisted of standard content i.e. non-dynamic textures [Wie+17]. Temporally, the sequences have an extent of 5 seconds and in some cases 10 seconds. All the specification about the content are given in Table B.1. 4 separate target rate points were provided for each of the 9 sequences see Table B.2. The performance of the proposed technology was evaluated against the anchors and other participants at these rate points.

<table>
<thead>
<tr>
<th>Sequence Name</th>
<th>Resolution</th>
<th>Frame Count</th>
<th>Frame Rate</th>
<th>Duration</th>
<th>Chroma Format</th>
<th>Bit depth</th>
</tr>
</thead>
<tbody>
<tr>
<td>CalmingWater</td>
<td>1920×1080</td>
<td>300</td>
<td>60</td>
<td>5</td>
<td>4:2:0</td>
<td>10bit</td>
</tr>
<tr>
<td>DropsOnWater</td>
<td>1920×1080</td>
<td>300</td>
<td>60</td>
<td>5</td>
<td>4:2:0</td>
<td>10bit</td>
</tr>
<tr>
<td>LampLeaves</td>
<td>1920×1080</td>
<td>300</td>
<td>60</td>
<td>5</td>
<td>4:2:0</td>
<td>10bit</td>
</tr>
<tr>
<td>TreeWills</td>
<td>1920×1080</td>
<td>600</td>
<td>60</td>
<td>10</td>
<td>4:2:0</td>
<td>10bit</td>
</tr>
<tr>
<td>CatRobot</td>
<td>2560×1600</td>
<td>600</td>
<td>60</td>
<td>10</td>
<td>4:2:0</td>
<td>10bit</td>
</tr>
<tr>
<td>DaylightRoad</td>
<td>2560×1600</td>
<td>300</td>
<td>60</td>
<td>5</td>
<td>4:2:0</td>
<td>10bit</td>
</tr>
<tr>
<td>FoodmarketPopcorn</td>
<td>2560×1600</td>
<td>300</td>
<td>60</td>
<td>5</td>
<td>4:2:0</td>
<td>10bit</td>
</tr>
<tr>
<td>FoodmarketSteam</td>
<td>2560×1600</td>
<td>300</td>
<td>60</td>
<td>5</td>
<td>4:2:0</td>
<td>10bit</td>
</tr>
<tr>
<td>ParkRunning2</td>
<td>2560×1600</td>
<td>250</td>
<td>50</td>
<td>5</td>
<td>4:2:0</td>
<td>10bit</td>
</tr>
</tbody>
</table>

Table B.1 Table showing specification of test sequences.
B.2 Experimental Set-up

<table>
<thead>
<tr>
<th>Sequence Name</th>
<th>Rate1</th>
<th>Rate2</th>
<th>Rate3</th>
<th>Rate4</th>
</tr>
</thead>
<tbody>
<tr>
<td>CalmingWater</td>
<td>1500</td>
<td>1800</td>
<td>3000</td>
<td>4300</td>
</tr>
<tr>
<td>DropsOnWater</td>
<td>2800</td>
<td>2000</td>
<td>4800</td>
<td>8000</td>
</tr>
<tr>
<td>LampLeaves</td>
<td>1500</td>
<td>800</td>
<td>3300</td>
<td>7500</td>
</tr>
<tr>
<td>TreeWills</td>
<td>350</td>
<td>550</td>
<td>1100</td>
<td>1600</td>
</tr>
<tr>
<td>CatRobot</td>
<td>450</td>
<td>1800</td>
<td>800</td>
<td>1400</td>
</tr>
<tr>
<td>DaylightRoad</td>
<td>450</td>
<td>750</td>
<td>1100</td>
<td>1600</td>
</tr>
<tr>
<td>FoodmarketPopcorn</td>
<td>500</td>
<td>750</td>
<td>1200</td>
<td>1900</td>
</tr>
<tr>
<td>FoodmarketSteam</td>
<td>470</td>
<td>650</td>
<td>1200</td>
<td>1700</td>
</tr>
<tr>
<td>ParkRunning2</td>
<td>800</td>
<td>1400</td>
<td>2200</td>
<td>4300</td>
</tr>
</tbody>
</table>

Table B.2 Table showing maximum bit-rate in kbit/s for a specific quality range.

B.2.3 Subjective Test Conditions

The testing conditions as per ITU Recommendation BT.500-13 document on methodology for subjective assessment of the quality of television pictures [13] were followed, which included laboratory environment specification and fixed viewing distance of 3×Height of the display. Details on testing equipment are listed in Table B.3.

<table>
<thead>
<tr>
<th>Test Conditions</th>
<th>University of Nantes</th>
<th>University of Bristol</th>
</tr>
</thead>
<tbody>
<tr>
<td>TV</td>
<td>LG OLED65W7V Size: 65” OLED 4K@60fps 10-bit panel Calibrated with i1 system (Xrite)</td>
<td>SONY KD-ZD9 Size: 65” OLED 4K@60fps 10-bit panel</td>
</tr>
<tr>
<td>PC</td>
<td>Core i7-4930k, RAM: 64GB, Windows 7, NVIDIA GTX980</td>
<td>Intel i7-4770K 4 <a href="mailto:cores@3.5GHz">cores@3.5GHz</a>, RAM: 64GB, Windows 7, NVIDIA GeForce GTX 1080</td>
</tr>
<tr>
<td>Player</td>
<td>MPC-HC (sequences in .nut containers using Ffmpeg)</td>
<td>Psychotoolbox with GStreamer</td>
</tr>
<tr>
<td>Voting Interface</td>
<td>Implemented in Matlab Selection of rating using the mouse.</td>
<td>Online Form including reference scale accessed on smartphones.</td>
</tr>
</tbody>
</table>

Table B.3 Testing Conditions

B.2.4 Subjective Test Methodology

The subjective testing comprised of two main phases, 1.) Pilot study and 2.) Main study. These studies are comprehensively explained later in the sub-sections. Pilot testing is run with participants outside the research team who are unfamiliar with the research goals. Pilot study is treated as the main study in order to collect appropriate data [16b]. Pilots are
B Annex: Results of the RR coding

important to capture any issues with study length, number of tasks performed, treatments order, randomization and study breaks. After revising the experimental set-up from pilot study, the main phase is started.

All the evaluation were carried out with similar experimental settings in two separate laboratories one in the University of Bristol (UK) and the other in the University of Nantes (France). Testing material was kept completely anonymous from the labs conducting the tests. The method named absolute category rating with hidden reference (ACR-HR) is followed for the subjective quality assessment which is one of the standard methods mentioned in the ITU-T recommendation P.910 [08]. Prior to a session, the observers were screened for (corrected-to-) normal visual acuity on the Snellen chart, and for normal colour vision using Ishihara chart.

Phase 1: Pilot Study

There were 9 source video sequences (SRCs), a subset of 99 hypothetical references (HRCs) was created from the submissions of all the participants (6) (only 3 were used for pilot), JEM anchors(1), HM anchors(1) and originals(1) for the testing phase 1 pilot study. Out of the 4 rate points for each SRC, only two rate points were selected for pilot (see Table B.4). In each case study the rate points were kept different in order to achieve a uniform distribution of qualities and also taking into account the Video Multimethod Assessment Fusion (VMAF) results [16c]. A scale of 5 grades i.e Bad→Poor→Fair→Good→Excellent is used for voting. Approximate duration of the test is 26 minutes ((99 sequences + 4 sequences for training) ×15sec (playing video and voting)). A total of 39 participants voted.

<table>
<thead>
<tr>
<th>Rate Points used for the Pilot</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequence Name</td>
</tr>
<tr>
<td>CalmingWater</td>
</tr>
<tr>
<td>DropsOnWater</td>
</tr>
<tr>
<td>LampLeaves</td>
</tr>
<tr>
<td>TreeWills</td>
</tr>
<tr>
<td>CatRobot</td>
</tr>
<tr>
<td>DaylightRoad</td>
</tr>
<tr>
<td>FoodmarketPopcorn</td>
</tr>
<tr>
<td>FoodmarketSteam</td>
</tr>
<tr>
<td>ParkRunning2</td>
</tr>
</tbody>
</table>

Table B.4 Sequences and Rate Points (1,2,3 or 4) used for the Pilot.

Phase 2: Main Study

Exactly same methodology was also followed in the phase 2 main study. All the remaining SRCs were used in the main study. The approximate duration of the test is 60 minutes with 5
minutes break ((243 HRCs + 4 training sequences)×15 seconds). A total of 53 participants voted.

**B.2.5 Subjective Results**

The two laboratories performed the subjective testing using the methods described in the previous section. Mean opinion score (MOS) is a well known measure used in the domain of quality of experience and telecommunications engineering representing overall quality of a system. Each laboratory calculated MOS for all the SRCs (total 9) at all rate point (R1, R2, R3 and R4) using its experimental data (participants opinion of the performance of technology). The subjective scores from both the labs were comparable as no significant difference in experimental observations was noticed. Similar results from two different labs under very similar test conditions is a cross-check on performance of the proposed technologies. In this thesis we only mention the final result after the data from two separate labs was merged together. The proposed submissions were compared with the corresponding anchors at similar rate. R1 is highest rate and decreases towards R4 which is the lowest rate point for study.

**Performance on Dynamic Texture Test Sequences**

A comparison of average MOS scores for all the SRCs at all rate points is done to judge the overall quality of the technology. Bit-rate decreases from R1→R4 (complete details are mentioned in Table B.2). In the case of continuous dynamic textures MOS score are shown in Figure B.3 and Figure B.4. Based on the MOS scores it is evident that at higher rates (R1 and R2) there is no significant improvement in the case of CalmingWater and DropsOnWater when compared to the anchors. At lower rate points (R3 and R4) the performance is significantly better in the case of CalmingWater and better for DropsOnWater when compared to the anchors. The proposed performance is better at the lower rates due to the fact that strong noticeable artefacts are visible in anchors e.g. blocking, blurring etc. due to codec often switching to intra prediction with heavy quantization of residual in case of the B-pictures of continuous dynamic textures (see Chapter 4). This Rate-Distortion decision is not optimum for continuous dynamic textures and as a result motion appears to freeze in the B-pictures and thus leading to discontinuity in motion or motion may appear to be very fast. Our proposed solution uses down-sampled B-pictures with a modified QP-offset kept constant for all the dynamic textures. The same Rate-Distortion concept works better on the down-sampled B-pictures due to large rate available for less samples. Therefore relatively less blocking and blurring is noticeable when upsampled using the proposed method.

In the case of discrete dynamic textures MOS score are shown in Figure B.5 and Figure B.6. Based on the MOS scores it is evident that the proposed technology performs better in the case of LampLeaves at lower rates and performs better at all ratepoints for TreeWills when compared to the anchors. In general, for discrete dynamic textures it is difficult to observe blocking and blurring artefacts due to the nature of the content that camouflages them under its high spatial and temporal frequency characteristic. This content has discernible appearance of motion in time and therefore artefact that is visually most susceptible to the human eye is the motion jerkiness which may occur at extremely low rates in B-pictures. Our model works well here as the granularity in motion is provided by down-sampled RR coded pictures and spatial sharpness using the compensated bandpass component.
Performance on Standard Test Sequences

In the case of standard sequences based on the MOS scores it is evident that in most cases our performance is almost similar to the anchors at all the rate points as shown in Figure B.7 - B.11. This already attests the performance of the model even beyond dynamic textures. In standard content B-pictures have relatively very low rate compared to dynamic textures. This is due to the fact that MC works very well here. Therefore our proposed method make no significant difference in most cases when compared to JEM anchors (CatRobot, DaylightRoad, Parkrunning). However, in the case of FoodMarketPopcorn our performance is overall better over JEM anchors in all cases. In the case of FoodMarketSteam the performance is significantly better at the lowest rate point due to presence of smoke (continuous dynamic texture). At lowest rate point such regions show strong blocking and blurring artefacts when compared to our proposed method.

B.2.6 Objective Results

Objective quality assessment was also performed for both the GC model and the further improved model described in Section 4.4.2. Well known quality measures PSNR, SSIM, Ms-SSIM, VIF2 and VMAF [WSB03; SB06; Ras17] were used to evaluate the quality of the proposed method against the anchors. In the case of continuous dynamic textures the proposed quality is similar to the anchors for all the rate points in all the quality metrics. Similar behaviour is also observed for the discrete dynamic textures except at larger rates the PSNR for LampLeaves is better for the anchor’s quality.

In the standard test sequences generally the quality of anchors is consistently improving with increase in the rate over the proposed results in all the quality metrics. However, in the case of FoodMarketPopcorn and FoodMarketSteam at lowest rate point the objective quality of the proposed is better over anchors. This result is strongly correlated with the subjective
B.2 Experimental Set-up

scores. Based on our observation it is challenging to match the subjective quality with the objective assessment. However, improved assessment methods like VMAF are relatively more correlated with the subjective quality for standard sequences. Improvements in such quality methods are crucial for the success of perceptual video coding.
B Annex: Results of the RR coding

Figure B.5 LampLeaves

Figure B.6 Treewills
B.2 Experimental Set-up

Figure B.7 CatRobot

Figure B.8 DaylightRoad
B Annex: Results of the RR coding

Figure B.9 FoodMarketPopcorn

Figure B.10 FoodMarketSteam
B.2 Experimental Set-up

Figure B.11 ParkRunning
Figure B.12 CalmingWater
B.2 Experimental Set-up

Figure B.13 DropsOnWater
B Annex: Results of the RR coding

Figure B.14 LampLeaves
B.2 Experimental Set-up

Figure B.15 TreeWills
B Annex: Results of the RR coding

![Figure B.16](CatRobot)
B.2 Experimental Set-up

Figure B.17 DaylightRoad
B Annex: Results of the RR coding

Figure B.18 FoodmarketPopcorn
B.2 Experimental Set-up

---

Figure B.19 FoodmarketSteam
Figure B.20 ParkRunning2
Bibliography


Bibliography


Bibliography


Bibliography


Bibliography

