Modeling the Perceptual Similarity of Static and Dynamic Visual Textures -Application to the Perceptual Optimization of Video Compression

Karam Naser

UNIVERSITE BRETAGNE LOIRE

UNIVERSITÉ DE NANTES

Thèse de Doctorat

Karam NASER

Mémoire présenté en vue de l'obtention du
grade de Docteur de l'Université de Nantes
Label européen
sous le sceau de l'Université Bretagne Loire

École doctorale : Sciences et technologies de l'information et de la communication

Discipline : Sciences et technologies de l'information et de la communication, section CNU 27
Unité de recherche : Laboratoire des Sciences du Numérique de Nantes (LS2N)

Soutenue le 19 Mai 2017

Modeling the Perceptual Similarity of Static and Dynamic Visual Textures - Application to the Perceptual Optimization of Video Compression

JURY

Président :	**M. Olivier DEFORGES**, Professeur des universités, Institut National des Sciences Appliquées de Rennes
Rapporteurs :	**M. Jens-Rainer OHM**, Professeur étranger, RWTH Aachen University
	M. Marco CAGNAZZO, Maître de Conférences, Telecom ParisTech
Examinateurs :	**M^{me} Marta MRAK**, PhD, Lead Engineer, BBC, Research and Development department
	M^{me} Amy REIBMAN, Professeur étranger, Purdue University
	M. Marc ANTONINI, Directeur de Recherche CNRS, Laboratoire L3S
Directeur de thèse :	**M. Patrick LE CALLET**, Professeur des universités, Université de Nantes
Co-encadrant de thèse :	**M. Vincent RICORDEL**, Maître de Conférences, Université de Nantes

To Nazar

Acknowledgment

I would like to express my sincere gratitude to my supervisor Prof. Patrick LE CALLET, for providing me the opportunity to join his research team and conduct my PhD. His deep knowledge and support was the backbone of this work. I appreciate our scientific discussions and research planning. Beside this, it was a great time working together. It used to be highly productive, significant and pleasant. Beside this, Patrick is a very unique person. Whenever he joins a table, it is certainly the happiest one. I was indeed lucky to spend my last three years working with him.

Equally, a big thank would go to my second supervisor Dr. Vincent RICORDEL. I appreciate his monitoring, daily interactions and in deep review of my work. His efforts were the most important to realize this work. Along with this, I am thankful to his help in several aspect of daily living. He was a special friend, who is much more experienced in life. With his presence, and also his family's, I had the feeling that I have my relatives in France.

Along with the supervisors, I am thankful to the jury members who carefully reviewed my work, and provided their comments on improving the manuscript. I appreciate also the fact that they gave a positive feedback on my work, and awarded me the PhD degree. In addition, I would like to thank the thesis committee, which consists of Prof. Olivier DEFORGES (Institut National des Sciences Appliquées de Rennes) and Prof. Jens-Rainer OHM (Institut fuer Nachrichtentechnik (IENT) in RWTH Aachen University) and my two supervisors, for the annual review of the work progress, accompanied by fruitful discussions. Thanks for providing your opinion and thoughts that have shaped this work.

I would like also to thank my colleagues in the Image Perception and Interaction (IPI) team, formerly Image and Video Communications (IVC) team. Our scientific discussions have helped me a lot in improving this work. I also appreciate your help in dedicating your time for several subjective experiments. Thanks also to the management bodies who facilitated my arrival to Nantes, and also provided all the necessary logistic aid during this work. I constantly felt as being well integrated into this research team, enjoying every social, sport, research and outreach activity, and of course the traditional "pot" events. Thanks Ahmed, Yashas, Suiyi, Lukas, Dimitri, Filippo, Romain, Laurent, Remy, Lucile, Ting and Manish for your nice companionship. Along with those people,

a full chapter of this thesis would be needed to acknowledge each member of that team, including the permanent, temporal and guest members.

My sincere thanks also go once more to Prof. Jens-Rainer OHM for providing me an opportunity to join his video coding team for three months internship. I appreciate his dedication and contribution to this work. I am also thankful for Dr. Mathias WIEN from IENT, for sharing his ideas, providing stimulating thoughts and also for his logistic help that made my stay in Aachen highly comfortable and convenient. I also acknowledge that in IENT, I learned the alphabets of signal processing and video coding. This was during my master studies in RWTH. My gratitude is also for Uday THAKUR, whom I closely worked with during this internship. It was my pleasure working together, which led to an interesting joint publication and promising research direction. Another thank is for the friendly help and hospitality of the other members of IENT, especially Olena CHUBACK, Hossein BAKSHI-GOLESTANI, Abin JOSE, christian FELDMANN, Jens SCHNEIDER, Johannes SAUER and Celemens JANSEN.

I would like also to thank Mr. Philippe WETZEL, the CEO of VITEC, who showed an interest in my research, with aim of integrating the research ideas into VITEC products. I am thankful also for the offering me a visit to the enterprise, with a close interaction with the researchers and engineers there. Thanks also for Dr. Jean-Marc THIESSE, who dedicated his time for discussing and evaluation of the Local Texture Synthesis algorithm that is developed within this work.

My greatest thank goes to my family. Their support and encouragement has paved the way towards this achievement. Thanks a lot Zahraa for being a wife that understands the researcher's life, and making it more interesting, motivating and joyful. Thanks for your continuous love and emotional support. Thanks also for my family in Iraq, father, mother and brother, for your constant support and encouragement. With your wishes and prayers, I was able to make this. Thanks also for all the extended family (the family of my wife) and all the friends who were beside me and providing me with a positive energy to make this achievement.

Last, but not least, this work was supported by the Marie Sklodowska-Curie under the PROVISION (PeRceptually Optimized VIdeo CompresSION) project bearing Grant Number 608231 and Call Identifier: FP7-PEOPLE-2013-ITN. I cannot imagine having better conditions for a PhD than those in PROVISION: a generous salary, periodic training, a wide network, top academic and industrial partners and most importantly great project-mates. With more than 15 researchers in PROVISION, I had the top training in both technical and transferrable skills, and also had the most exciting days and evenings. I will always remember our jokes and funny times, and will be looking forward for the next common project.

Contents

List of Tables

List of Figures

1

Introduction

The visual scene is a rich source of information. A huge amount of data is daily perceived by our human visual system, and intelligently processed to grasp the important and meaningful visual cues. In other words, this huge data is massively compressed and stored in our memory, in a way much more advanced than our current capturing and storing technologies.

The visual information can broadly be classified into two categories: structures and textures. The structures represent the shape of the scene, while the textures fill in the gap between structures. An example of the superposition of structures and textures is drawing a piece of art, in which the painter would draw first the structure part, composed of the dominant edges, and would then fill in the rest with textures. Due to this, the visual information conveyed by structures is much valuable when compared with textural information.

There have been several studies attempting to understand texture perception, with the main intention to reveal texture similarity. This is because judging the similarity by the human visual system is a complex task that highly diverges from the simple mathematical correlation. These studies have helped in developing many intelligent visual applications such as scene understanding, synthesis, recognition and others.

This thesis is concerned about texture perception, and aiming at providing a perceptual model for texture similarity that mimics the neural processing in the human visual system. For that, a comprehensive review of the theories of texture perception is carried out, and an extrapolation link is provided between the perception of texture images and texture videos. This perceptual model shall bridge the gap between the theories on perception and the applications in computer vision.

The fundamental application that is drawn from this work is video compression. This application is nowadays in a very high demand, due to the massive deployment of the multimedia over the internet, as well as the huge amount of contents that are daily produced, stored, and transferred. The objective is to enhance the current technology by the means of texture perception. In other words, the objective is to perceptually optimize the technology to produce a better experience.

1.1 Contribution

In this work, three main contributions have been achieved:

1. **Local Texture Synthesis**:
   This is an algorithm that can efficiently encode texture images, by utilizing a texture synthesis model. Unlike the other synthesis-based coding techniques, this one is fully compatible with the coding standard, for which no need for altering the end-users' software and/or hardware.
2. **V1-E**
   V1-E is a perceptual model of texture similarity that is generalized for both static and dynamic textures. It asymptotically follows the neural processing in the human visual system. The model has shown an excellent performance in texture recognition and retrieval, as well as prediction of visual characteristics such as distortion sensitivity.
3. **Perceptual Rate-distortion Optimization Framework**
   Based on V1-E, the texture similarity is explored in order to optimize the video coding system to provide an improved rate-quality performance. The model is a perceptual rate-distortion model, in which the perceptual distortions are visually assessed, and used to tune the encoder to reduce these distortions. The model has shown significant improvement over the state of the art video compression techniques.

1.2 Thesis Organization

The thesis is organized in three parts, as shown in Fig. 1.1. In the first part (Part I), a comprehensive review on state of the art about visual texture perception is presented, along with texture similarity and its application in image and video compression. It starts in Chapter 2 with definition of textures, and theories about texture perception. Then, it moves forward to the concept of texture similarity, covering most of the approaches and tools for similarity estimation. In the last chapter (Chapter 3), an overview

of the use of texture similarity in image and video compression is given, with a brief overview about first contribution of the local texture synthesis algorithm.

Part II is dedicated to the proposed perceptual model of texture similarity, which is named as V1-E. The main theory and reasoning are given in Chapter 4. The performance evaluations are carried out in two chapters. First, the evaluation of V1-E as a similarity metric in the context of texture retrieval and recognition is carried out in Chapter 5. Second, it was evaluated as a features extractor, to predict the visual properties associated with texture that are in link with video compression is given Chapter 6.

The last part is concerned about the employment of the model in the video compression scenario. It consists of two chapters. Chapter 7 describers the proposed framework of perceptual optimization, accompanied by initial experimentation with texture images. The full implementation, subjective evaluation and generalization is then provided in chapter 8.

The thesis is concluded in Chapter 9, where the summary and outlook are provided. The manuscript encompasses also two appendixes, where Appendix A shows the generated texture video dataset (HomoTex), and Appendix B provides a list of the generated scientific papers during this PhD work.

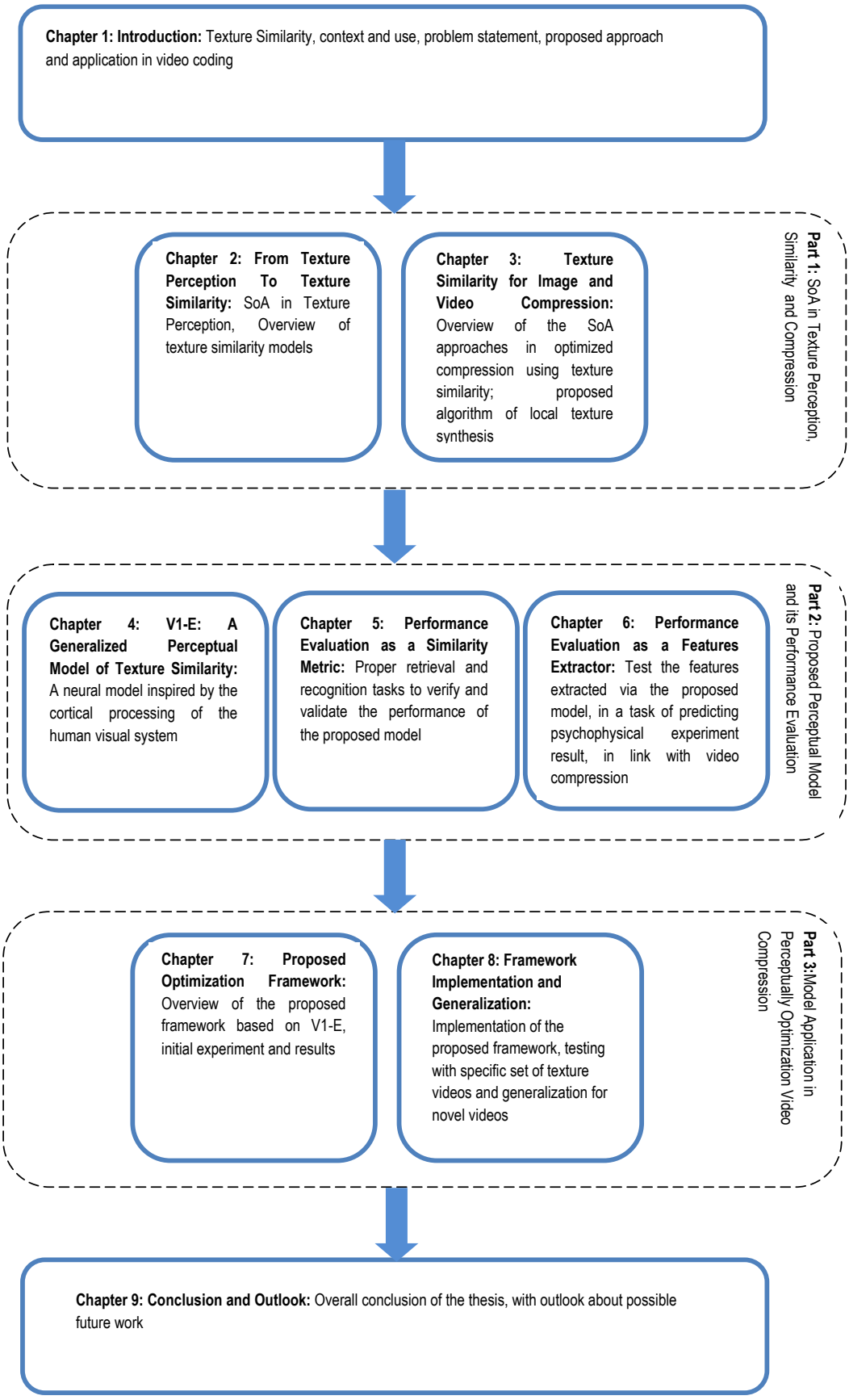


Figure 1.1 – Overview of thesis organization.

SoA in Texture Perception, Similarity and Compression

2

From Texture Perception to Texture Similarity

Textures are homogeneous visual phenomena commonly appearing in the visual scene. They are usually characterized by random structures with some stationarity. They have been well studied in different domains, such as neuro-science, vision science and computer vision, and showed an excellent performance in many applications for machine intelligence.

This chapter provides a review about texture perception, covering both classical and modern theories. After understanding this mechanism, the chapter moves forward to the concept of texture similarity, providing a survey about the existing models and their link to the perceptual studies. It also provides the information about benchmarking tools to differentiate the performance of each model.

2.1 Introduction

Textures are a fundamental part of the visual scene. They are random structures often characterized by homogeneous properties, such as color, orientation, regularity and etc. They can appear both as static or dynamic, where static textures are limited to spatial domain (like texture images shown in Fig. 2.1), while dynamic textures involve both the spatial and temporal domain Fig. 2.2.

Research on texture perception and analysis is known since quite long time. There exist many approaches to model the human perception of textures, and also many tools

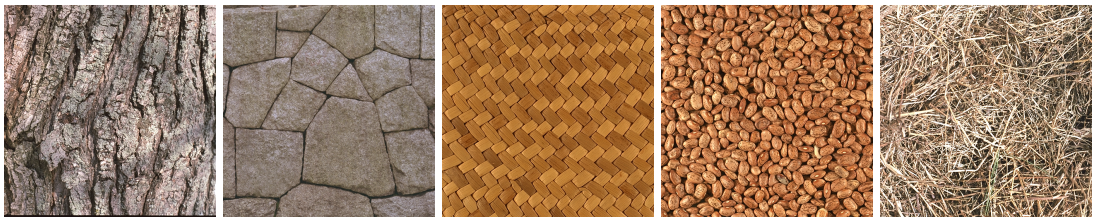


Figure 2.1 – Example of texture images from VisTex Dataset.

to characterize texture. They have been used in several applications such as scene analysis and understanding, multimedia content recognition and retrieval, saliency estimation and image/video compression systems.

Expressing texture similarity is quite a challenging task. This is because the similarity highly deviates from point-wise comparison. However, texture similarity is a key tool for many machine intelligence applications, such as recognition, classification, synthesis, etc. These applications are the key element in many modern technologies, associated with image and video understanding. For example, it could be used for scene analysis, which is employed for driving assistant system, or more generally robot vision. Another interesting application is medical image analysis, where textures are the main components of these images. The other application, which is the context of this thesis, is image and video compression, where texture similarity has often been used to improve the existing compression system (details in Chapter 3).

To elaborate more on the difficulty of expressing texture similarity, Fig. 2.3 shows examples of similar textures. In this figure, each group of the three textures has overall similar textures, but there is still a large difference if one makes a point by point comparison. Thus, the human visual system does not compute similarity using pixel comparison, but rather considers the overall similarity in the semantics. For this reason, simple difference metrics, such mean squared error, can not accurately express texture (dis-)similarity, and proper models for measuring texture similarity has always been studied. This is even more difficult in the case of dynamic textures, because there exists a lot of change in details over time, the point-wise comparison would fail to express the visual difference.

There exists a large body of reviews on texture analysis and perception. For example, the review of Landy [11][6] as well as the one from Rosenholtz [12] give a detailed overview of texture perception. Other reviews, such Tuceryan et al. in [13], cover most aspects of *static* texture analysis for computer vision applications, such as material inspection, medical image analysis, texture synthesis and segmentation. On the other side, reviews about dynamic textures, for example [14, 15], do not provide any information about the perceptual part of the textures.

In contrast to those reviews, this chapter covers the two aspects of texture perception and texture similarity. The objective is to establish the link between the perceptual stud-

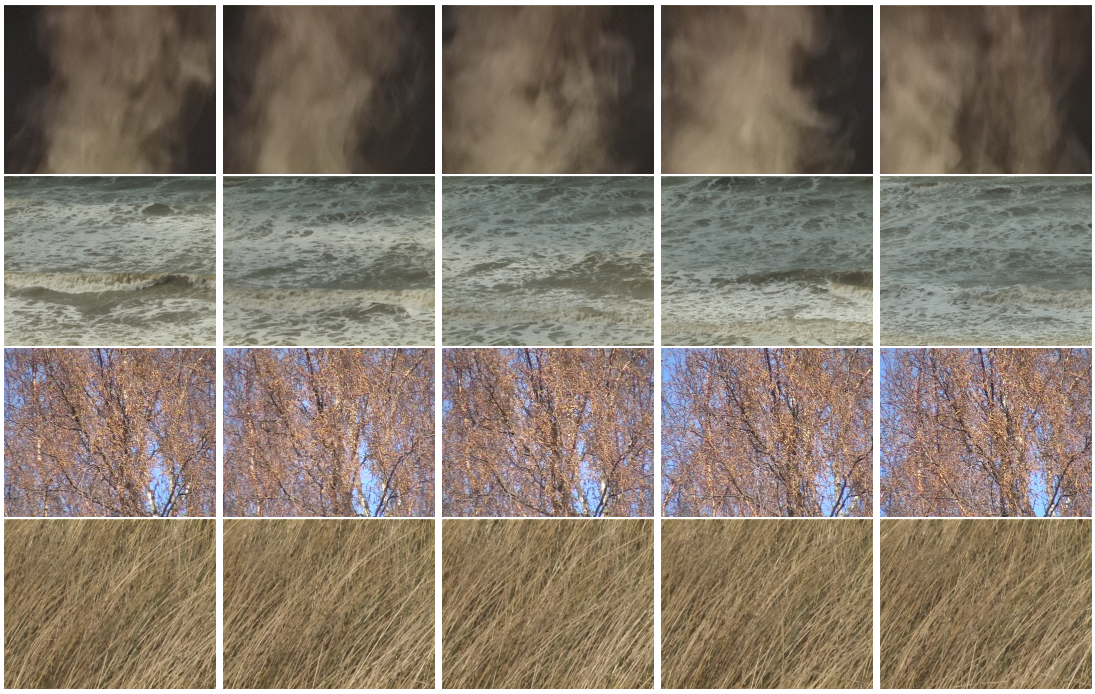


Figure 2.2 – Example of dynamic textures from DynTex Dataset [5]. First row represent the first frames, and next rows are frames after respectively 2 seconds.

ies on visual textures, and texture similarity models designed for different applications. Unlike other reviews, the chapter covers both static and dynamic textures.

The chapter is organized as follows: Section 2.2 discusses the meaning of texture in both technical and non-technical contexts, and proposes a generalized definition that is used through this work. The details of texture perception, covering both static texture and motion perception, are given in section 2.3. The different models of texture similarity are reviewed in section 2.6, with benchmarking tools in section 2.7. A general discussion and conclusion is then given in the end of the chapter in section 2.8.

2.2 What is Texture

Linguistically, the word texture significantly deviates from the technical meaning in computer vision and image processing. According to Oxford dictionary [16], the word refers to one of the followings:

1. *The way a surface, substance or piece of cloth feels when you touch it.*
2. *The way food or drink tastes or feels in your mouth.*
3. *The way that different parts of a piece of music or literature are combined to create a final impression.*

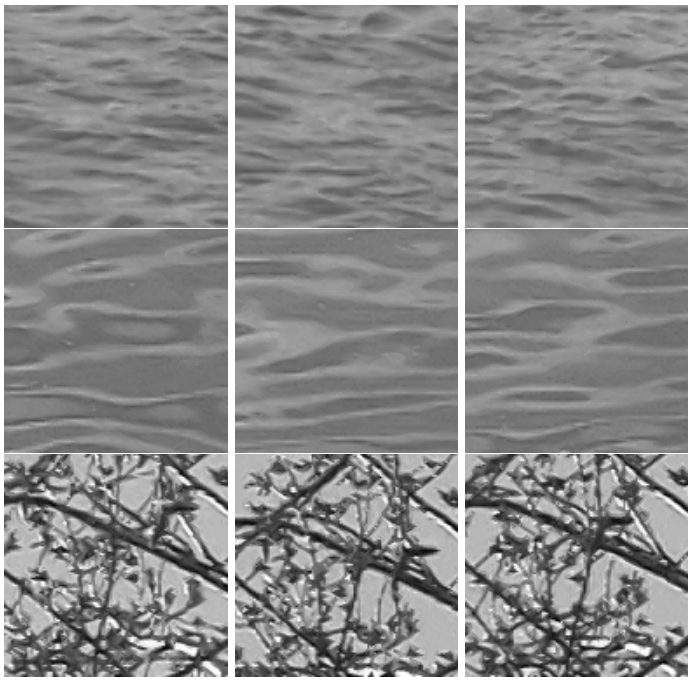


Figure 2.3 – Three examples of similar textures, having a large pixel-wise differences. These images were cropped from dynamic texture videos in DynTex dataset [5].

However, technically, the visual texture has many other definitions, for example:

— *We may regard texture as what constitute a macroscopic region. Its structure is simply attributed to pre-attentive patterns in which elements or primitives are arranged according to placement order* [17].

— *Texture refers to the arrangement of the basic constituents of a material. In a digital image, texture is depicted by spatial interrelationships between, and/or spatial arrangement of the image pixels* [18].

— *Texture is a property that is statistically defined. A uniformly textured region might be described as "predominantly vertically oriented", "predominantly small in scale", "wavy", "stubbly", "like wood grain" or "like water"* [11].

— *We regard image texture as a two-dimensional phenomenon characterized by two orthogonal properties: spatial structure (pattern) and contrast (the amount of local image structure)* [19].

— *Images of real objects often do not exhibit regions of uniform and smooth intensities, but variations of intensities with certain repeated structures or patterns, referred to as visual texture* [20].

— *Textures, in turn, are characterized by the fact that the local dependencies between pixels are location invariant. Hence the neighborhood system and the*

accompanying conditional probabilities do not differ (much) between various image loci, resulting in a stochastic pattern or texture [21].

— *Texture images can be seen as a set of basic repetitive primitives characterized by their spatial homogeneity* [22].

— *Texture images are spatially homogeneous and consist of repeated elements, often subject to some randomization in their location, size, color, orientation* [23].

— *Texture refers to class of imagery that can be characterized as a portion of infinite patterns consisting of statistically repeating elements* [24].

— *Textures are usually referred to as visual or tactile surfaces composed of repeating patterns, such as a fabric* [25].

The above definitions cover mostly the static textures, or spatial textures. However, the dynamic textures, unlike static ones, have no strict definition. The naming terminology changes a lot in the literature. The following names and definitions are summary of what is defined in research:

— Temporal Textures:

1. They are class of image motions, common in scene of natural environment, that are characterized by structural or statistical self similarity [26].
2. They are objects possessing characteristic motion with indeterminate spatial and temporal extent [27].
3. They are textures evolving over time and their motion are characterized by temporal periodicity or regularity [28].

— Dynamic Textures:

1. They are sequence of images of moving scene that exhibit certain stationarity properties in time [29][14].
2. Dynamic textures (DT) are video sequences of non-rigid dynamical objects that constantly change their shape and appearance over time [30].
3. Dynamic texture is used with reference to image sequences of various natural processes that exhibit stochastic dynamics [31].
4. Dynamic, or temporal, texture is a spatially repetitive, time-varying visual pattern that forms an image sequence with certain temporal stationarity [32].
5. Dynamic textures are spatially and temporally repetitive patterns like trees waving in the wind, water flows, fire, smoke phenomena, rotational motions [33].

— Spacetime Textures:

1. The term “spacetime texture” is taken to refer to patterns in visual spacetime that primarily are characterized by the aggregate dynamic properties of elements or local measurements accumulated over a region of spatiotemporal support, rather than in terms of the dynamics of individual constituents [34].

— Motion Texture:

1. Motion textures designate video contents similar to those named temporal or dynamic textures. Mostly, they refer to dynamic video contents displayed by natural scene elements such as flowing rivers, wavy water, falling snow, rising bubbles, spurting fountains, expanding smoke, blowing foliage or grass, and swaying flame [35].

— Texture Movie:

1. Texture movies are obtained by filming a static texture with a moving camera [36].

— Textured Motion:

1. Rich stochastic motion patterns which are characterized by the movement of a large number of distinguishable or indistinguishable elements, such as falling snow, flock of birds, river waves, etc. [37].

— Video Texture:
Video textures are defined as sequences of images that exhibit certain stationarity properties with regularity exhibiting in both time and space [38].

It is also worth mentioning that in the context of component based video coding, the textures are usually considered as details irrelevant regions, or more specifically, the region which is not noticed by the observers when it is synthesized [39, 40, 41].

As seen, there is no universal definition of the visual phenomena of textures, and there is a large dispute between static and dynamic textures. Thus, for this work, we consider the visual texture as:

A visual phenomenon, that covers both static and dynamic textures, where static textures refer to us as homogeneous regions of the scene that are typically composed of small elements (texels) arranged in a certain order, they might exhibit simple motion such as translation, rotation and zooming. On the other hand, dynamic textures are textures that evolve over time, allowing both motion and deformation, with certain stationarity in space and time.

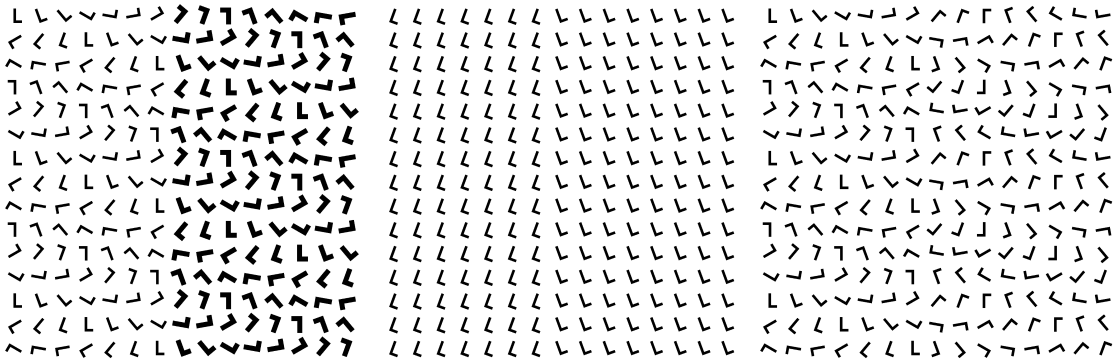
2.3 Static Texture Perception

Static texture perception has attracted the attention of researchers since decades. There exists a bunch of research papers dealing with this problem. Most of the studies attempt to understand how two textures can be visually discriminated, in an effortless cognitive action known as pre-attentive texture segregation.

Julesz extensively studied this problem. In his initial work in [42, 43], he posed a question if the human visual system is able to discriminate textures, generated by a statistical model, based on the k^{th} order statistics, and what is the minimum value of k

that beyond which the pre-attentive discrimination is not possible any more. The order of statistics refers to the probability distribution of the pixels values, in which the 1^{st} order measures how often a pixel has certain color (or luminance value), while the 2^{nd} order measures the probability of obtaining a combination of two pixels (with a given distance), and the same can be generalized for higher order statistics.

First, Julesz conjectured that the pre-attentive textures generated side-by-side, having identical 2^{nd} order statistics but different 3^{rd} order and higher, cannot be discriminated without scrutiny. In other words, textures having difference in the 1^{st} and/or 2^{nd} order statistics can be easily discriminated. This can be easily verified with the textures given in Fig. 2.4. The textures are generated by a small texture element (letter L) in three manners. The first one (Fig. 2.4a) is by having differences in the 1^{st} order statistics, in which the probability of black and white pixels is altered (different sizes of L). The second one (Fig 2.4b) is by having differences in the 2^{nd} order statistics (with identical 1^{st} order statistics). This is done by rotating one texture with respect to the other. The third (Fig. 2.4c) is by having difference in 3^{rd} order statistics (with identical 1^{st} and 2^{nd} order statistics) by using a mirror copy of the texture element (L) in the right texture. One can easily observe that conjecture holds here, as we just observe the differences pre-attentively when the difference is below the 2^{nd} order statistics. In the original work of Julesz in [43], one can find also similar examples that support this conjecture.



(a) Textures pair with different 1^{st} order statistics

(b) Textures pair with different 2^{nd} order statistics

(c) Textures pair with different 3^{rd} order statistics

Figure 2.4 – Examples of pre-attentive textures discrimination. Each image is composed of two textures side-by-side. (a) and (b) are easily distinguishable textures because of the difference in the 1^{st} and the 2^{nd} order statistics (resp.), while (c), which has identical 1^{st} and the 2^{nd} but different 3^{rd} order statistics, is not.

However, it was realized then it is possible to generate other textures having identical 3^{rd} order statistics, and yet pre-attentively discriminable [44]. An example is provided

here in Fig. 2.5. In this figure, the two side-by-side textures are generated by four squares (2x2 squares) texture element, in which the left texture has an even number of black (or white) blocks in each of its 2x2 squares, whereas the right one has an odd number. This led to the modified Julesz conjecture and the introduction of the texton theory [45]. The theory proposes that the *pre-attentive texture discrimination system cannot globally process third or higher order statistics, and that discrimination is the results of few local conspicuous features, called textons.* This has been previously highlighted by Beck [46], where he proposed that the discrimination is a result of differences in first order statistics of local features (color, brightness, size, etc.).

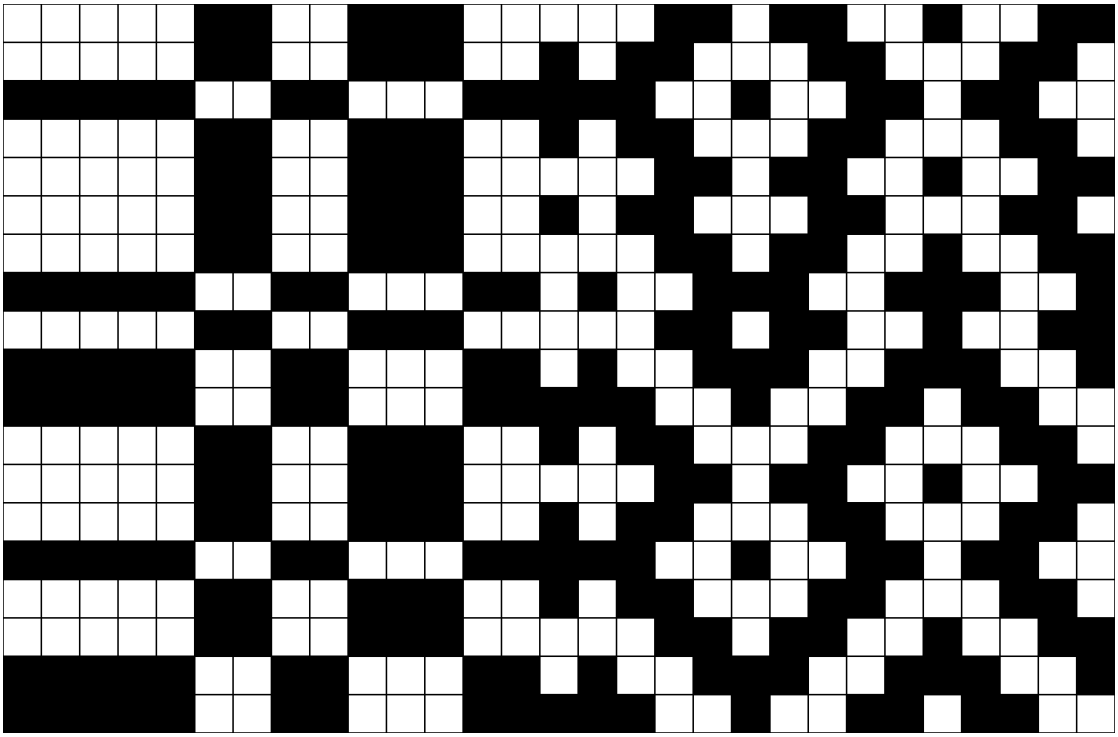


Figure 2.5 – Example of 2 textures (side-by-side) having identical 3^{rd} order statistics, yet pre-attentively distinguishable.

On the other side, with the evolution of the neurophysiological studies in the vision science, the research on texture perception has evolved, and several neural models of human visual system (HVS) were proposed. The functionality of the visual receptive field in [47], has shown that HVS, or more specifically the visual cortex, analyzes the input signal by a set of narrow frequency channels, resembling to some extent the Gaborian filtering [48]. Accordingly, different models of texture discrimination have been developed, based on Gabor filtering [49, 50], or difference of offset Gaussians [51], etc. These models are generally performing the following steps:

1. Multi-channel filtering
2. Non linearity stage
3. Statistics in the resulting space

The state of the art texture perception model based on the multi-channel filtering approach is known as back-pocket model (according to Landy [11, 6, 12]). This model, shown in Fig. 2.6, is employed in the task of texture discrimination. It consists of three fundamental stages: linear-, non-linear- and linear-filtering. For this reason, the back-pocket model is sometimes called LNL model (for Linear - Non-linear - Linear process) or FRF model (for Filter - Rectify - Filter process) where the non-linearity stage is considered as rectification process. The first linear stage accounts for the linear filtering that resembles the spatial filtering in the visual cortex. This is followed then by a non-linear stage, which is described as full-wave or half-wave rectification. This stage is required to avoid negative responses. It can also account for lateral neurons interactions, in the sense that a certain neuron response is attenuated by the surrounding ones in a phenomenon known as lateral inhibition. It can also account for response normalization, depending on the model design. The last stage is also another layer of spatial filtering, which performs averaging of higher spatial support than the first stage, such that the responses are combined for large spatial area. Then, the responses from all the neurons are combined, with a certain pooling and decision mechanism, to deduce the discrimination boundaries.

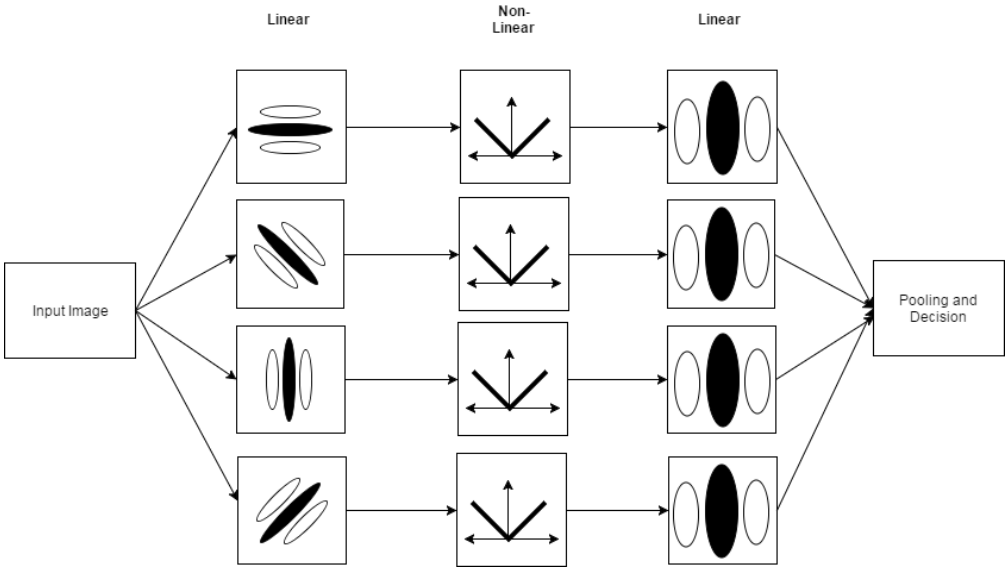


Figure 2.6 – The Back-pocket perceptual texture discrimination model [6] showing the three layers of linear-, non-linear- and linear-filtering.

2.4 Motion Perception

Texture videos, as compared to texture images, add the temporal dimension to the perceptual space. Thus, it is important to include the temporal properties of the visual system in order to understand its perception. For this reason, the section provides an overview of studies on motion perception.

The main unit responsible for motion perception is the visual cortex [52]. Generally, the functional units of the visual cortex, which are responsible for motion processing, can be grouped into two stages:

1. **Motion Detectors**

   The motion detectors are the visual neurons whose firing rate increases when an object moves in front of the eye, especially within the foveal region. Several studies have shown that the primary visual cortex area (V1) is the place where the motion detection happens [53, 54, 55, 56]. In V1, simple cells neurons are often modeled as spatio-temporal filters that are tuned to a specific spatial frequency and orientation and speed. On the other hand, complex cells perform some non-linearity on top of the simple cells (half/full wave rectification and etc.).

   The neurons of V1 are only responsive to signal having the preferred frequency-orientation-speed combination. Thus, there is still a lack of the motion integration from all neurons. Besides, the filter response cannot cope with the aperture problem. As shown in Fig. 2.7, the example of the signal in the middle of the figure shows a moving signal with a certain frequency detected to be moving up, while it could actually be moving up-right or up-left. This is also true for the other signals in the figure.

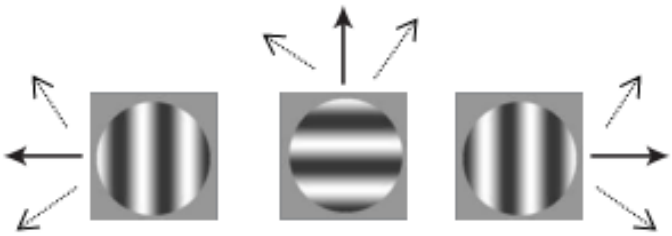


Figure 2.7 – Examples of the aperture problem: Solid arrow is the detected direction, and the dotted arrow is the other possible directions.

2. **Motion Extractors**

   The motion integration and aperture problem are solved at a higher level of the visual cortex, namely inside the extra-striate middle temporal (MT) area. It is

generally assumed that the output of V1 is directly processed in MT in a feed-forward network of neurons [53, 57, 58, 54]. The velocity vectors computation in the MT cells can be implemented in different strategies. First, the intersection of constraints, where the velocity vectors will be the ones that are agreed by the majority of individual motion detectors [53][59][60]. Other than that, one can consider maximum likelihood estimation, or a learning based model if the ground truth is available. An example of this could be MT response measured by physiological studies [54], or ground truth motion fields such as [61, 62].

It is also worth mentioning that there are other cells responsible for motion perception. For example, the medial superior temporal (MST) area of the visual cortex is responsible for motion perception during eye pursuit or headings [63, 64]. Another thing, the above review is concerning the motion caused by a luminance traveling over time, which is known as the first order motion. However, there exist the second and third order motion which is due to contrast moving and feature motion (resp.). These are outside the scope of this work. This is because the proposed perceptual model considers only the first order motion to be representative for the phenomena of textures.

2.5 Higher Order Visual Processing

The overall visual processing is depicted in Figure 2.8. Beyond the V1 area, we can differentiate two pathways. The above is called the dorsal stream, while the lower is called the ventral stream. The dorsal stream is responsible for the motion analysis, while the ventral stream is mainly concerned about the shape analysis. For this reason, the dorsal stream is known as the *"where"* stream, while the ventral is known as the *"what"* stream [52].

One plausible assumption about texture perception is that texture has no shape. This means that visual texture processing is not in the ventral stream. Beside this, one can also assume that the type of motion is not a structured motion. Thus, it is not processed by the dorsal stream as well. Accordingly, the resulting perceptual model is only due to V1 processing. That is, the perceptual space is composed of proper modeling of V1 filters a long with their non-linearity process.

In the next section, we will see that such an assumption hold, and most of the perceptually inspired texture similarity models do not go beyond V1 neural processing.

2.6 Texture Similarity Models

After reviewing the existing studies on texture perception, the goal here is to bridge this knowledge to the aspect of texture similarity. In the following subsections, several models of texture similarity models are reviewed. These models are assumed to have an

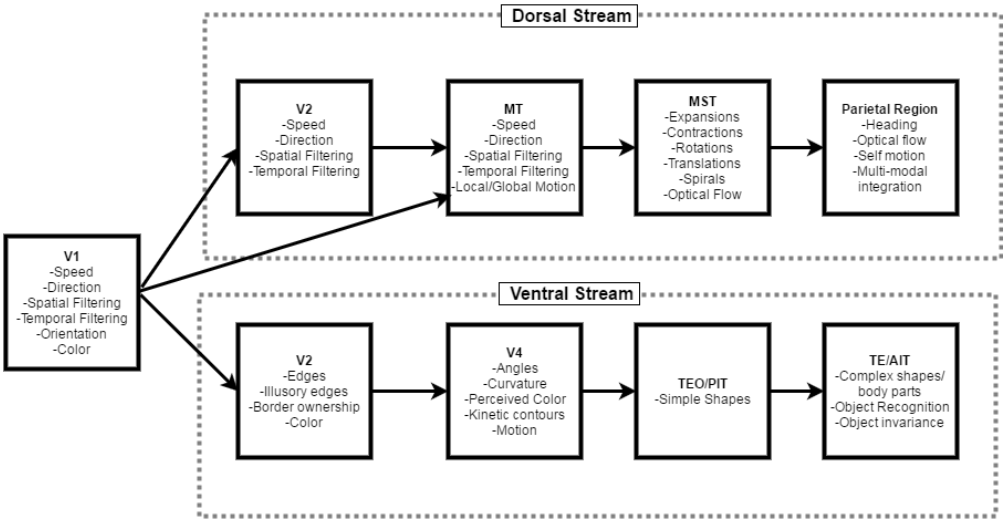


Figure 2.8 – Hierarchy of the visual system, as in [7].

inspiration of the human visual perception. Other models are thus excluded from this review.

2.6.1 Transform Based Modeling

Transform based modeling has gained lots of attention in several classical as well as recent approaches of texture similarity. This is because of the direct link with the neural processing in the visual perception. As explained in section 2.3, both neural mechanisms of static texture and motion perception involve kind of subband filtering process.

One of the first early approaches for texture similarity was proposed by Manjunath et al. [65], in which the mean and standard deviation of the texture subbands (using Gabor filtering) are compared and the similarity is assessed accordingly. Following this approach, many other similarity metrics are defined in a similar way, using different filtering methods or different statistical measures. For example, the Kullback-Leiber divergence on wavelet coefficients is used in [66] and [67]. Other approach is by using the steerable pyramid filter [68] and considering the dominant orientation and scale [22].

Knowing the importance of subband statistics, Heeger et al. proposed to synthesize textures by matching the histogram of each subband of the original and synthesized textures. To overcome the problem of irreversibility of Gabor filtering, they used the steerable pyramid filter [68]. The resulting synthesized textures were considerably similar to the original, especially for the case of highly stochastic textures. The concept has also been extended by Portilla et al. [23], where larger number of features defined in the subband domain are matched, resulting in a better quality of synthesis.

The significance of the subband statistics has led more investigation of texture similarity in that domain. Recently, a new class of similarity metrics, known as structural similarity, has been introduced. The structural texture similarity metric (STSIM) was first introduced in [69]. Then, it was enhanced and further developed in [70], [2] and [71]. The basic idea behind them is to decompose the texture, using the steerable pyramid filter, and measure statistical features in that domain. The set of statistics of each subband contains the mean and variance. Besides, the cross correlation between subbands is also considered. Finally, these features were fused to form a metric that showed a high performance in texture retrieval.

The filter-bank approach, which was applied for static textures, has been also used in dynamic texture modeling by several studies. However, the concept was used in a much smaller scope compared to static textures. In [72], three dimensional wavelet energies were used as features for textures. A comparison of different wavelet filtering based approaches, which includes purely spatial, purely temporal and spatio-temporal wavelet filtering, is given in [33].

A relatively new study on using energies of Gabor filtering is found in [73]. The work is claimed to be inspired by the human visual system, where it resembles to some extent the V1 cortical processing (section 2.3).

Beside this, there exist also other series of papers, by Konstantinos et al. [34, 31], which employed another type of subband filtering, which is the third Gaussian derivatives tuned to certain scale and orientation (in 3D space). The approach was used for textures representation recognition and also for dynamic scene understanding and action recognition [74].

2.6.2 Auto-regressive Modeling

The auto-regressive (AR) model has been widely used to model both static and dynamic textures, especially for texture synthesis purposes. In its simplistic form, AR can be expressed in this form:

$$s(x,y,t) = \sum_{i=1}^{N} \phi_i s(x+\Delta y_i, y+\Delta y_i, t+\Delta t_i) + n(x,y,t) \tag{2.1}$$

Where $s(x,y,t)$ represents the pixel value at the spatio-temporal position (x,y,t), ϕ_i is the model weights. Δx_i,Δy_i and Δt_i are the shift to cover the neighboring pixels. $n(x,y,t)$ is the system noise which is assumed to be white Gaussian noise.

The assumption behind AR is that each pixel is predictable from a set of its neighboring spatio-temporal pixels, by the means of weighted summation, and the error is due to the model noise $n(x,y,t)$. An example of using model for synthesis can be found in [75, 76, 77].

The auto-regressive moving average (ARMA) model is an extension of the simple AR model that is elegantly suited for dynamic textures. It was first introduced by Soatto and Dorreto [29, 14] for the purpose of dynamic texture recognition. The ARMA model is mathematically expressed in this equation:

$$\begin{aligned} x(t+1) &= Ax(t) + v(t) \\ y(t) &= \phi x(t) + w(t) \end{aligned} \tag{2.2}$$

Where $x(t)$ is a hidden state and $y(t)$ is the output state, $v(t)$ and $w(t)$ are system noise (normally distributed) and A, ϕ are the model weights as in AR. Typically, the output state represents the original frames of the image sequence. Comparing Eqn. 2.2 with Eqn. 2.1, it is clear that the model assumes that the hidden state $x(t)$ is modeled as an AR process, and the observed state is weighted version of the hidden state with some added noise.

Both AR and ARMA can be used to measure texture similarity. This has been used in texture recognition, classification, segmentation and editing [78, 79]. Other than this, ARMA has been extended by several studies. For example, by using Fourier domain [80], by including several ARMA models with transition probability [81], using higher order decomposition [82] and others [83, 84].

Although there is no direct link between the texture perception and the auto-regressive models, we can still interpret its performance in terms of Julesz conjectures (section 2.3). The assumption behind these models is that textures would look similar if they are generated by the same statistical model with a fixed set of parameters. Similarly, Julesz has conjectured that the textures are indistinguishable if they have the same first and second order statistics. Thus, auto-regressive models can be understood as an extension of this conjecture, in which the condition for similarity is better stated.

2.6.3 Texton Based Modeling

Recalling that textons are local conspicuous features (section 2.3), a large body of research has been put to define some local features that can be used to measure the texture similarity. One of the first approaches, and still very widely used, is the local binary pattern approach (LBP) [19]. This approach is simply comparing each pixel with each of its circular neighborhood, and gives a binary number (0-1) if the value is bigger/smaller than the center value. The resulting binary numbers are gathered in a histogram, and any histogram-based distance metric can be used.

The approach has gained a lot of attention due to its simplicity and high performance. It was directly adopted for dynamic textures in two manners [85]: First, by considering the neighborhood to be cylindrical instead of circular in the case of Volume Local Binary Pattern (V-LBP); second, by performing three orthogonal LBP on the xy, xt and yt planes, which is therefore called Three Orthogonal Planes LBP (LBP-TOP).

Several extensions of the basic LBP model have been proposed. For example, a similarity metric for static textures known as local radius index (LRI)[1, 86], which incorporates LBP along with other pixels to neighbors relationship. Besides, there is another method that utilizes the Weber law of sensation, which is known as Weber Local Descriptor (WLD) [87].

Rather than restricting the neighborhood relationship to binary descriptors, other studies have introduced also trinary number [88, 89, 90] and what is known as texture spectrum.

It is also worth mentioning that some studies consider the textons as the results of frequency analysis of texture patches. The study of Liu et al. [91] considered the marginal distribution of filter bank responses as the "quantitative definition" of texton. In contrast, textons are defined [92] as the representation that results from codebook generation of a frequency histogram.

2.6.4 Motion Based Modeling

The motion based analysis and modeling of dynamic textures has been considered in large body of studies. This is because motion can be considered as a very important visual cue, and also because the dynamic texture signal is mostly governed by motion statistics. To elaborate on motion analysis, let us start with basic assumption that we have an image patch $I(x, y, t)$ in a spatial position (x, y) and at time (t), and this patch would appear in the next frame, shifted by $(\Delta x, \Delta y)$. Mathematically:

$$I(x, y, t) = I(x + \Delta x, y + \Delta y, t + 1) \tag{2.3}$$

This equation is known as *Brightness Constancy Equation*, as it states that the brightness does not change from one frame to another. Eqn. 2.3 can be simplified by employing the Taylor expansion as follows (removing the spatial and temporal indexes for simplicity):

$$I = \sum_{n=0}^{\infty} \left(\frac{I_{xn}}{n!} \times \Delta x + \frac{I_{yn}}{n!} \times \Delta y + \frac{I_{tn}}{n!} \times \Delta t \right) \tag{2.4}$$

where I_{xn}, I_{yn} and I_{tn} are the n^{th} order partial derivative with respect to x, y and t. The equation can be further simplified by neglecting the terms of order higher than one. Then the it becomes:

$$I_x \times V_x + I_y \times V_y = -I_t \tag{2.5}$$

where V_x, V_y are the velocities in x and y directions ($V_x = \Delta x / \Delta t$ and so on). The solution of Eqn. 2.5 is known as *optical flow*. However, further constraints are needed to solve the equation because of the high number of unknowns. One of the constraints is the smoothness, in which a patch is assumed to move with the same direction and

speed between two frames. This is not usually the case for dynamic texture, in which the content could possibly change a lot in a short time instant. Accordingly, there exists also another formulation of the brightness constancy assumption that does not require the analytical solution. This is known as the *normal flow*. It is a vector of flow, that is normal to the spatial contours (parallel to the spatial gradient), and its amplitude is proportional to the temporal derivative. Mathematically, it is expressed as:

$$\mathbf{NF} = \frac{-I_t}{\sqrt{{I_x}^2 + {I_y}^2}}\mathbf{N} \tag{2.6}$$

where **N** is a unit vector in the direction of the gradient.

The normal flow, as compared to the optical flow, is easy to compute. It needs only the image derivatives in the three dimensions (x, y, t), and no computation of the flow speed is needed. One drawback of normal flow is that it can be very noisy (especially for low detailed region) when the spatial derivatives are low. For this reason, a threshold is usually set before evaluating any statistical property of the normal flow.

The motion based modeling of dynamic textures was pioneered by Nelson and Palonan in [26], where they used normal flow statistics for dynamic textures classification. This model has been extended in [93] to include both the normal flow and some static texture features (coarseness, directionality and contrast). Other than that, Peteri et al. [94] have augmented the normal flow with a regularity measure, computed from correlation function.

The optical flow has been also used in dynamic texture analysis. In [95], the authors compared different optical flow approaches to normal flow, and showed that the recognition rate can be significantly enhanced by optical flow.

Similar to the concept of co-occurrence matrix, Rahman et al. have developed the concept of *motion co-occurrence* [96], in which they compute the statistics of occurrence of a motion field with another one for a given length.

It is also worth mentioning here there are other approaches beyond the concept of brightness constancy. Since dynamic textures can change their appearance over time, it is more logical to move towards *brightness conservation assumption*. It can be mathematically expressed as [97, 98]:

$$I(x, y, t)(1 - \Delta x_x - \Delta y_y) = I(x + \Delta x, y + \Delta y, t + 1) \tag{2.7}$$

Where Δx_x and Δy_y are the partial derivatives of the shifts in x and y. Comparing this equation to Eqn. 2.3, the model allows the brightness I to change over time to better cover the dynamic change inherited in the dynamic textures. The model has been used for detecting dynamic textures [97], in which regions satisfying this assumption are considered as dynamic textures. However, further extensions of this idea were not found.

2.6.5 Others

Along with other aforementioned models, there exist other approaches that cannot be straightforwardly put in one category. This is because the research on texture similarity is quite matured, but still very active.

One major approach for modeling texture and expressing similarity is by using the fractal analysis. It can be simply understood as an analysis of measurements at different scales, which in turn reveals the relationship between them. For images, this can be implemented by measuring the energies of a Gaussian filter at different scales. The relationship is expressed in terms of the fractional dimension. Recent approaches of fractal analysis can be found in [99, 100, 101].

Another notable way is to use the self avoiding walks. In this, a traveler walks through the video pixel using a specified rule and memory to store the last steps. A histogram of walks is then computed and considered as features for characterizing the texture (cf. [102, 103]).

Beside these, there exist also other models that are based on the physical behavior of textures (especially dynamic textures). This includes models for fire [104], smoke [105] and water [106].

Although these models suit very well specific textural phenomenon, they cannot be considered as perceptual ones. This is because they are not meant to mimic the visual processing, but rather the physical source. For this reason, these are out of scope of this work.

2.7 Benchmarking and Comparison

After viewing several approaches for assessing the texture similarity (section 2.6), the fundamental question here is how to compare these approaches, and to establish a benchmark platform in order to differentiate the behavior of each approach. This is of course not a straightforward method, and a reasonable construction of ground truth data is required.

Broadly speaking, comparison can either be performed *subjectively* or *objectively*. In other words, either by involving observers in a kind of psycho-physical test, or by testing the similarity approaches performance on a pre-labeled dataset. Both have advantages and disadvantages, which are explained here.

The subjective comparison is generally considered as the most reliable one. This is because it directly deals with human judgment on similarity. However, there are several problems that can be encountered in such a methodology. First is the selection and accuracy of the psycho-physical test. For example, a binary test can be the simplest for the subjects, and would result in very accurate results. In contrast, this test can be very slow to cover all the test conditions, and possibly such a test would not be suitable.

Metric	Retrieval rate (%)
PSNR	4
LBP	90
Wavelet Features [66]	84
Gabor Features [65]	92
STSIM	96
LRI	99

Table 2.1 – Retrieval rate as a benchmark tool for different texture similarity metrics. Results obtained from [1, 2].

Second, the budget-time limitation behind the subjective tests would result in a limited testing material. Thus, it is practically unfeasible to perform a large scale comparison with subjective testing.

Accordingly, there exist few studies on the subjective evaluation of texture similarity models. For example, the subjective quality of synthesized textures were assessed and predicted in [107, 38], and adaptive selection among the synthesis algorithms was provided in [108]. The similarity metrics correlation with subjective evaluation was also computed in [109, 110].

As explained earlier, subjective evaluation suffers from test accuracy and budget time-limitation. One can also add the problem of irreproducibility, in which the subjective test results cannot be retained after repeating the subjective test. There is also a certain amount of uncertainty with the results, which is usually reported in terms of confidence levels. To encounter this, research in computer vision is usually led by objective evaluations.

One commonly used benchmarking procedure is to test the performance on recognition task. For static textures, two large datasets of 425 and 61 homogeneous texture images are cropped into 128x128 images with substantial point-wise differences [2]. The common test is to perform a retrieval test, in which for a test image if the retrieved image is from the correct image source, it is considered as correct retrieval. This is performed for all of the images in the dataset, and the retrieval rate is considered as the criteria to compare different similarity measure approaches. For example, Table 2.1 provides the information about the performance of different metrics. In this table, one can easily observe that simple point-wise comparison metric like the Peak Signal to Noise Ratio (PSNR) provides the worst performance.

For dynamic textures, a similar task is defined. Commonly, the task consists of classification of three datasets. These are the UCLA [111], DynTex [5] and DynTex++ [112] datasets. For each dataset, the same test conditions are commonly used. For example, DynTex++ contains 36 classes, each of 100 exemplar sequences. The test

Metric	Recognition Rate (%)
VLBP	94.98
LBP-TOP	94.05
WLBPC [113]	95.01
CVLBP [3]	96.28
MEWLSP [4]	98.48

Table 2.2 – Recognition rate on the DynTex++ as a benchmark tool for different texture similarity metrics. Results obtained from [3, 4].

condition is to randomly assign 50% of the data for training and the rest for testing. The train data are used for training the models, and the recognition rate is reported for the test data. The procedure is repeated 20 times and the average value is retained. This is shown in Table 2.2.

2.8 Discussion and Conclusion

This chapter reviewed the studies about texture perception, and linked them to the existing models of texture similarity. The chapter focused on both static and dynamic textures.

First, it has been realized that there is a lack of universal definition of the textures, in the sense that static textures tends to be differently defined when compared to dynamic textures. Besides this, there is a large controversy on understanding what dynamic textures are. Since the scope of this thesis is both static and dynamic textures, a proper definition was developed, that is meaningful and covers what we understand as a textural phenomena.

Second, it was observed that despite the extensive studies on static texture perception, dynamic textures are not yet well explored. This is because most of the perceptual studies consider texture images rather than texture videos. The chapter thus reviewed the static texture perception, and included also motion perception, in order to cover all the visual processing that could be associated with dynamic textures. The overview of the neural processing carried out in this chapter revealed that the fundamental part of the visual processing for texture is assumed not to happen in higher levels of the visual cortex. In fact, the primary visual cortex (V1) was assumed to be responsible for texture perception. This is because textures are generally considered as regions without specific shapes and do not have any structured motion. This indicates that the further processing in the ventral stream (shape stream) or the dorsal stream (motion stream) is not necessary to explain texture perception. This is the keystone argument in the proposed perceptual texture model that is developed in this thesis, which is presented in

Chapter 4.

Third, the chapter linked the knowledge of texture perception to texture similarity. An overview of the existing similarity models for both static and dynamic textures was provided. Different models have been classified into 4 categories: Transform-, Auto-regressive-, Texton- and Motion-based modeling. The transform based modeling can be considered as a direct link between texture perception and texture similarity. This is because it is based on the fact that a similar transform occurs in the V1 cortical processing area, and further processing is not fully known. However, it was realized that such an approach is mostly limited to static textures, which ignores its potential use for dynamic ones. Texton based modeling is highly indirect, as it assumes that the similarity is due to the distribution of the texture element, and ignores the actual neural processing. The difference between these two approaches, transform and texton modeling, is usually referred to the bottom-up vs. up-down modeling. Bottom-up modeling of the human visual system starts from the actual/observed low level processing, up to the cognition level, whereas the top-down modeling formulate a hypothesis about the processing, and validate it accordingly. The other two models are, to some extent, less perceptual and more computational models. This is because the human visual system is, for sure, neither following any auto-regressive model nor computing the motion as those models assume. The chapter also listed other models that are purely non perceptual models, but rather follow the properties of the textures themselves.

Finally, the chapter included also benchmarking tools for assessing the performance of different models. The pros and cons of subjective and objective evaluations are highlighted. Due to the simplicity, and the capability of performing large-scale testing, the objective evaluations are generally preferred. For both static and dynamic textures, common testing conditions have already been established. The evaluation results have clearly shown that simple metrics, based on pixels comparison such as PSNR, cannot be used for computing texture similarity.

3

Texture Similarity for Image and Video Compression

Image/Video compression is the key technology that enables several applications related to storage and transmission. The current technologies enable content capturing with large spatio-temporal resolutions, which end up with massive amount of data. For this reason, it is practically impossible to deploy uncompressed content because of the limited physical memory and transmission channel capacity. Thus, compression is the only solution in the current applications related to multimedia.

As textures represent a major part of visual scenes, there have been different studies on how to efficiently compress this part. This can be either by exploiting the intrinsic properties of textures, or utilizing the perceptual aspects of them. Among the different approaches, there exists a large body of methods for texture similarity based image and video compression, which is in line with this work. Accordingly, this chapter is dedicated to provide an overview of the state of the art image/video compression techniques, utilizing texture similarity models.

The chapter is organized as follows: The introduction is given in section 3.1. The overview of the state of the art video compression standard, known as HEVC, is provided in section 3.2. The texture based approaches are given in section 3.3 and section 3.4, while the conclusion is given in section 3.5.

3.1 Introduction

Visual scene is characterized by high amount of redundancies in both spatial and temporal domain. For example, if one analyzes a given image or video of natural scene, it would be observed that there is a large correlation between neighboring pixels. This is also true for the temporal domain, because the amount of change from one frame to another is not expected to be exceptionally high.

Due to this, the image and video compression techniques are based on exploiting these redundancies to yield a compact representation of the visual signal. This is mainly achieved by the prediction mechanism (intra- and inter-picture prediction) and the transform coding. Beyond this, entropy coding is employed to further exploit the statistical redundancies inherited in the visual scene.

Beyond these redundancies, the image/video compression techniques further compress the visual signal by the means of quantization. This step provides a lossy compression that can highly reduce the amount of bits needed to encode the scene.

Since the scope of this thesis is mainly about video compression, the following sections will mostly focus on the video compression part.

3.2 State of the Art in Video Compression

Video compression techniques have been evolving in terms of successive standards. Currently, the latest standard is known as the High Efficiency Video Coding (HEVC) [114]. It has shown a significant improvement over the previous standard, known as Advanced Video Coding (AVC), in the sense that it provides up to 50% bitrate saving [115], where both objective and subjective verifications are performed.

To give a brief overview of HEVC, Fig. 3.1 depicts the block diagram of its reference encoder. First, it divides the input frame into equal size square units called Coding Tree Units (CTU). CTU's have a maximum dimension of 64x64 pixels for luma channels, and can be further divided into smaller units, called coding units (CU), in a quad-tree manner. A prediction unit (PU), which can either be the full CU or a part of it, is the unit where intra- or inter-picture prediction is performed. The residual signal (after prediction) is then quantized. The quantized values are then binary encoded using the entropy encoder and saved in the bitstream.

HEVC is very flexible in terms of block partitioning. While CTU can be partitioned into several CU's in quad-tree manners, each CU can also be portioned into several transform units (TU) and PU's. Furthermore, the restriction for square blocks is not required for PU's, thus allowing high flexibility for motion compensation, and resulting in efficient compression system.

It is also worth mentioning that beside HEVC, there exist other video compression standards (for example, the VP8, VP9 and VP10 [116] series of standards). These are

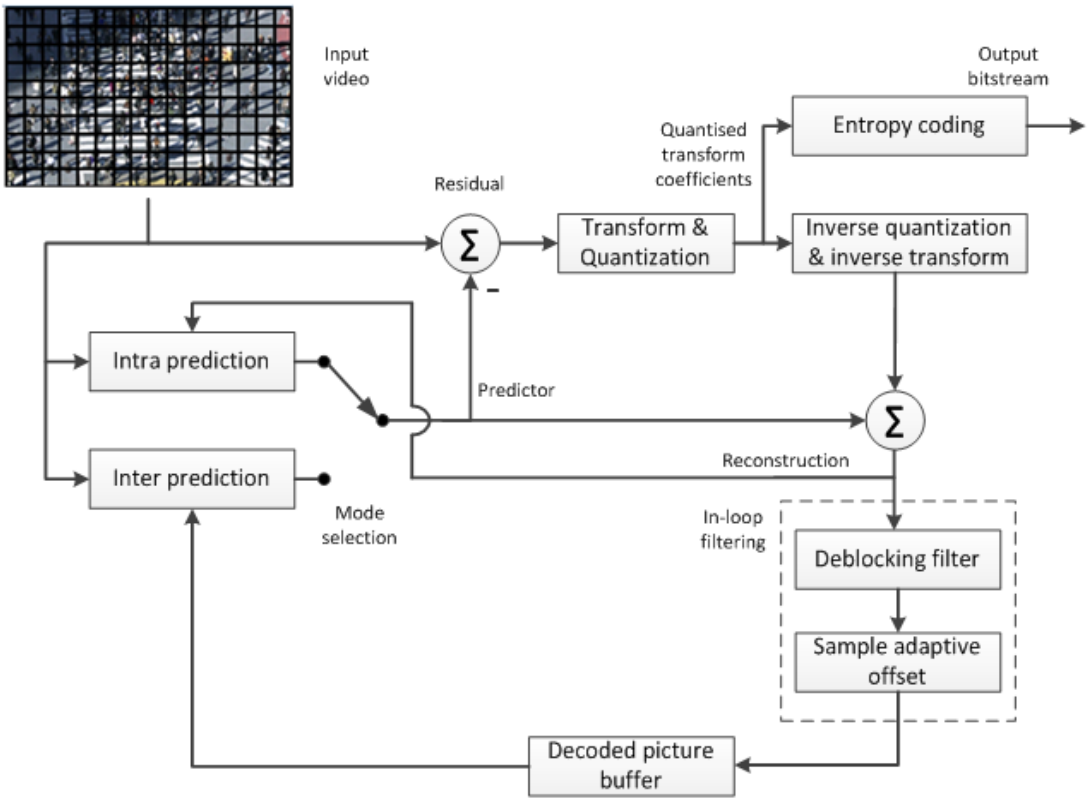


Figure 3.1 – Simplified reference HEVC encoder [8].

not covered in this thesis as they did not show an improvement over HEVC [117]. Thus, HEVC is the state of the art benchmark for video compression standards. Other than this, these coding approaches are not fundamentally different from HEVC.

3.3 Texture Based Perceptual Compression

In the context of compression, texture is usually referred to homogeneous regions of high spatial and/or temporal activities with mostly irrelevant details. According to this, textures would usually consume high amount of bitrate for unnecessary details. Thus, a proper compression of texture signal is needed. In the following subsections, an overview of different approaches for texture similarity in video compression is provided.

3.3.1 Texture Removal Approaches

In the context of image/video compression, the common hypothesis is that texture would look similar, if a good synthesis algorithm is used. By synthesizing the texture,

there is no need to encode it but rather to encode the synthesis parameters, which need to be significantly easier to encode in order to provide an improved compression ratio.

One of the first approaches for synthesis based coding was introduced by Ndjiki Nya et al. in [118][119]. The proposed algorithm consists of two main functions: texture analyzer (TA) and texture synthesizer (TS). The TA is responsible of detecting regions of details irrelevant textures, via spatial segmentation and temporal grouping of segmented textures. The TS, on the other hand, is responsible of reproducing the removed parts in the decoder side. TS contains two types of synthesizers, one employs image warping, which is used to warp texture with simple motion (camera motion mostly), the other one is based on Markov Random Fields and is responsible for synthesizing textures containing internal motion. This algorithm was implemented in the video coding standard of H.264 [120], in which irrelevant texture signals are skipped by the encoder, and only the synthesis parameters are sent to the decoder as a side information.

Ndjiki-Nya et al. produced several extensions of the above mentioned approach. In [121], a rate distortion optimization was also used for the synthesis part. The rate is the number of bits required to encode the synthesis parameters and the distortion accounts for the similarity between the original and synthesized textures, in which they used an edge histogram as well as color descriptor for computing the objective quality. A review of their work, as well as others, is given in [122].

Similar to these approaches, many other researchers have developed texture removal algorithms varying in their compression capability, complexity, synthesis algorithm and distortion measure. Interested reader may refer to [39] and [41]. For HEVC, there exist also initial investigations about the pyramid based synthesis [123] and motion based synthesis for dynamic textures [124].

Recently, as a part of our study on texture synthesis for video compression, a new approach for texture synthesis has been proposed in [9]. In this approach, half of the frames are encoded, and the rest are synthesized based on subband linear phase interpolation. This is shown in Fig. 3.2, where each intermediate frame is skipped at the encoder side, and synthesized at the decoder side after reconstructing the previous and next frames. With this approach, significant amount of data is removed, which results in large bitrate saving. Visually, the synthesized frames as compared to the compressed frames, at a similar bitrate, are in much better quality (Fig. 3.3). There is significant reduction of the blocking artifacts. The results have been verified by subjective testing, where it was shown that observers tend to prefer the synthesis-based model against the default compression, for the equivalent bitrate levels.

3.3.2 Texture Simplification Approaches

One problem of the synthesis based approaches is the necessity of modifying the existing standard by modifying the decoder side. This is certainly undesired as it required changing the users' software and/or hardware, and thus could negatively impact

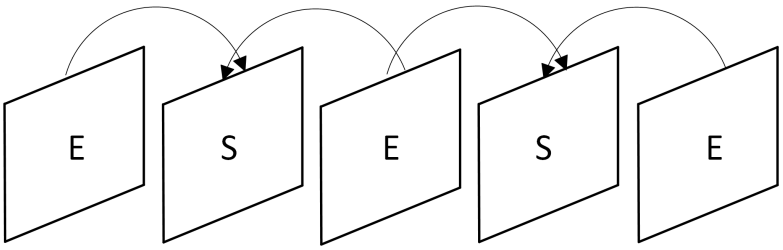


Figure 3.2 – Dynamic texture synthesis approach for alternative frames [9]. E is a decoded picture and S is synthesized one.

the user experience. To encounter this problem, Dumitras et al. in [125] proposed a "texture replacement" method at the encoder, in which the encoder synthesis some texture areas in a way that it is simpler to encode. In doing this, the encoded image/video would be the simplified synthetic signal, which would have a similar look to the original one. Accordingly, it is only a pre-processing step, which does not require any further modification of the encoder and decoder. However, the approach was only limited to background texture with simple camera motion.

In one of our studies, we presented a new *online* synthesis algorithm that is fully compatible with HEVC. It is named as Local Texture Synthesis (LTS [10]). The algorithm, as described in Fig. 3.4, generates for each block to be encoded B a set of synthetic blocks $\underline{B}$ containing n blocks ($B1$,$B2$,...,Bn) that are visually similar to B. A subset $\underline{B'}$ out of $\underline{B}$ that has blocks with good match to the given context is only maintained. Then, the encoder tries encoding block by replacing its content by the contents in $\underline{B'}$, and will then select the block Bj such that Bj has the minimum rate and distortion. Thus, the algorithm tries to replace the contents while encoding, by visually similar ones, such that the contents will be easier to encode.

An example for comparing the behavior of LTS against HEVC is shown in Fig 3.5. Due to the simplification procedure of the contents in LTS, one can achieve about 10% bitrate saving. On the other hand, there are also some visual artifacts due to this simplification. By carefully examining the differences, we can see that some of the wall boundaries are eliminated in LTS. This is because encoding an edge cost more than a flat area, and thus LTS would choose to replace this edge by another possible synthesis that is easier to encode.

3.3.3 Texture Prediction Approaches

As explained earlier, the prediction mechanism, accompanied by transform coding, are the main tool for exploiting the spatio-temporal redundancies in order to provide efficient compression system. The texture synthesis approaches can be used to predict the texture signal, and result in less residual to be encoded. Some few studies have used

Figure 3.3 – Examples of visual comparison between default compression and proposed method in [9]. Left: original frames, middle: compressed frames with HEVC and right: synthesized frames at the decoder side.

this idea, and showed its significance in video compression.

First, the spatial prediction scheme was presented in [126]. It is named as extended texture prediction, which employed Markovian model to synthesize the prediction signal. It was provided as an extra mode, which the encoder can choose if it provides better rate-distortion trade-off. Similarly, in the temporal domain, the auto-regressive moving average (ARMA) (section 2.6.2) model was used in [127] to synthesize prediction frames. These approaches have shown a notable bitrate reduction, besides, they maintain the pixels fidelity, where this property eliminates the problem of designing a reliable criterion to assess the quality of the synthesized regions.

3.4 Indirect Approaches

Beside texture synthesis based coding; there also exist several studies on perceptually optimizing the encoder based on texture properties. These studies fall generally into the category of noise shaping, where the coding noise (compression artifact) is distributed to minimize the perceived distortions. Examples can be found in

[128, 129, 130, 131, 132]. Besides, textures are considered as non-salient areas, and less bitrate is consumed there [133, 134].

The other indirect use of texture similarity measure is to exploit the analysis tools and features from that domain in image and video compression. For example, in [135], the visual redundancies of dynamic textures can be easily predicted by a set of features, such as normal flow and gray level co-occurrence matrix. Similarly, the optimal rate-distortion parameter (Lagrangian multiplier) can be predicted as well [136].

3.5 Discussion and Conclusion

In this chapter, different approaches of image and video compression, utilizing texture similarity are presented, with the first overview of the current state of the art video compression standard, known as the high efficiency video coding (HEVC). Three categories have been identified, which are: texture removal, texture simplification and texture prediction. Beyond these, there are also indirect approaches that partially utilize texture similarity for image and video compression.

Texture removal approaches omit large part of the textures, and rely on texture synthesis for generating back the missing part. These approaches are commonly used as they can provide significant bitrate saving. This is because the omitted part is encoded only by synthesis parameters, which are either easier to encode than original data, or can be derived at the decoder side. In one of the presented algorithms, by Thakur et al. [9], the experimental results showed about 50% bitrate saving on the same bitrate. The subjective evaluations showed the general tendency of preferring this algorithm over HEVC.

Texture removal approaches requires significant changes in the coding standard. This is because they add extra processing and coding units that are not available in the original standard. This would be an issue with deploying this kind of solutions in practice. The reason is that both software and hardware video coding systems follow the existing standard, so that the content can be accessed by a universal decoder. Violating the standard will directly lead to change of the end-users' existing software/hardware, which would negatively impact the user experience, unless new generation of video coding is being initiated. Another problem, which is more serious, is that a lot of complexity is added to the decoder side. It is generally tolerated to increase the complexity at the encoder side, because video are encoded once, and this can be done in the content provider side, but they are decoded as many times as they viewed, shared and received from broadcasting. Increased decoder complexity means extra processing power with the users' limited device capacity, which is why it is generally avoided.

To mitigate these problems, texture simplification approaches have been proposed. In these approaches, the synthesis is performed on the encoder side only, and the decoder is agnostic to this process. This means any standard decoder can be used to decode the

bitsteam, and view the contents. In this category, 2 approaches were identified. The first is introduced by Dumitras [125], in which some texture are replace by a synthesized ones which are simpler to encode. This idea was further extended by one of our work in [10], where each block is replaced by a set of synthesized ones, provided that some statistical constraints are met. The encoder selects the synthetic block the minimizes the immediate rate-distortion cost. Typically, this type of approaches showed lesser bitrate saving, as compared to the texture removal approaches because no omission is performed, but solve the issues associated with standard violation and decoder complexity.

As both texture removal and texture simplification approaches replace the original texture by a synthetic ones, the resulting textures would usually be pixel-wise very different. This is, of course, not a problem with this type of contents, as the overall similarity is preserved. Nevertheless, in some applications the pixel fidelity is important. For example, if one is interested in large scale evaluation of performance of the similarity based approaches, with the current standard, a proper evaluation metric is required, that is generally pixel based one such as PSNR. Other than this, it can be safer to preserve the pixel fidelity rather than perceptual fidelity, because the latter can hardly be expressed mathematically. For this purpose, another category of similarity based approaches are proposed, namely prediction based approaches. In this category, the texture synthesis is used to generate a prediction signal in the encoder, which can be used to enhance the prediction mechanism and thus the overall coding efficiency. The same problems as in the removal based approaches still exist.

Other than those approaches, the chapter provides an overview of the indirect approaches. These approaches do not utilize texture similarity, but rather texture properties to allocate distortions, or to optimize encoder parameters. A very limited overview is provided as it is outside the scope of this work.

It has been observed that all the proposed approaches do not fully exploit the knowledge about texture similarity. They rather rely on the reverse-engineered concept, in the sense that texture synthesis is employed instead of a straightforward texture similarity. This is because texture similarity and texture synthesis are mutually inter-related. For a good synthesis, the synthesized textures need to be similar to the original one. On the other hand, a good similarity model can be used for synthesis purpose by understanding what makes textures look similar, and reverse-engineer the process. This can also be understood as a top-down approach, instead of bottom up one, i.e. the hypothesis that is made here is that texture would look similar because of the used synthesis algorithm, neglecting the important part of the neural processing in the human visual system.

The lack of bottom-up approach was the main motivation behind this work. The work proposes a perceptual model of textures (Part II), which can be used both as a similarity metric as well as a features extractor. The aim is to utilize this to perceptually optimize the video coding system to fill the hole with a new bottom-up approach. Due to

the several issues identified with violating the standard, we aimed at providing a solution with full compatibility (Part III).

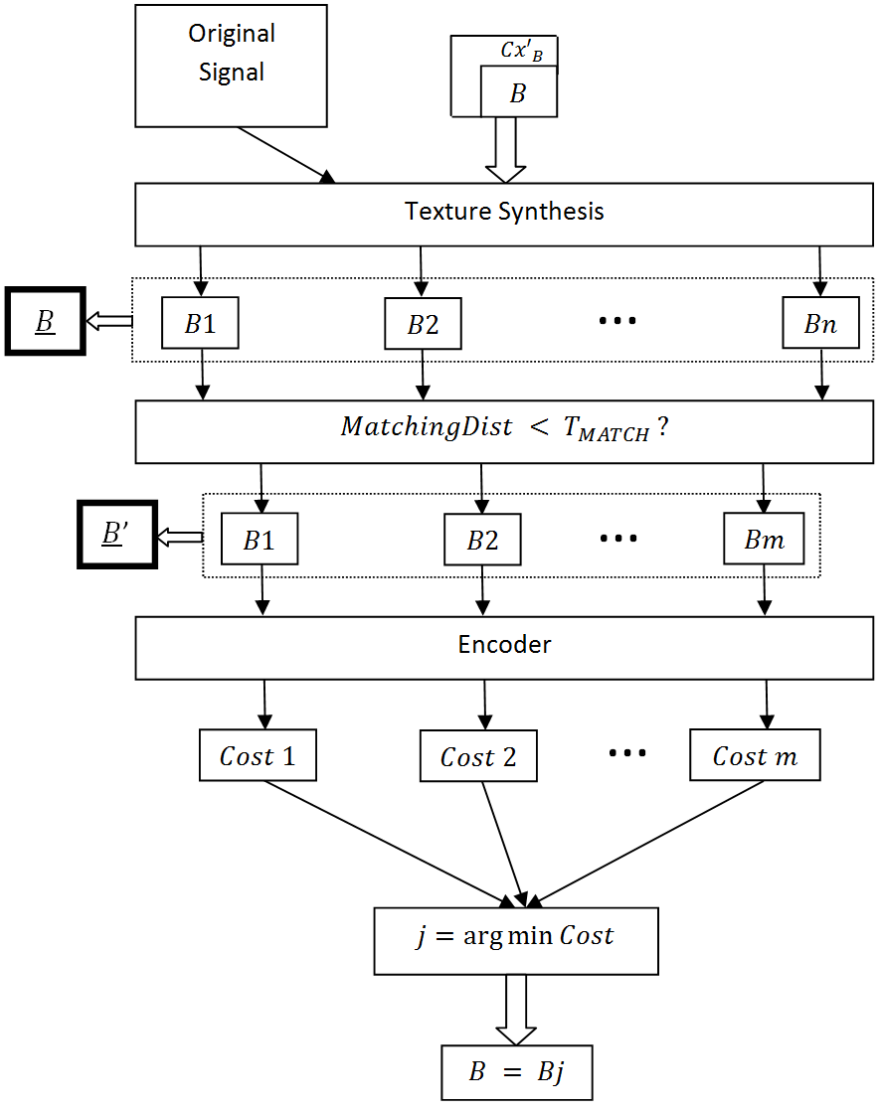


Figure 3.4 – Algorithmic overview of the local texture synthesis approach in [10].

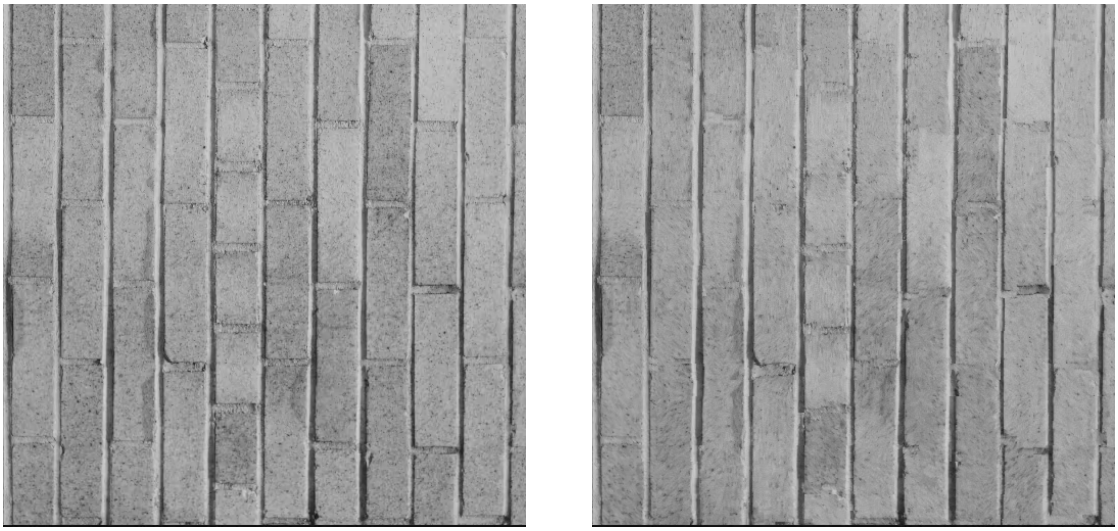


Figure 3.5 – Compressed texture with QP=27. Left: default encoder, right: LTS. Bitrate saving=9.756%.

Proposed Perceptual Model and its Performance Evaluation

4

V1-E: A Generalized Perceptual Model of Texture Similarity

In the previous chapters, an overview of texture perception and texture similarity models was established. It has been observed that there exist several models for texture similarity, that can be directly mimicking the neural processing of the visual system, or inspired by a certain properties of HVS. The first is known as bottom-up modeling; while the second is top-down one. The bottom-up modeling is generally more preferable as it does not require a hypothesis formulation about the visual system, but directly resembles its observed behavior.

It was also observed that, in contrast to static textures, dynamic texture perception is highly under explored. Most of the perceptual studies are based on texture images rather than texture video. This led to the fact the most perceptual texture similarity models are limited for static textures, and the dynamic texture ones are more computational than perceptual models.

In this chapter, we propose a generalized perceptual model of texture similarity that covers both static and dynamic textures. The model is inspired by the knowledge of neural processing in HVS, more specifically, the knowledge about the human visual cortex. The model can be directly used as a similarity metric, and also to extract a set of robust perceptual features.

The rest of the chapter is organized as follows: In section 4.1, the general introduction of the neural processing in the human visual system is provided. Section 4.2 gives the rationale behind the proposed model, and its details are presented in section 4.3. The use of the model as a feature extractor and similarity metric is given in section 4.4, with

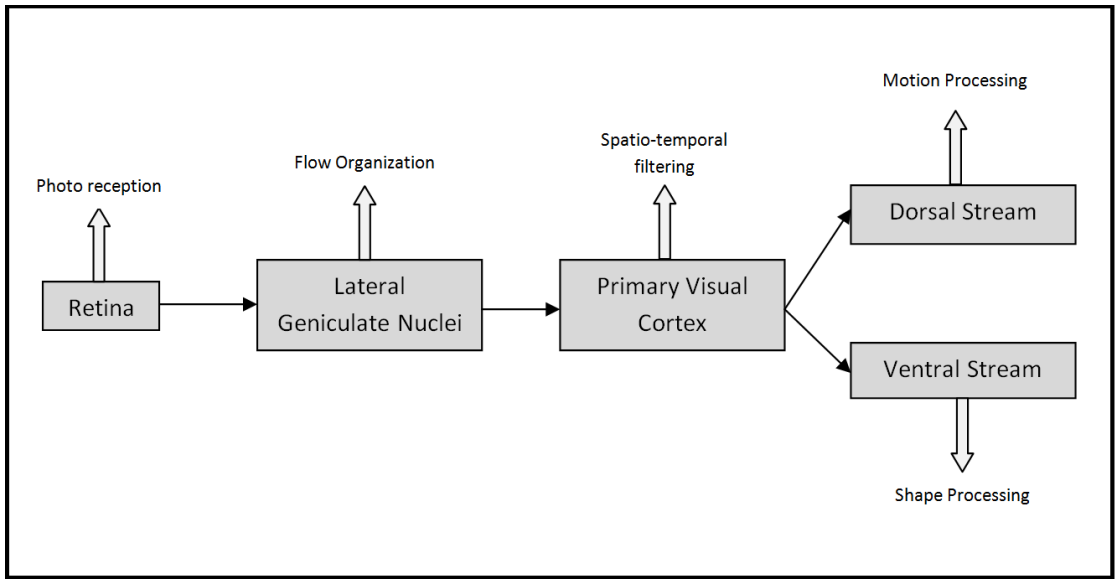


Figure 4.1 – Simplified block diagram of HVS.

a conclusion in section 4.5.

4.1 Introduction

Up to our knowledge, a perceptual model that governs both static and dynamic textures does not exist. The main issue is that although extensive perceptual studies on texture images exist, the texture videos have not been yet completely explored.

The aim here is to propose a perceptual texture model, based on the neural processing of the human visual system. For this purpose, a careful insight to the human visual system is necessary.

There are several stages of human visual system. In Fig. 4.1, a simplified architecture of the human visual system is provided. First, it starts with the human eye that contains the retina, the visual information captured in the retina travels through the visual pathway to the visual cortex, where the main tasks are motion and shape processing. Below, the summary of each component is provided. For details, the reader is referred to [137, 138, 139].

1. **Retina**

   The retina is a spherically shaped layer of the human eye. It is responsible for converting the incoming light into neural signal, to be transported by the optical nerves to the higher layers of visual perception. The retina contains two types of receptors, known as rods and cones, which are responsible for low light

level (scotopic) and high level light (photopic) vision. In the retina, there is a specific spot where the highest concentration of photo receptors exists. This spot is named the fovea. It is the one that extracts the visual information in the area towards which the viewer is focusing, and the corresponding mechanism is called foveal vision. Similarly, there is also the peripheral vision, where much less photo receptors exist, and thus fewer details are perceived.

In terms of signal processing, the retina processing can be considered as a passive processing. This is because it is, more or less, a sampling unit. Although there exists some spatial and temporal filtering operations, but it is negligible when compared to the higher level processing in the visual cortex.

2. **Lateral Geniculate Nucleus**

   After converting the incoming light to a neural signal, the visual information is transported through the optic nerves to further processing units in HVS. The next processing unit is the Lateral Geniculate Nucleus (LGN). LGN acts mainly as an information relay, in the sense that it transmits the information back and forth between the retina and visual cortex. The forward stream, from the retina to the visual cortex, is for analyzing the scene, such as extracting the motion and shape. On the other hand, the backward stream carries the brain commands for view localization, i.e. for re-centering the fovea towards the desired object or part of the scene.

   Similar to the retina, the signal processing part is negligible at this unit. Thus, it is also considered as a passive unit in the neural model.

3. **Primary Visual Cortex**

   The primary visual cortex, denoted as V1, is the main unit where neural processing is performed. V1 has a large set of receptive fields that are tuned to different spatial orientations, spatial frequencies, temporal flickering frequencies and motion directions. For this reason, it is always modeled as a set of spatio-temporal filters. This model is also supported by the evidence from the cortical response recordings carried out by Hubel and Wiesel [47]. It was observed that the spatial response resembles to some extent the Gabor filtering. Hubel and Wiesel distinguished two types of cells: simple and complex cells, where simple cells are linear, and complex cells are not linear.

4. **Higher Order Cortical Processing**

   Beyond V1, further cortical processing is performed in the higher level areas of the visual cortex, known as extrastriate cortex. Referring back to Fig. 2.8, one can distinguish two analysis chains. First, the chain of motion processing, called dorsal stream, is responsible for analyzing the output of the V1 motion detectors (section 2.4) and extracting the actual object motion by integrating the individual motions. It is able also to re-direct the fovea, via the feedback channel

in LGN, to track the object of interest. Second, the ventral stream analyzes the edges extracted by V1, and integrated them to reconstruct the shapes existing in the scene.

4.2 Primary Visual Cortex Inspired Model

Textures, both static and dynamic, are neither defined by a specific shape, nor structured motion. The reason is that textures are stochastic visual phenomena, and they are perceived as an abstract part of the scene, where the details are mostly irrelevant.

In addition, we do not assume eye pursuit for texture perception. This is because it is assumed that we are not following an object, or following a motion path for texture contents. Accordingly, the proposed texture perceptual model is not concerned with shape and motion analysis.

By looking at the block diagram of HVS in Fig. 4.1, we can plausibly assume that texture perception is mainly happening before the dorsal and ventral streams. In other words, we are disregarding the processing that occurs for motion analysis (dorsal stream) and also the processing for shape analysis (ventral stream). Accordingly, the model is up to V1 processing. In addition, since the neural processing in the previous units (Retina and LGN) are considered as passive processing (section 4.1), only a proper modeling of the neural processing happening in V1 is adequate for this model.

The V1 inspired modeling for textures is extensively used in the literature. It is very similar to the back-pocket model described in section 2.3, in the sense that it mimics the neural processing in the visual cortex. It has been successfully mainly applied for *static* texture similarity task, but has not been well explored for the *dynamic* textures (section 2.6.1).

The main motivation behind this model is the outstanding performance of its static texture version. For example, the performance in texture synthesis of the well-known Portilla and Simoncelli approach [23], showed that the texture similarity is mainly based on subband statistics. The resulting synthesized textures look very similar to the original one, despite the point-wise differences. The model has been re-adjusted in order to design a texture similarity metric, called structure-texture similarity metric (STSIM) [70], which has shown an excellent performance in the context of similar texture retrieval.

Accordingly, the proposed model can be considered as an extension of the existing ones in the temporal domain, in order to cover both static and dynamic textures. In contrast to other models, the proposed model considers only the energy of the bandpass signals, resulting from cortical processing, as the features used for texture similarity, and thus named as V1-E. This means that the model does not consider the inter-band correlations, or higher order statistical features for measuring texture similarity. This is mainly to constrain the complexity of the model. As shown in the verification and validations tests (next chapters), such a simple model is adequate for its purpose.

4.3 V1-E: Details

V1-E is a perceptual texture model that is inspired by the cortical processing of HVS. It mimics the spatio-temporal filtering that happens in the primary visual cortex area (V1). In the following, the details of the filtering process are provided.

Spatially, the cortical processing was first studied by Hubel and Wiesel [47]. It has been observed that the simple cells perform a bandpass filtering that can be asymptotically modeled as Gabor filtering. However, other studies considered this type of filtering to be differences of offset of Gaussians [51, 140], or differences of offset differences of Gaussians [141]. Along with this, there has been a lot of progress in subband filtering that exhibits good trade-of between cortical mathematical processing. For example, the Gabor filtering approach proposed by Manjunath [65] can provide the best parameters subband frequencies and orientations setting for minimizing the cross-band interactions. On the other hand, there has been always a need for a reconstructable subband transform, i.e. from image domain to cortical domain and vice-versa, for many applications such as texture synthesis. For such purposes, the cortex transform [142] or the steerable pyramid filter [68] can be used.

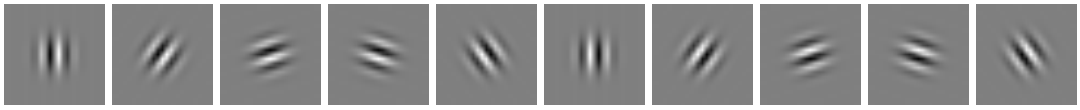
Temporally, the processing is slightly different, in the sense that there exist two types of processing. The first type of neurons, which are about 68% of the total neurons [143], perform low pass filtering, and the rest is bandpass filtering [144]. The low pass filtering is used for shape analysis, which is further analyzed in the ventral stream, while the bandpass filtering is used for motion extraction in the dorsal stream. However, some other studies do not consider this behavior, and simply expand the spatial filtering to the temporal domain. For example, both spatial and temporal V1 filtering are considered as derivative of Gaussian functions in [53], and direct Gabor filtering in [54]. The latter has shown a high correlation with the neural recordings, by learning the parameters of Gabor filtering.

On top of the linear filtering, there exists a non-linear mechanism. This is mainly due to two things. First, the neurons response to a signal is experienced in terms of neurons firing rate. The firing rate can only be a positive number, which mandates some rectification (full or half wave rectification). Beside this, the neurons responses are not fully linear. There exists a certain range where the saturation occurs, which results in the second non linearity.

Beside the simple cells, the complex cells are characterized by having a shift independent response. In other words, their response is relatively less sensitive to the stimuli position in comparison to the simple cells. This leads to the same modeling of the V1 simple cells, and eliminating the spatial phase dependency.

For V1-E, one of the recent modeling of V1 is used [60], which is named Feed-Foward V1-MT network (FFV1MT). This model was developed in order to mimic the human visual system mechanism of motion perception, by properly representing the neural processing in V1 and MT cells. The model has been tested in computing the

(a) Real (cosine) part.



(b) Imaginary (Sine) part.

Figure 4.2 – An example of the FFV1MT spatial filters' impulse responses of a fixed spatial frequency and 10 orientations.

optical flow and has shown its powerfulness in that task [62].

The V1 cortical processing in FFV1MT is represented by separable spatial and temporal filtering. This is done in order to reduce the overall complexity of computing the cortical subband signals. The spatial filtering is modeled as a series of Gaborian kernels, as given in this equation:

$$SF(x, y, \theta, f_s) = Be^{-(x^2+y^2)/2\sigma^2} e^{j2\pi(f_s cos(\theta)x + f_s sin(\theta)y)} \tag{4.1}$$

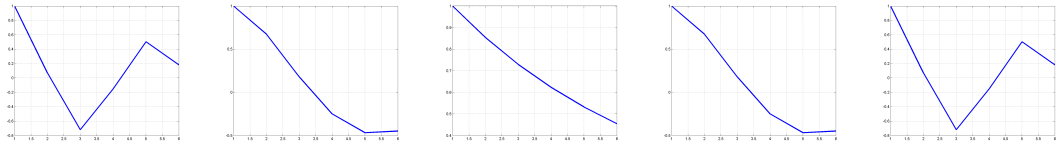
Where x and y are the spatial coordinate of the pixels, θ is the spatial orientation, f_s is the spatial frequency of the subband signal, σ is the support (width) of the Gaussian filter and B is a normalization factor. An example of the filters impulse responses is shown in Fig. 4.2, in which a fixed spatial frequency (f_s) and 10 orientations (θ) are used.

For taking into account different values of the spatial frequency, FFV1MT considers a pyramid based multi-scale approach. The original image is first smoothed by Gaussian filter, in order to avoid aliasing problem, and subsampled sequentially by a factor of 2. In a consequence, applying the same filters, as in Fig. 4.2, will lead to dyadic spatial frequency analysis. This method provides high design simplicity and computation efficiency.

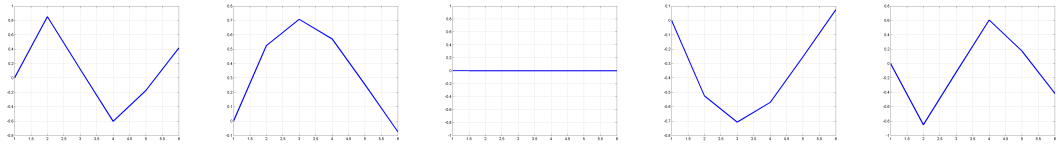
The temporal filtering approach in FFV1MT is modeled as bandpass filter, with an exponential decay. The impulse response is given in this equation:

$$TF(t, f_t) = e^{(-t/\tau)} e^{j2\pi(f_t t)} \tag{4.2}$$

Where f_t is the temporal frequency,τ is a constant which defines the temporal extent. The ratio of f_t over f_s is considered as the neuron preferred velocity magnitude, i.e. the velocity for which the neuron exhibits the highest firing rate. According to this, the number of temporal filters needs to be high enough to achieve high resolution motion analysis.



(a) Real (cosine) part.



(b) Imaginary (sine) part.

Figure 4.3 – An example of the FFV1MT temporal filters' impulse responses for a fixed spatial frequency. Each subfigure shows the response for 3 velocities. From left to right: Negative to positive velocity direction. Middle: zeros veolcity. The figures shows that the real part is even symmetric while the imaginary part is odd symmetric. In each plot, the x-axis is the time and y-axis is the response value of the filter.

An example of the impulse response for 3 velocity values is shown in Fig. 4.3. It should be noted that the real part has an even symmetry, while the imaginary part has an odd symmetry. This is also the same for the spatial filtering, which can be mathematically verified from the governing equations (Eqn. 4.1 and Eqn. 4.2).

The spatio-temporal filtering can be achieved by cascading the spatial and temporal filters. First, the even and odd spatio-temporal filters can be expressed in this manner:

$$STF_E(x, y, t, \theta, V) = SF_E(x, y, \theta, f_s)TF_E(t, f_t) - SF_O(x, y, \theta, f_s)TF_E(t, f_t) \quad (4.3)$$

$$STF_O(x, y, t, \theta, V) = SF_O(x, y, \theta, f_s)TF_E(t, f_t) + SF_E(x, y, \theta, f_s)TF_O(t, f_t) \quad (4.4)$$

Where the subscripts E and O denote the even and odd parts (resp.), and V is the velocity under consideration.

The complex cells, as described earlier, are position independent when compared to the simple cells. To take this into account, the complex cells are assumed to compute the energy of the quadrature pairs (even and odd parts) resulting from the simple cells response ($Rs(x, y, t, \theta, V)$). Mathematically, this can be expressed as:

$$Rc(x, y, t, \theta, V) = Rs_E(x, y, t, \theta, V)^2 + Rs_O(x, y, t, \theta, V)^2 \quad (4.5)$$

where Rc is the complex cell response, and Rs_E and Rs_O are respectively the simple cells even and odd responses. Rs_E is obtained by convolving the input signal with the

even spatio-temporal filter (Eqn. 4.3). Like-wise, Rs_O is obtained from the odd filter (Eqn. 4.4).

This is followed by lateral inhibition, in which the amplitude of the response is normalized by the energy of the surrounding neurons. It should be noted that the non-linearity of V1 is implicitly modeled as full-wave rectification in Eqn. 4.5.

In V1-E model, the lateral inhibition is not considered. This is first to reduce the computations, and second, the goal is not to model the complex cells. It is well known that the complex cells are mostly used for motion estimation, and V1-E is not a motion perception model. However, the energy is an important feature that can capture the visual information in textures, as well be shown in the next chapters.

It should be noted that the model has spatial and temporal restrictions. Spatially, it does not consider peripheral processing in HVS. In other words, it considers only the foveal vision. The foveal vision is limited to a narrow viewing angle (1-4 degrees of visual angle). For this reason, the spatial filters have limited spatial extents. Temporally, the analysis window is considered to be very short. It is assumed that the motion integration in the visual cortex takes an average time of 200ms [145]. Thus, this time period is considered as the length of the analysis window in FFV1MT.

4.4 V1-E as Features Extractor and Similarity Metric

One of the aims behind the developed perceptual model is to be used as a features extractor. The features can be used to characterize textures, as well as to feed a general machine learning approach for different texture related applications.

The complete block diagram of the model is given in Fig. 4.4. The input visual signal is a short term foveated signal, that means a signal with 1 to 4 degrees of visual angles and 200ms temporal length. The signal is first processed by the spatial filters, defined by their spatial frequency (or scale) and spatial orientation. The resulting subband signals are further processed by the temporal filters, for extracting the velocity related components. The output quadrature pairs are used to extract the energies (Eqn. 4.5), which is then used as texture features. This model, as highlighted before, is a generalized model for both texture images and videos. However, for still images, the temporal filtering should be eliminated.

For measuring texture similarity, there are several ways to combine the extracted features to form a similarity measure. One could possibly use machine learning tools, such as linear regression, to pool the features and to yield a similarity score. It is generally true that the more complex the tools are, such as support vector machine or artificial neural networks, the better the performance is. However, the intention here is to design a metric, that is independent of the dataset used for training and validation. Beside this, the metric is preferred to be bounded, such that when its score is maximum, perfect similarity is achieved, and the minimum metric score correspond to no similarity.

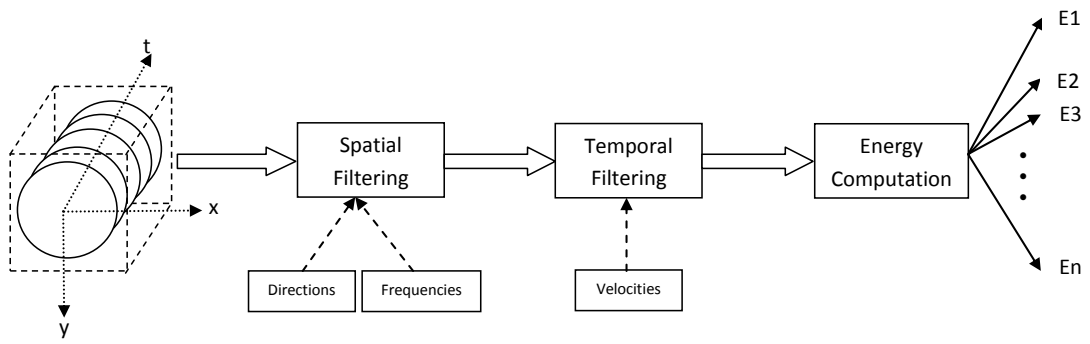


Figure 4.4 – Block diagram of V1-E features extractions (E1,E2,...,En).

In order to achieve this, we consider the following mathematical expression for measuring the similarity between 2 features (Ei_{T1} and Ei_{T2}) corresponding to the i^{th} energy of two textures under consideration ($T1$ and $T2$)

$$Sim(Ei_{T1}, Ei_{T2}) = \frac{2Ei_{T1}Ei_{T2}}{Ei_{T1}^2 + Ei_{T2}^2 + \epsilon} \tag{4.6}$$

This expression is the same that is used in STSIM [69]. It is bounded between 0 and 1, in which the score of 1 refers to the perfect similarity. ϵ is small value introduced here to avoid division by zero. The overall similarity is considered as the average similarity across all the energy components, as given in this equation.

$$SIM(T1, T2) = \frac{1}{n}\sum_{i=1}^{n} Sim(Ei_{T1}, Ei_{T2}) \tag{4.7}$$

4.5 Conclusion

In this chapter, a generalized model for texture similarity is proposed. The model covers both static and dynamic textures, and it can be considered as an extension of transform based similarity models, described in section 2.6.1, with more in detailed modeling of the human visual system.

There are several stages of visual processing in the human visual system. Among them, the cortical processing stage is considered as an active processing stage in this model. This is because other levels show a very marginal effect when compared to the cortical area. However, the cortical processing itself is complex one, with many cascaded units. The proposed model relies on the argument that textures are neither defined by shapes nor structured motion, which leads to the exclusion of higher level

of analysis associated with shape perception (ventral stream) and motion perception (dorsal stream), and thus it only considers the initial processing unit, named as V1.

The neural processing in V1 can be generally modeled as bank of spatio-temporal band-pass filters. The exact parameters of these filters are not fully known. Thus, we relied on a recently introduced computational model in [60].

The proposed model considers only the energies, thus named V1-E, of sub-band signals as the features for texture similarity,. This is because they are actually assumed to be computed by the V1 complex cells in order to provide a shift invariant response. The computed energies are then fused to produce a similarity score for each subband, where the average is finally taken to yield the overall similarity.

It should be noted that the model is based on cortical processing. This makes it limited to the spatio-temporal support of this type of processing. Spatially, it is limited to the foveal vision, which is a small part of the visual scene with up to 4 degrees of the visual angel. Temporally, it is limited to very short time, which is arrount 200ms. Another limitation is that it is neglecting any eye movement. Thus, it is considering the textures to be localized short term homogeneous signal, which are not tracked by the human eye. In the next chapter (Chapter 5), we show that this model performs as perfect similarity metric when dealing with such type of signals, and it also shows an excellent performance when extending it to larger extents. Besides; it can provide a powerful set of perceptual features for predicting texture properties associated with the perceived distortions due to video compression (Chapter 6).

5

Performance Evaluation as a Similarity Metric

After having introduced the proposed perceptual texture similarity model in Chapter 4, it is necessary to check its performance and compare it to the state of the art models. As already highlighted in section 2.7, there are broadly two mechanisms of testing: subjective and objective. Due to the difficulties associated with the subjective testing, and also the lack of the benchmarking results, this chapter focuses on the objective testing. For this, two test scenarios are defined. The first is a retrieval test, which is about finding identical textures in a dataset, given a query sample. The second is a recognition test, which targets similar textures that belong to the same category of the query sample.

5.1 Introduction

V1-E, the developed perceptual texture similarity model, is a generalized model that covers both static and dynamic textures. It is inspired by the cortical processing happening in the primary visual cortex (V1). In order to evaluate its performance, the model is first tested by considering its design constraints. In other words, it is *verified* for the same conditions that it was designed for. This type of testing is considered as a *verification* test in contrast to the *validation* test, which is about assessing the validity of the model for other testing scenarios.

As explained in section 4.2, V1-E is developed with the following assumptions:

1. It is meant for textural contents: homogeneous signals with relatively irrelevant details.
2. It is a foveated model. Spatial extents are less than 4 degrees of the visual angle.
3. It considers a short term cortical response, about 200ms.
4. No eye pursuit is assumed, which mandates eliminating any motion trajectory.

Accordingly, to examine its performance, a proper dataset of videos that complies with the above constraints needs to be designed. For this purpose, HomoTex dynamic textures dataset is designed in this work, which is fully explained in section 5.2.1.

There are many approaches that can be employed for the verification testing, but the goal here is to test the performance in terms of measuring texture similarity. This could either be performed subjectively or objectively. Subjectively, this is usually performed through psycho-physical experiments, by measuring the perceived texture similarity according to visual scores from observers. Objectively, it can be performed by designing a dataset with similar textures groups, and checking the metric response in this dataset. The advantage and disadvantage of each methodology was discussed in section 2.7.

At this level, the objective evaluation is considered. This is to have a large scale evaluation of the proposed model in terms of measuring similarity. We designed a specific test in order to examine the performance of the V1-E as a texture similarity metric, within the constraints of the model itself. For this, identical texture retrieval task was performed, considering the complaint textures dataset of HomoTex. Details are given in section 5.2.

Beyond this, the model is tested with a more general framework that does not consider the model constraints. This is known as validation test, in contrast to the previous verification test. In fact, the evaluation tests for dynamic textures are pretty much well defined. There exists a set of common test conditions, which are used to compare the performance of different approaches. A couple of datasets are designed for this purpose.

The first dataset that is used for this purpose is known as UCLA dataset [14, 146], where UCLA stands for University of California/Los Angeles. However, this dataset contains few number of texture videos. For this reason, DynTex dataset [5] has been created to be a comprehensive dataset, involving large number of textures, exhibiting also some camera panning. For the task of recognition, a subset of DynTex is used to generate a new dataset, known as DynTex++ [112], in which each video is labeled by its corresponding class.

In order to perform the recognition task, an extension of the proposed model from a short term dimension to a larger one is needed. A discussion about different extension possibilities is discussed in section 5.3.3.

The rest of the chapter is organized as follows: Section 5.2 provides the details of the retrieval test, the used dataset and experimental results. Section 5.3 is concerned about the recognition test, with the details about the datasets of DynTex++ and UCLA. The general discussion and conclusion is then provided in section 5.4.

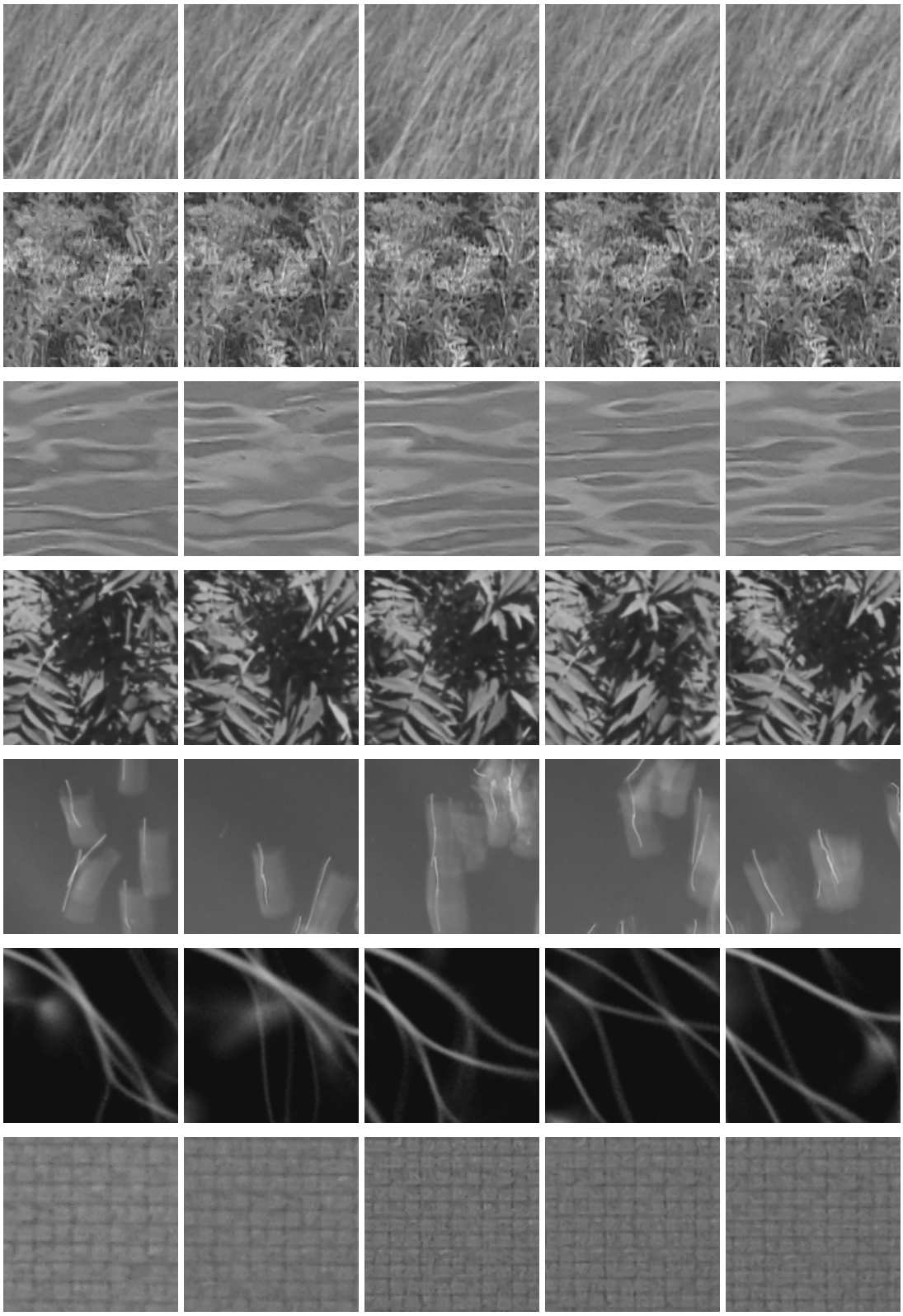


Figure 5.1 – Examples of texture classes used in the verification test. Each row shows the first image of 5 texture videos, out of 50, belonging to the same class.

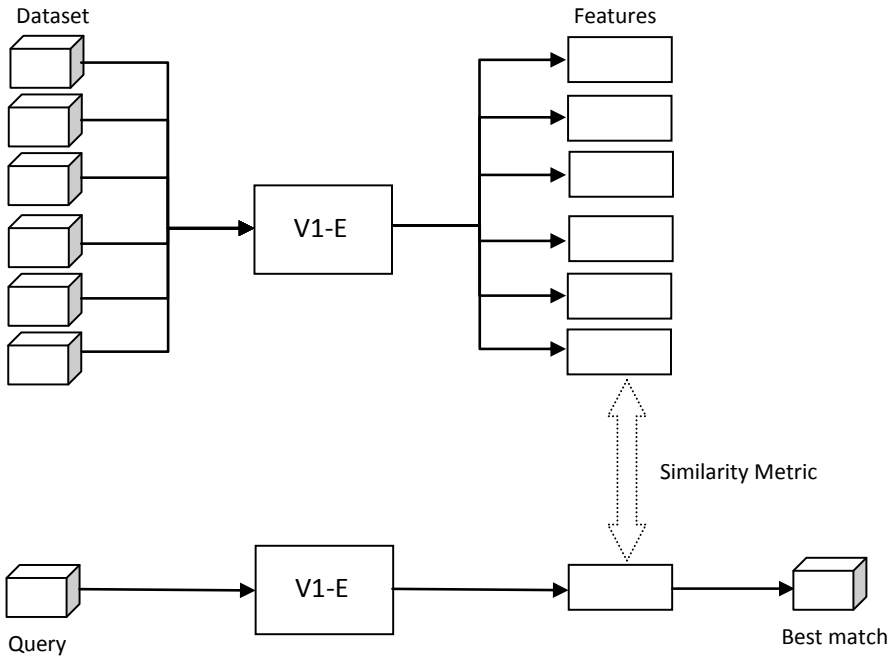


Figure 5.2 – Architecture of the retrieval system, inspired by [67].

5.2 Texture Retrieval Test

The general architecture of the retrieval test is shown in Fig. 5.2. The dataset consists of a large number of textures, for which the V1-E is used to extract the textural features represented by V1 energies. A query texture, i.e. the one that we are interested in extracting its identical samples from the dataset, is also analyzed by V1-E model to extract its feature vector. The V1-E similarity metric is then computed between this feature vector and all the vectors from the dataset, and the sample which has the highest similarity is retrieved as the best match. As a performance measure, the retrieval rate is used, where it is defined as the ratio of number of the correctly retrieved textures, to the number of queries. In the following subsections, the details of the implementation and results of the retrieval system are provided.

5.2.1 HomoTex: Homogeneous Textures Dataset

As explained in section 5.1, it is required to design a dataset that meets the constraints of V1-E model. For this purpose, the new dataset has been manually designed from the other existing datasets.

For texture videos, there is a limited number of available datasets. One of the commonly used one in computer vision applications is the DynTex dataset [5]. This dataset contains a large number of videos (650), each with a resolution of 720x576 and 25 frames per second. This dataset is the most important one, as it covers almost all types of textures, captured both by static and moving cameras.

Besides this, a relatively new dataset of natural scenes, focusing on textural contents is developed in 2015 [147], named as BVI textures. This dataset contains high quality videos, subjectively annotated with perceived distortion due to video compression.

The two datasets were used to generate the dataset used in this work (HomoTex). First, some source videos from the two datasets were manually cropped to cover only homogeneous areas of textures. Temporally, only contents that exhibit some inconsistency over time are eliminated. Overall 47 texture videos, meeting the constraints, were collected from the two datasets. For further information, the complete dataset is shown in the appendix (Fig. A.1).

It is assumed that the textures are viewed at standard viewing distance. Within this distance, 1 degree of visual angle roughly corresponds to 64x64 pixels. Since the dataset is designed to be within the foveal vision, the spatial extent was fixed at 128x128, which is a good trade-off between foveal vision, spatial homogeneity and video coding consideration as will be seen in Chapter 8. The 200ms temporal window corresponds to 5 frames for DynTex, and 30 frames for BVI.

The HomoTex dataset is further reduced by removing videos containing least dynamics. This is done in order to focus on more difficult contents, with lots of variations over time. Then, we only considered 38 out of 47 video, and neglected that last 9 textures shown in Fig. A.1. Nevertheless, the complete texture dataset was used for other tests in this work, namely for distortion sensitivity estimation (Chapter 6) and testing the proposed perceptual rate-distortion estimation framework (Chapter 8).

5.2.2 Experiment Details and Results

With the resulting 38 texture videos from HomoTex, a special objective test is designed in order to verify the performance of V1-E as a texture similarity metric. The main idea is to generate a ground truth data, in which similar (and dis-similar) textures are labeled. To do so, it is hypothesized that textures look similar regardless of the capturing/viewing period. For example, looking at the sea wave would result in the same perception at a given time instant or some instant later or before. This is because the same physical phenomenon is present, and the difference in the details is negligible.

Accordingly, all the 38 videos, which are of 10 second length, are split into non-overlapping chunks of 200ms. The resulting chunks from the same source videos are labeled as similar and others as dissimilar. By doing so, the evaluation procedure is a classification problem. Overall, the dataset for evaluation contains 1800 texture videos, distributed in 38 classes, and each class has 50 class members.

It should be noted that this type of testing is also carried out for testing and comparing the performance of the state of the art image texture similarity metrics, such as STSIM and LRI [86]. However, it is considered as a retrieval task rather than a classification task. For texture videos, the similar procedure is used in [148] to test the performance of the local binary pattern extension for texture videos. However, in both cases, the spatial and temporal dimensions are randomly selected, without considering the visual perception part.

To give an idea about the resulting videos, Fig. 5.1 shows some examples of the resulting texture classes. Within each class, one can easily notice the point-wise differences due to the difference in sampling time. The evolution over time, not possible to be shown in the figure, is also significantly different by point-wise visual examination. However, the overall similarity of each class contents is much higher when compared to the other classes.

In order to assess the performance of V1-E, the classification test is performed as follows. For each class sample, the similarity score is computed with all the other textures from the dataset. If the maximum similarity is achieved with respect to any sample of the same class, the sample is considered to be correctly classified. The classification rate is considered as the tool for performance assessment, which is mathematically computed as the ratio of the number of correctly classified samples to the total number of samples in the given class. Thus, it is the same as the retrieval rate that is defined before (section 5.2). For this reason, the results are reported in terms of retrieval rate rather than classification rate.

To avoid the over-fitting problem, the dataset is randomly divided into training and testing sets of equal sizes, i.e. 50% of the data for training and 50% for testing. In this way, the similarity scores for each testing sample are computed against the training samples, and the classification rate is computed as before. The experiment is repeated 20 times, in order to reduce the possible bias due to the random division.

The performance is compared to the well-known similarity measure of Local Binary Pattern (LBP) for dynamic textures, which is known as Local Binary Pattern - Three Orthogonal Planes (LBP-TOP) [85]. This is because this metric has shown an excellent performance for texture recognition and is generally considered as the benchmark for other metrics. Secondly, LBP-TOP implementation, in contrast to other texture similarity approaches, is available from the authors website. This made it easy to have the reference implementation for fair comparison.

For V1-E, three parameters sets need to be specified (Fig. 4.4). First, the selected

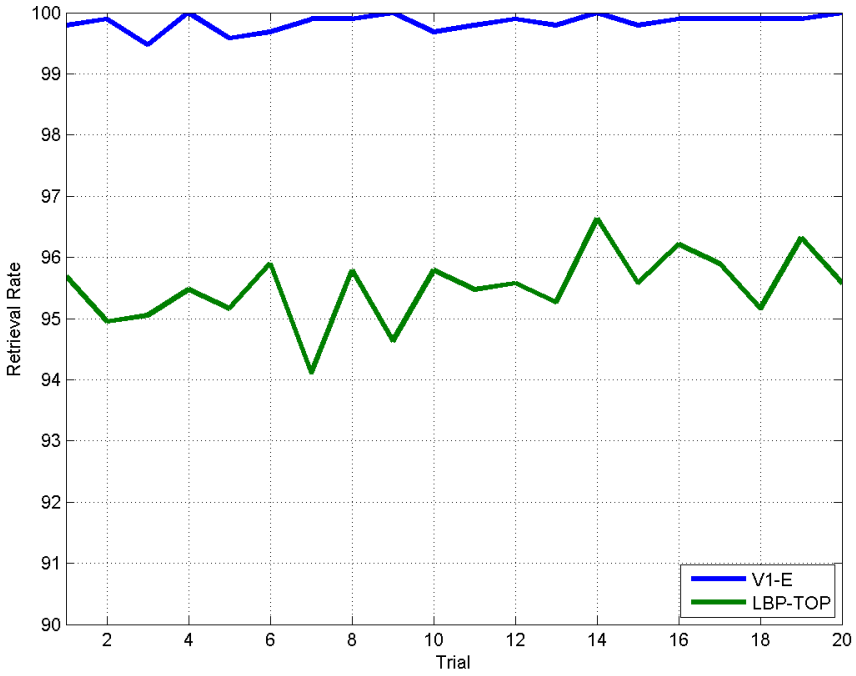


Figure 5.3 – Performance evaluation in terms of retrieval rate for both V1-E and LBP-TOP. The experiment is repeated with 20 trials. Clearly, V1-E outperforms LBP-TOP for all the trials.

number of orientations is 8, which means that for each spatial frequency, 8 equally spaced spatially oriented bandpass filters are used. Second, 5 spatial frequencies were used. As explained in section 4.3, the frequencies are considered as dyadic. Then, the number of frequencies corresponds to the number of scales. Third, 7 velocity values are considered, that contains positive, negative and zeros velocities. Overall, the number of energy components used inside similarity metric is 280 (8x7x5). For LBP-TOP, the default implementation considers a joint histogram XY,XT and YT channels with the length of 178.

The results of the performance evaluation test are shown in Fig. 5.3. For each trial, the V1-E shows an outstanding retrieval rate. In many trials, it can reach the 100% limit. This is significantly better than LBP-TOP. On average, the rate is 99.84% for V1-E and 95.51% for LBP-TOP. This shows that V1-E improves the rate by more than 4% (on average) when compared to LBP-TOP.

The results clearly indicate that the proposed model can well express the texture similarity, when the data fits the model constraints. It can then be concluded that it is faithfully achieving its design goal.

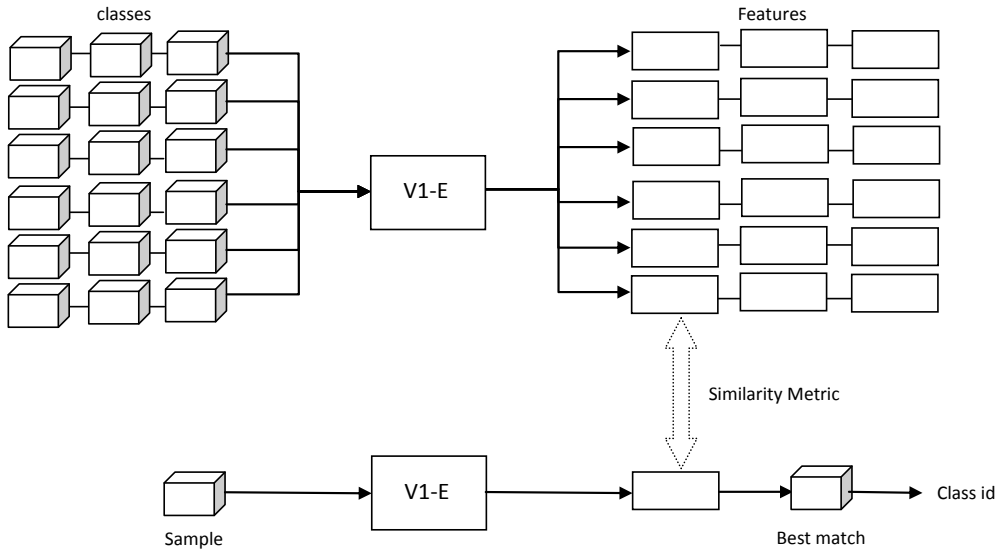


Figure 5.4 – Architecture of the recognition system.

5.3 Texture Recognition Test

The recognition test is very similar to the retrieval test. However, in recognition the task is rather about recognizing the category of the query texture. In other words, the task is about finding which class the query texture belongs to. It should be noted that the words recognition and classification are interchangeably used by different studies for the same task that we are tackling here.

The general architecture is given in Fig. 5.4. Similar to the retrieval test, V1-E is used to compute the feature vectors of the all video in the dataset. The similarity metric is used to find the most similar sample in the dataset to the query texture. If the class of this texture is the same as the query class, it is considered as correctly classified. For performance measure the classification rate, which is very similar to the retrieval rate, is used. In the following subsections, the test is fully explained and the benchmarking results are provided.

5.3.1 UCLA Dataset

UCLA dataset has been designed during the pioneer work of Doretto et al. on dynamic texture analysis. The work has shown significant progress in texture recognition, synthesis and editing [29, 14, 79], based on the auto-regressive moving average model

(section 2.6.2). The dataset contains 50 classes of textures, such as boiling water, candles, flowers, plants and others. Each class contains 4 samples, thus, the overall number of textures is 200. Each video is of 160x110 spatial resolution, with 75 frames and 15 frames per second.

Currently, the UCLA dataset version by Chan et al. [146] is used. This version contains cropped videos from the original UCLA videos. The cropped videos contain only the representative texture part, to eliminate/reduce the background inhomogeneity problem. The resulting textures are shown in Fig. 5.5.

There exist two procedures for objective evaluation with this dataset. First is the leave-one-out cross validation test. In this test, for each class of 4 textures, 3 are used for training and the 4^{th} texture for testing. The process is repeated for some trials and the mean classification rate is used as an evaluation criteria. The second procedure is the four-folds cross validation test. This procedure is proposed in [112], and commonly used for benchmarking of similarity models.

5.3.2 DynTex++

To overcome the limitations of UCLA dataset, that present in terms of low resolution and few number of texture videos. DynTex was developed to cover wide range of dynamic textures, and to have multiple captures of the same phenomena, allowing different camera pans. This led to generate much higher number of videos, which are 650. Each video is at least 10 seconds long. The spatial resolution is 720x576, with 25 frames per second.

Although there is a classification benchmark for DynTex, for example Alpha dataset with three classes of sea, grass and trees, the native videos are not directly used for recognition task. Usually, homogeneous videos are segmented, both spatially and temporally, into non-overlapping patches. The patches belonging to the same source videos are labeled as the same class. This procedure is first presented in [85], which is very similar to the one that is followed in the verification test in section 5.3.

A more systematic classification benchmark has been introduced by Ghanem et al. in [112], in which a subset of DynTex was used to generate a large scale classification dataset, named DynTex++. The main reasons made UCLA more suitable for classification task, when compared to DynTex, is highlighted by Ghanem as follows:

1. *Its DT sequences have already been pre-processed from their raw form, whereby each sequence is cropped to show its representative dynamics in absence of any static or dynamic background.*
2. *Only a single DT is present in each DT sequence.*
3. *In each DT sequence, no panning or zooming is performed.*
4. *Ground truth labels of the DT sequences are provided.*

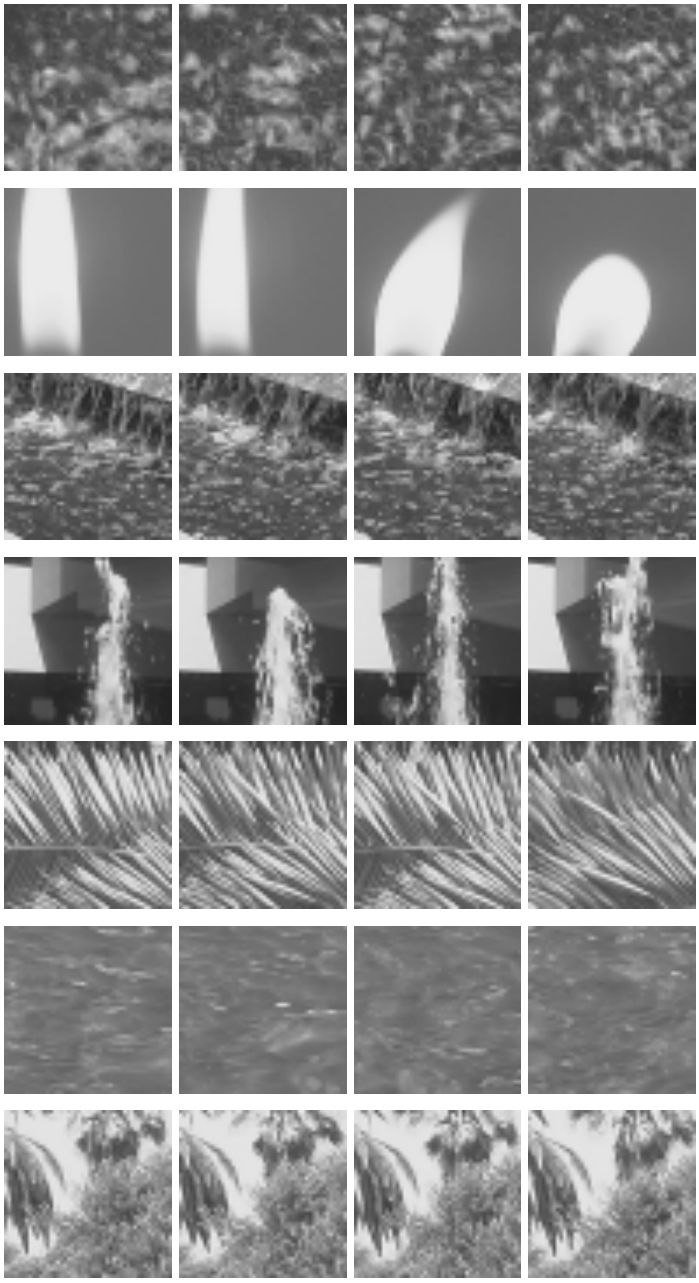


Figure 5.5 – Examples of some classes of UCLA dataset. Each row shows the first image of texture videos belonging to the same class.

Accordingly, the aim of DynTex++ is to organize the texture videos available in DynTex, to provide richer benchmark than UCLA. DynTex++ contains 3600 texture videos, equally organized into 36 classes, with 100 samples per class. Each video is 50x50 pixels, with 50 frames. The videos are manually selected in the way that they do not possess dynamic background, do not have more than one dynamic texture and do not have any camera pan or zoom.

To have an idea about the different classes, Fig. 5.6 shows some examples of the texture videos from the dataset. It could be noticed that this dataset is more challenging than UCLA. First, it has larger number of samples per class. This means a lot of deviations within one class. Second, it contains some videos that are defined by their shapes and motions, which can hardly be considered as dynamic textures. For example, the traffic videos shown in the figure (second row) have distinguishable shapes in each sample. This is certainly not complying with the design principle of the proposed perceptual texture model V1-E. In addition, the smoke videos in the fifth row are not homogeneous; one can clearly distinguish the background from the smoke.

The common evaluation procedure on DynTex++ is a classification task. In this task, 50% of the data is used for training, and 50% for testing. This assignment to training and testing is random. It is repeated 20 times, and the average classification rate is reported for benchmarking.

5.3.3 Extending V1-E for General Sequences

V1-E, as explained in details in section 4.3, is designed for a foveated signal that has a short temporal window (about 200ms). In the verification test (section 5.2), the dataset for testing considered these constraints. However, the two datasets considered in the validation test, namely UCLA and DynTex++, do not have this restriction. Spatially, both datasets do not have large extent, so it can possibly be assumed they are within the foveal vision. This is not true in the temporal domain. In UCLA, each sample video is 5 seconds long, where as DynTex++ is 2 seconds. This poses the question on the way that V1-E can handle videos longer than its temporal window.

The approach that is considered in this work is depicted in Fig. 5.7. For each 200ms, V1-E is used to extract the features, which are the energies of the cortical response (section 4.4). The features are pooled in the temporal domain, by employing 4 statistical measures. The statistical measures considered are the mean, standard deviation, skewness and kurtosis. This results in 4 groups of features, for each of which the same similarity metric of native V1-E is used (section 4.4). Let us denote the similarity due to mean energy as Sim_M, due to the standard deviation of the energy as Sim_S, due to the skewness as Sim_{Sk} and due to the kurtosis as Sim_{Kr}. Then, the overall similarity is computed in this way:

Figure 5.6 – Examples of some classes of DynTex++ dataset. Each row shows the first image of texture videos belonging to the same class.

Model version	α	β	γ	δ
V1-E-1	1	0	0	0
V1-E-2	2	1	0	0
V1-E-3	4	2	1	0
V1-E-4	8	4	2	1

Table 5.1 – Different parameters settings.

Method	Year	Classification Rate (%)
AR-LDS [111]	2001	89.9
Spacetime orientation structure [34]	2012	81.0
Low dimensional LBP based scheme[149]	2016	95.0
MEWLSP [4]	2016	96.5
V1-E-1	-	**97.50**
V1-E-2	-	**98.00**
V1-E-3	-	**98.00**
V1-E-4	-	**97.00**

Table 5.2 – Averaged classification rate on UCLA dataset, using leave-one-out cross validation scheme, of different dynamic texture recognition approaches. LBP-TOP and VLBP results are copied from [4].

$$Sim = \frac{\alpha Sim_M + \beta Sim_S + \gamma Sim_{Sk} + \delta Sim_{Kr}}{\alpha + \beta + \gamma + \delta} \tag{5.1}$$

Several values of the weights (α,β,γ,δ) are tested, which are shown in Table 5.1. The rationale behind the numbers in the table is that it is assumed that lower order statistical measures are more important than the higher ones. This means that the mean value is more important than the standard deviation. Thus, the weights are selected in the way that it becomes half when the statistical order is doubled. The first version of V1-E similarity metric in the table tests the influence of the mean only; the second tests the added value due to the standard deviation and so on.

5.3.4 Benchmarking Results

In this section, the classification rate of the four versions of V1-E extensions (Table 5.1) is reported, following the same testing procedure defined for both UCLA (section 5.3.1) and DynTex++ (section 5.3.2). For the purpose of performance comparison to other approaches, many results were collected from literature. It should be noted that the naming terminology is the same as in [4].

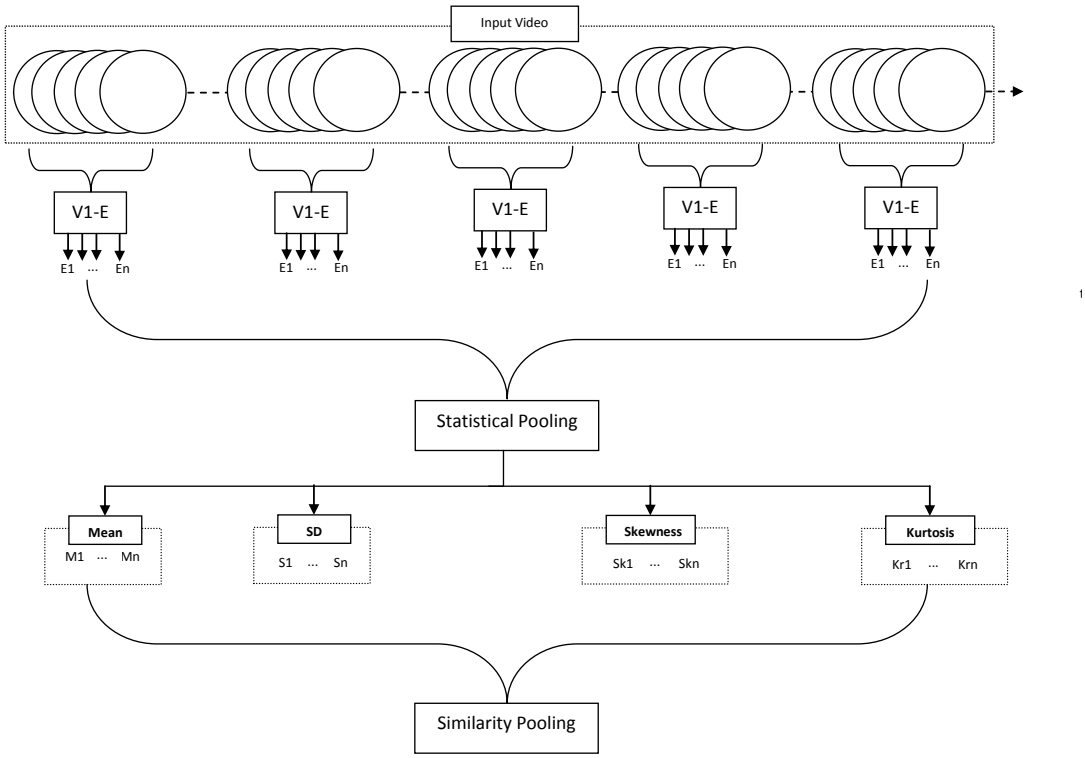


Figure 5.7 – Extended V1-E model.

Method	Year	Classification Rate (%)
VLBP [85]	2007	89.9
KDT-MD [150]	2007	89.5
LBP-TOP[85]	2007	94.5
DFS [99]	2011	89.5
3D-OTF [101]	2012	87.1
CVLBP [3]	2016	93.0
Low dimensional LBP based scheme [149]	2016	95.0
MEWLSP [4]	2016	96.5
V1-E-1	-	**97.50**
V1-E-2	-	**98.50**
V1-E-3	-	**98.50**
V1-E-4	-	**97.00**

Table 5.3 – Averaged classification rate on UCLA dataset, using four folds cross validation scheme, of different dynamic texture recognition approaches. LBP-TOP and VLBP results are copied from [4].

Method	Year	Classification Rate (%)
VLBP [85]	2007	94.98
LBP-TOP [85]	2007	94.05
DL-PEGASOS [112]	2010	63.7
DFS [99]	2011	89.9
3D-OTF [101]	2012	89.17
PCA-cLBP/PI-LBP/PD-LBP [151]	2013	91.9
DDLBP with MJMI [152]	2014	95.8
NLSSA [153]	2014	92.4
KSSA [153]	2014	92.2
DKSSA [153]	2014	91.1
Chaotic vector [154]	2016	69
MBSIF [155]	2014	97.17
High level feature [156]	2015	64.22
DNGP [157]	2015	90.2
WLBPC [113]	2016	95.01
Low dimensional LBP based scheme [149]	2016	96.28
MEWLSP [4]	2016	98.48
V1-E-1	-	**90.88**
V1-E-2	-	**91.79**
V1-E-3	-	**92.14**
V1-E-4	-	**92.11**

Table 5.4 – Averaged classification rate on DynTex++ dataset, using 50% of the data for training and 50% for testing, of different dynamic texture recognition approaches. LBP-TOP and VLBP results are copied from [4].

First, the results for the leave-one-out cross validation scheme for the UCLA dataset (Table 5.2) indicates clearly that the proposed method outperforms the state of the art methods. This is true even for the simplest version of V1-E, which includes only the mean of energies (V1-E-1). Increasing the statistical features (standard deviation, skewness and kurtosis) can generally improve the performance. However, the kurtosis has an adverse effect. It could be due to the pooling parameters (Table 5.1), and better combination can be found to achieve better results.

The other test of 4-fold cross validation scheme shows similar performance as the leave-one-out cross validation. The results shown in Table 5.3, indicates clearly that the classification rate is the highest when compared to the others. The same conclusion can be drawn regarding the higher order statistical pooling features.

Second, the results with DynTex++ are slightly different. Although V1-E provides an excellent classification rate (Table 5.4), it is below the best performance. Interestingly, the local binary patterns of VLBP and LBP-TOP, which were developed in 2008, outperforms many other models that are up-to 2015. However, in the verification test (section 5.2), LBP-TOP could not compete with V1-E. In comparison to the results on UCLA dataset, both LBP-TOP and VLBP are way beyond the other models. This is partly highlighted in the introduction of the datasets (section 5.3.1 and section 5.3.1). DynTex++ contains many videos that are defined by their shapes and motions (Fig. 5.6). This is certainly against the purpose of V1-E, which analyzes the textures on the assumption that they are treated as adequately shapeless phenomena, without structured motion (section 4.2).

5.4 Discussion and Conclusion

In this chapter, large scale evaluations of the proposed perceptual texture similarity model are performed. The evaluations included both verification and validation tests, in which the verification test is mainly concerned about testing the performance within the design constraint of the model, while the validation test is a generalization test in order to assess the performance with general dataset, that is not biased towards the model itself.

With the same concepts used for developing the V1-E model, a dataset of texture videos has been designed. This dataset, named HomoTex, consists of 38 texture videos, having homogeneous contents with a spatial resolution that is with then the foveal vision. Each video was further temporally split into 50 non-overlapping patches, each of 200ms temporal extent. As the resulting dataset is complaint with model constraint, a proper test was designed to assess the performance of the model in terms of similarity measure. The test was about the ability of this model to retrieve identical textures, given a query sample. In this context, identical textures are the ones that belong to the same original video (before splitting). It has been realized that the proposed model pro-

vides an outstanding performance, in which the correct classification rate can reach up to 100%, in a large dataset where 50% of it was used for training and validation. It also outperforms the well-known dynamic texture model of LBP-TOP.

The result obtained from the verification test is to some extent biased to the proposed model. The benchmarking with other models was not possible, since the dataset is not yet published, and the reference implementation of other models, apart from LBP, is missing.

For this reason, we were also interested into assessing its performance in a more general testing scenario, where the benchmarking data are available. We considered this as a validation test, in contrast to the verification test, that is not biased to the proposed model. For this purpose, the common recognition tasks are performed, in which 2 datasets are used (UCLA and DynTex++). This task necessitated the extension of V1-E from short term foveal resolution to a general one. Different statistical pooling methods have been discussed, such as considering the mean of V1-E features only, or involving higher order features like standard deviation, skewness and kurtosis.

The results indicate that the V1-E shows an excellent performance. It also outperforms the state of the art methods on UCLA dataset. However, for DynTex++ it is not the best. This is possibly due to the complexity of DynTex++, which highly conflicts with the design principle of V1-E.

6

Performance Evaluation as a Features Extractor

The proposed model (V1-E) has been tested as a similarity metric in the previous chapter. It showed an excellent performance in both retrieval and recognition tasks. However, V1-E is also a model that can extract textural features, that might be used for other purposes rather than texture similarity analysis. The objective here is to test the performance of these features in predicting visual properties associated with textures. Unlike the previous tests of retrieval and recognition, this time a psychophysical testing is involved to extract these properties.

As of its vital importance in this work, a link to video compression is brought here. The properties that we study here are related to the visual sensitivity to the perceived distortions due to video compression. The objective is to understand to what extent the perceptual texture features, obtained via V1-E, can be used to predict these visual properties.

The rest of the chapter is organized as follows: The general introduction about the context is given in section 6.1. Section 6.2 provides the details of the psychophysical test with its results. In section 6.3, the performance of V1-E in terms of predicting the perceptual properties, resulting from the psychophysical test, is examined, while the conclusion is given in section 6.4.

6.1 Introduction

V1-E is a perceptual texture model, which can be either used as a similarity metric, or features extractor (section 4.4). One of the intention of developing this model is to be used in the context of image/video compression. In this chapter, the first investigation about perceptual properties of textures involving compression is carried out.

The latest video compression standard is known as high efficiency video coding (HEVC) [114]. A short overview of it is given in section 3.2. It is basically a hybrid video coding that utilizes both signal prediction and transform in order to provide a compact representation of the video sequences. An entropy based binary coding (CABAC [114]) is used to achieve the minimum amount of information to be stored or transmitted over channels. These mechanisms (prediction, transform and entropy coding) rely on the statistical redundancies of the input signals, such as spatial and temporal correlation. However, beside this, there are also perceptual redundancies that can be further exploited to enhance the coding performance.

It is well known that the human visual system can detect differences when a certain perceptual threshold is crossed. The just noticeable difference/distortion (JND), is the threshold at which the change of certain physical quantity causes a perceptual difference. It is of huge importance in many applications involving perceptual optimization. An example of this, in the scope of this work, is permitting the coding system to further compress the input signal, while assuring an equivalent perceptual quality. In other words, the idea is to exploit the existing perceptual redundancies in the input signal.

Typically, JND threshold is estimated based on low level mechanisms of human vision, namely contrast sensitivity [158][159]. Such methods are of limited scope, and can poorly perform in the region of apparent distortion (suprathreshold region) as indicated in [160]. According to this, it is argued that the threshold can be properly estimated from the natural image sequences themselves, taking into account computational features describing them. However, our scope is limited here to textured signals.

The perceptual distortion sensitivity is a general term concerning how much sensitive the human visual system is to a particular distortion type. Two properties are studied here. First is concerned about the perceptual redundancies, which refer to the amount of redundant visual information that can be eliminated without causing visual difference. Second is the perceptual tolerance, which precisely tells how much distortion can be tolerated without changing the perceived quality.

V1-E, as a features extractor, is tested in order to understand whether it can predict these properties. This can be considered as an extra test, beside the verification and validation test (Chapter 5), to look into V1-E powerfulness and weakness.

6.2 Perceptual Distortion Sensitivity Estimation

6.2.1 Method and Apparatus

The perceptual distortion sensitivity, as discussed in section 6.1, is concerned about the amount of how much the visual system is sensitive to a particular distortion. There are many ways to study this, and several psycho-physical approaches can be employed for this purpose.

In this work, a particular interest in image and video compression is considered. It is thus important to study the perceptual distortion sensitivity in a manner that is directly applicable to the compression scenario. Thus, the goal here is to understand the sensitivity towards this type of distortion, and apply that knowledge to enhance the compression efficiency.

For an image sequence under consideration, which is compressed up to a certain ratio, an experiment is designed to quantify the amount of further compression that can be added, without altering the perceived quality. In other words, to quantify how much redundant visual information can be eliminated, without causing any perceived difference.

This type of study is well established in psycho-physics. It is related the task of threshold detection, or more generally psychometric estimation. The psychometric function (Ψ) is generally an "s" shaped function that relates the physical change to the detection/discrimination probability [161]. The general mathematical form of it is [162]:

$$\Psi(x, \alpha, \beta, \gamma, \lambda) = \gamma + (1 - \gamma - \lambda)F(x, \alpha, \beta) \tag{6.1}$$

F is an "s" shaped function that usually takes the range [0-1], for example Weibull, logistic and cumulative Gaussian. γ is the guess rate, and λ is the lapse rate. The other two parameters (α and β) represent the shape parameters of F, such that α corresponds to the shift in x and β correspond to the slope of F.

Estimating the psychometric function, or the subjective threshold, is a classical task in psychophysics. There are generally two types of methods, adaptive vs. non-adaptive. An example of the non adaptive methods is the method of limits. This method is used to measure a certain perceptual threshold by changing the value of a physical quantity until the perceptual threshold is detected. This can be in either ascending or descending (or both) manner. In the method of adjustment, the observer himself controls the stimuli level to determine the perceptual threshold. Another method, which is known as the method of constant stimuli, is used to measure the psychometric function at constant point. The psychometric function can be then interpolated by different functions using either least square or maximum likelihood estimation. The details of these methods, as well as others, are described in [163].

The adaptive methods, as compared to non-adaptive ones, are usually faster and more precise. There exists a large body of research dealing with this topic. One of the first, and the highly inspiring, method is known as Parameter Estimation by Sequential Testing (PEST) [164]. In this method, the stimuli level is tested, and the instantaneous probability is estimated. Using the Wald test, it can be determined whether the level is within the range of target probability. When it is not the case, the stimulus is level is changed within a constant step size. Once an inversion in the order happens, the step size is halved. The process continues until the minimum step size is achieved. Several extensions of the PEST test has been proposed, for example: more virulent PEST [165], ML-PEST[166], QUEST [167] and Zest [168]. For more details, the reader is referred to the reviews in [169] and [170].

The state of the art method is known as Updated Maximum Likelihood (UML) test [171, 172]. It defines three search ranges of the psychometric function (λ, β and γ) as in Eqn. 6.1. Based on the observer response, it measures the log likelihood value of all of the possible psychometric functions in the search range, and places the next stimulus level in the critical point of the curve. The test is shown to provide a fast convergence compared to other methods.

The UML method was employed here. The physical quantity that was studied is related to HEVC compression, which is the relative rate R_r. Given a reference video compressed with a rate of R_1, and the same video at another rate R_2, R_r is defined as:

$$R_r = (R_2 - R_1)/(R_1) \tag{6.2}$$

On the perceived scale, the preference probability is considered. This means that for a given reference video with a rate of R_1, the preference probability measures how much the same video with a rate of R_2 is preferred. In other words, how the relative relate affects the perceived preference. To do so, the observers in the psycho-physical tests were asked to select among the two compression levels (at R_1 and R_2) of a video, the decoded video that they prefer.

An example of the resulting psychometric function is given in Fig. 6.1. In this figure, we can see that sequences having a large negative relative rate are not preferred (and vice-versa), which reflects the fact that negative relative rate represents lower bitrate due to higher compression, which would necessarily result in lower visual quality, and thus less preference. The same is true for the opposite case. An interesting and most informative point on the curve is the point of 50% preference probability. This point is the one at which no clear preference towards any of the compared pairs is present. The observers would randomly prefer any one of the tested videos, as they do not perceive any difference. This point is commonly known as the point of subjective equality. Ideally, this point should correspond to the point where the compared videos are exactly the same perceived quality, which is the point of zero relative rate, but it appears at relative rate of approximately -10% (Fig. 6.1). This can be interpreted in the way that the

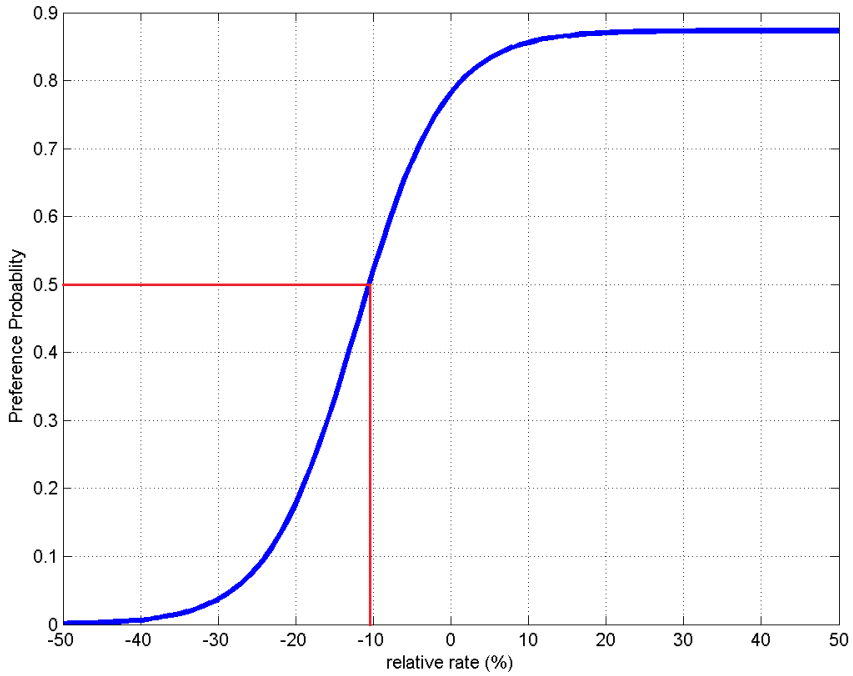


Figure 6.1 – An example of the measured subjective preference pychometric function. Red line represents the point of subjective equality.

given video is perceptually equivalent to the same video being compressed at a higher compression level, namely the given video possesses a certain amount of perceptual redundancy which can be exploited to produce 10% bitrate saving.

The UML test was conducted with 25 naive observers, with normal or corrected (to normal) vision. They received written instructions on using the software as well as the task they have to perform. A screen shot of the used software is shown in Fig. 6.2, in which two videos are simultaneously shown, and the observer task is to select the sequence with better perceived quality.

The subjective test was conducted in a professional room specifically designed for subjective testing. It complies with the ITU recommendations regarding the room lighting and screen brightness [173]. The used screen was a TVLogic LVM401 with a resolution of 1920x1080 at 60Hz. The viewing distance was 3H, where H is the screen height. The test duration was less than half an hour for all of the observers.

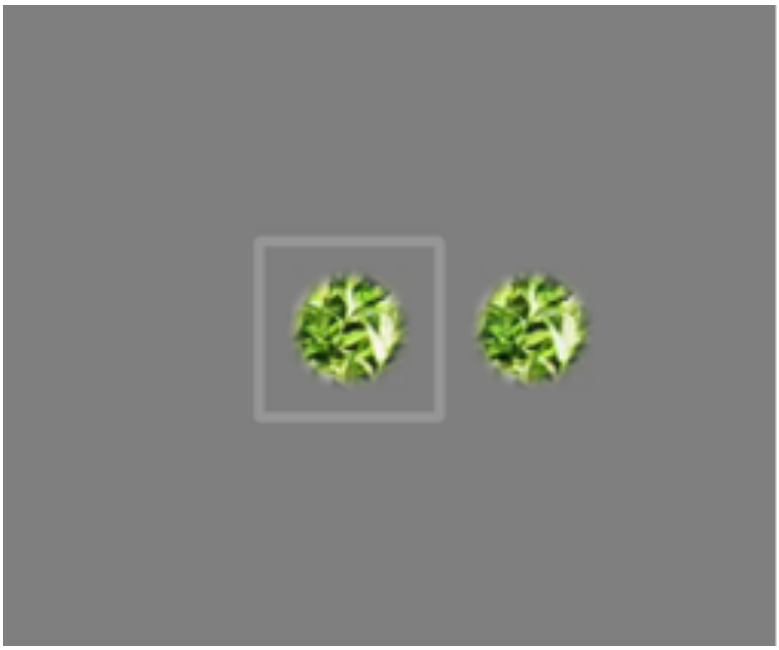


Figure 6.2 – Screen shot of the software used for psychophysical experiment.

6.2.2 Material

In most of the tasks involving visual quality assessment, the test materials are sequences having divergent contents, extending from 5 to 10 seconds long. However, other studies of psychophysical threshold estimation use simplistic signals with controlled properties (for example: bars, Wavelet-Gaussian patches). In this work, which covers both perceptual quality assessment and video compression, a combination of both is required.

The main goal of the video compression standard (HEVC), is to provide the best trade-off between rate and distortion. Thus, HEVC encoder selects the best prediction mode, splitting depth and etc. according to the instantaneous rate and distortion measure. The distortion is computed with a limited knowledge about the spatial and temporal part of the signal. Spatially, the maximum extent is limited by the Coding Tree Unit (CTU) size, with a maximum 64x64 pixels, and temporally it is limited by the decoded picture buffer, which is limited by few number of pictures. For this reason, the optimal perceptual optimization model should consider the smallest possible spatio-temporal extent.

Visually, the spatio-temporal extent is limited by the foveal vision in the spatial domain, and the minimum fixation time in the temporal domain. The exact mathematical numbers are discussed in Chapter 4 in developing V1-E. Taking into consideration these numbers, and trying to capture the textural phenomena for a shortest period, samples

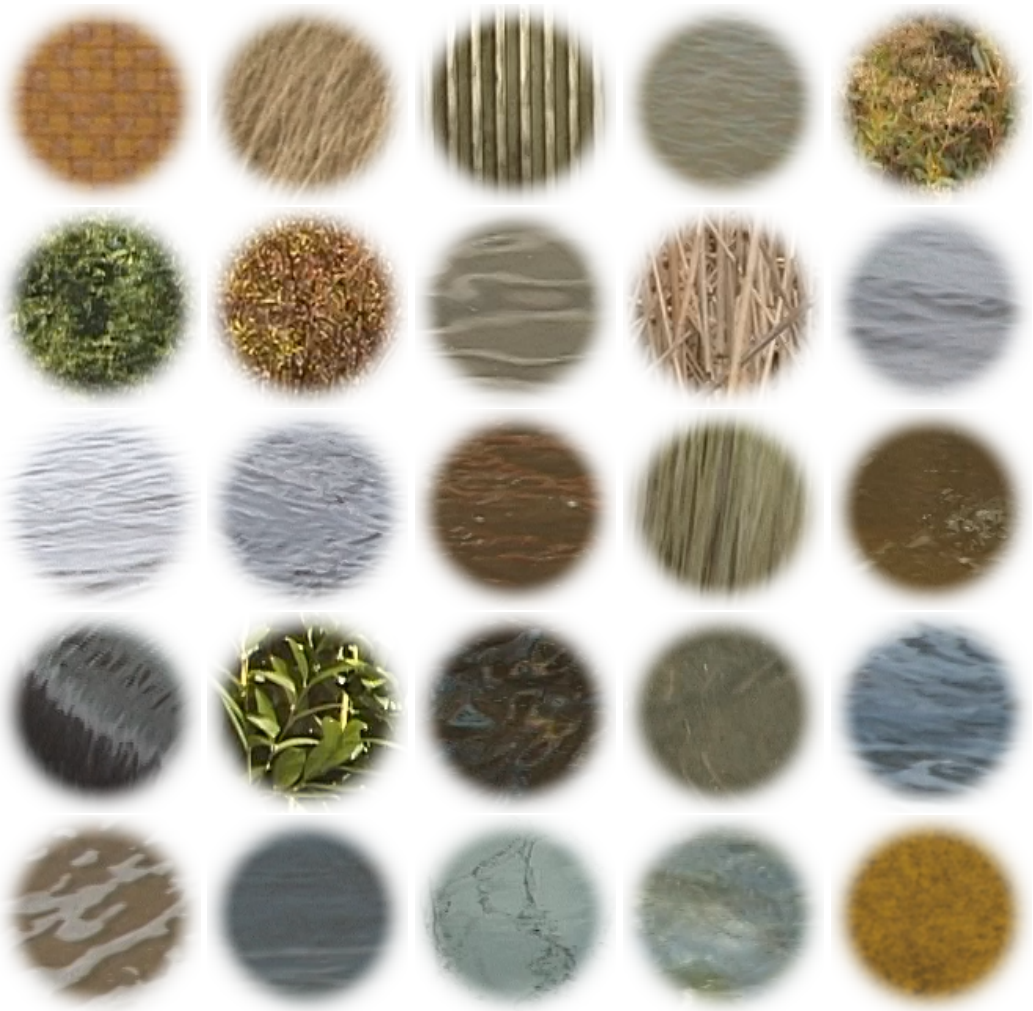


Figure 6.3 – Dataset used in this work (from HomoTex section 5.2.1), with SeqId from 1 to 25 (from top left to bottom right).

from the developed dataset (HomoTex dataset section 5.2.1) were used for the experiment. Temporally, they were limited to 500ms, which is assumed to be sufficient to represent the textural contents.

Accordingly, 25 sequences (shown in Fig. 6.3) were collected. For clarity, each video was assigned a sequence identity (SeqId) from 1 to 25. A circular windowed version (as shown in Fig. 6.3) was used in the experiment. The window diameter (91 pixels) was selected in order to show 1.5 degrees of visual angle, and the rest of each video was gradually faded to the background level using gaussian filter. This is done such that the signal falls within the foveal vision. Temporally, as the initial signal is quite short, it was repeated upon the end of the sequence with time reversal to avoid the temporal discontinuity artifact.

The sequences were compressed to 3 quality levels (high, medium, and low) using the HEVC reference encoder (HM 16.2 [8]). Certainly, these quality levels depend on the content, but overall a large range of quantization parameter (QP) was covered (minimum QP was 22, whereas the maximum was 47). This resulted in 75 source videos (SRCs), which were compared to other compression levels (HRCs) to obtain the preference probability using the UML procedure section 6.2.1.

6.2.3 Results

The results of the psychophysical test are 75 psychometric preference functions (as the one shown in Fig. 6.1), each of them represents the probability of preferring a given HRC over another HRC. For each function, the threshold of 50% probability of preference is retained, which represents the point of subjective equality (section 6.2.1). An example of one sequence is shown in Fig. 6.4, where we can see that the redundancy in the high quality region (low QP) is higher than for low quality region (large QP).

The overall average relative rate from the three quality points for all the sequences is shown in Fig. 6.5. We can see clearly that for most of the sequences, the corresponding subjective equality does not appear at the same bitrate. This clearly indicates that there are high perceptual redundancies, which can be exploited to reduce the bitrate, while maintaining an equivalent subjective equality.

6.3 Perceptual Redundancy and Distortion Tolerance Prediction

After measuring the 75 psychometric functions, two perceptual criteria in the context of distortion sensitivity can be defined. The first is called the perceptual redundancy, which quantifies the amount of visual information that can be omitted without causing any visual disturbance. The other is the distortion tolerance, which specifies for a given

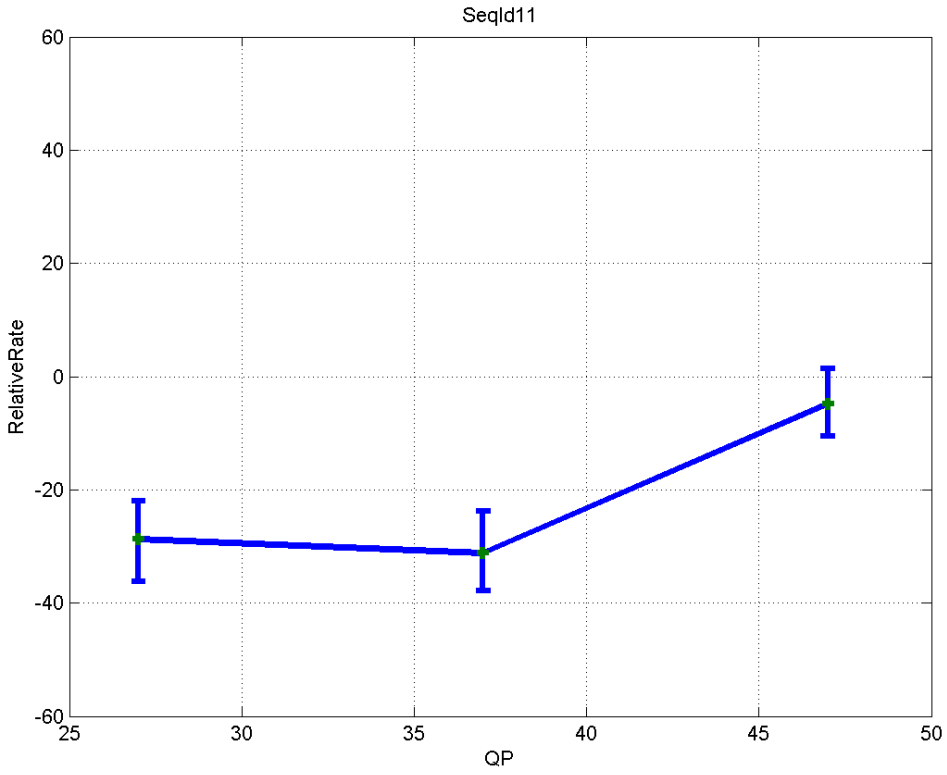


Figure 6.4 – An example of relative rated at equivalent subjective quality for the video with SeqId=11. Error bars correspond to 95% confidence interval.

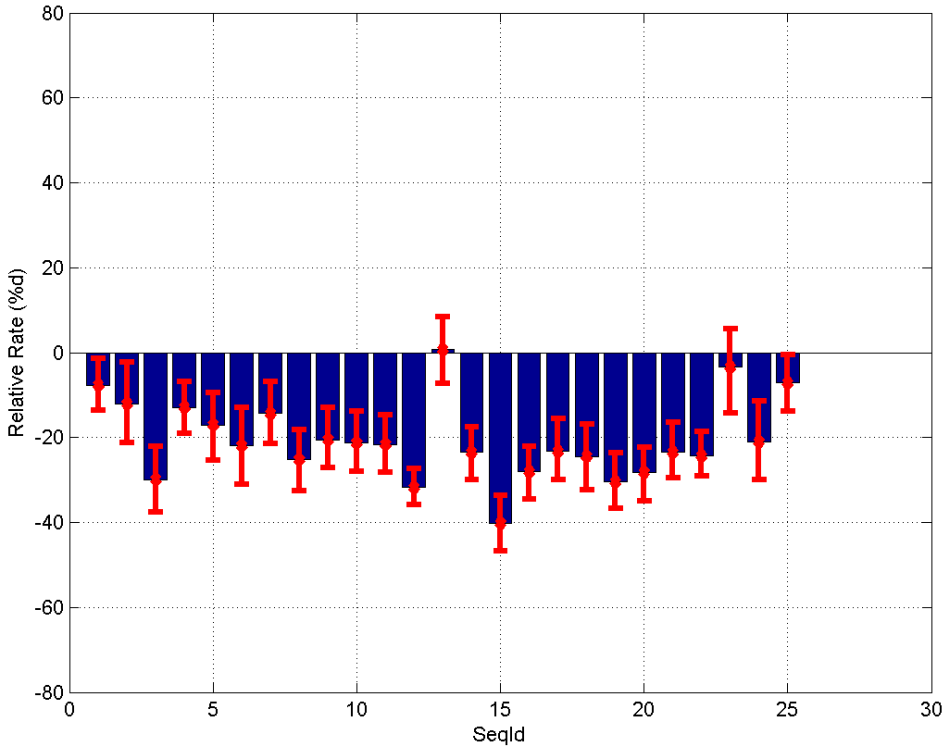


Figure 6.5 – Overall average relative rate of all videos. Error bars correspond to 95% confidence interval.

video how much the visual system can tolerate distortions.

The visual redundancy is expressed in terms of the objective metric of Peak Signal to Noise Ratio (PSNR). For each of the textures, at a certain compression level, specified by its quantization parameter (QP) value, the visual redundancy is the amount of minimum PSNR ($PSNR2$) that can be used, in contrast to the target PSNR ($PSNR1$), without making a visual difference. In other words, it measures how much signal fidelity, measured in PSNR, can be lost without being noticed by the observers. Then, the visual redundancy, expressed by $PSNR2$, can be predicted with features analysis by employing a regression model.

In our published work in [135], computationally simple features were used for the regression problem. Accordingly, we used the following set of descriptors: the standard Spatial Information (SI) and Temporal Information (TI) [174], The Colorfulness (CF) [175], the Gray Level Co-occurrence Matrix (GLCM) [176], and the set of dynamic texture descriptors defined in [94]. The GLCM descriptor combines 4 features, which are contrast, correlation, energy and homogeneity. Similarly, the following descriptors are defined in [94] for normal flow vectors: Divergence, Curl, Peakness and Orientations. For the frame based features, such as SI and TI, we experimented different temporal pooling strategies, such as temporal mean and standard deviation. A linear regression model was trained to predict the redundancies. Similarly, V1-E can be used as a features extractor for the linear regression model, and the performance can be compared. For purpose of simplicity, the dimensionality of V1-E has been reduced by considering less number of spatial frequencies, orientations and velocities. Overall, 9 features (V1 energies) were considered, corresponding to 4 orientations and two velocities. The performance of the linear regression process (normalized data), in terms of leave-one-out cross validation is given in Table 6.1.

The results indicate that the redundancies can be well predicted, with a high precision. It also shows that V1-E outperforms the proposed features set of [135], which indicates the importance of the perceptual features obtained by V1-E.

The possible use of this model is shown in Fig. 6.6. The input video, which is assumed to be homogeneous texture with a short spatio-temporal extent, is first analyzed by V1-E to extract the set of representative features. The features can be used to predict the amount of visual redundancies (MinPSNR) as compared to the target compression level, specified by QP. MinPSNR could be fed to the encoder (HEVC), so that it can try further compression, limited by MinPSNR, to achieve higher bitrate saving, while keeping the same perceived quality.

Beside the visual redundancy, one could also study the visual distortion tolerance. In contrast to the visual redundancy, the distortion tolerance is independent of the compression level, and thus it is a sequence dependent property. Mathematically, it is defined in the work as:

Features set	Number of features	Mean Squared Error	Mean Absolute Error
Combined features set [135]	9	0.002	0.038
V1-E	9	0.0013	0.036

Table 6.1 – Performance evaluation of perceptual redundancies predictors, using leave-one-out validation procedure.

$$DT = \frac{PSNR1 - PSNR2}{PSNR1} \tag{6.3}$$

Where $PSNR1$ and $PSNR2$ are as before: reference PSNR and lower PSNR that does not cause any observed difference. It should be noticed that the distortion sensitivity is inspired by the Weber law of sensation, which indicates that the perceived difference in a stimuli is proportional to the initial stimuli. In the case under study, the Weber law can be mathematically expressed as:

$$PSNR1 - PSNR2 = \alpha \times PNSR1 \tag{6.4}$$

Comparing Eqn. 6.3 with Eqn 6.4, the constant of the proportionality (α) is equivalent to the distortion tolerance. Since the psychophysical test covers three quality levels, the distortion sensitivity is considered as the average of the three levels.

To illustrate the distortion tolerance, Fig. 6.7 shows examples of images from videos with 3 Distortion tolerance values, at 3 equal compression levels. The lowest distortion tolerance value is found in the top sequence, the highest is for the sequence on the bottom, while the center sequence is the sequence exhibiting the median tolerance among the 25 sequences. One can observe that with increasing compression (from left to right in Fig. 6.7), the details of the sequence with lowest tolerance are easily diminished, in contrast to sequences with higher tolerance, the details are more persistent, and thus the human visual system can tolerate higher amount of distortions.

To test if V1-E can predict these properties of the sequences, SVM regression model was trained using the same number of features for the visual redundancy learning. This means 8 features were used, corresponding to the energies of V1-E obtained for 4 orientations, 2 velocities and 1 spatial frequency. The results in Table 6.2 show that the perceptual tolerance can be predicted with V1-E. However, the prediction is less accurate compared to the perceptual redundancies estimation (Table 6.1). The results could possibly be improved by expanding the V1-E features space, but this could cause an over-fitting because the number of samples is very small, which is the number of source videos (25).

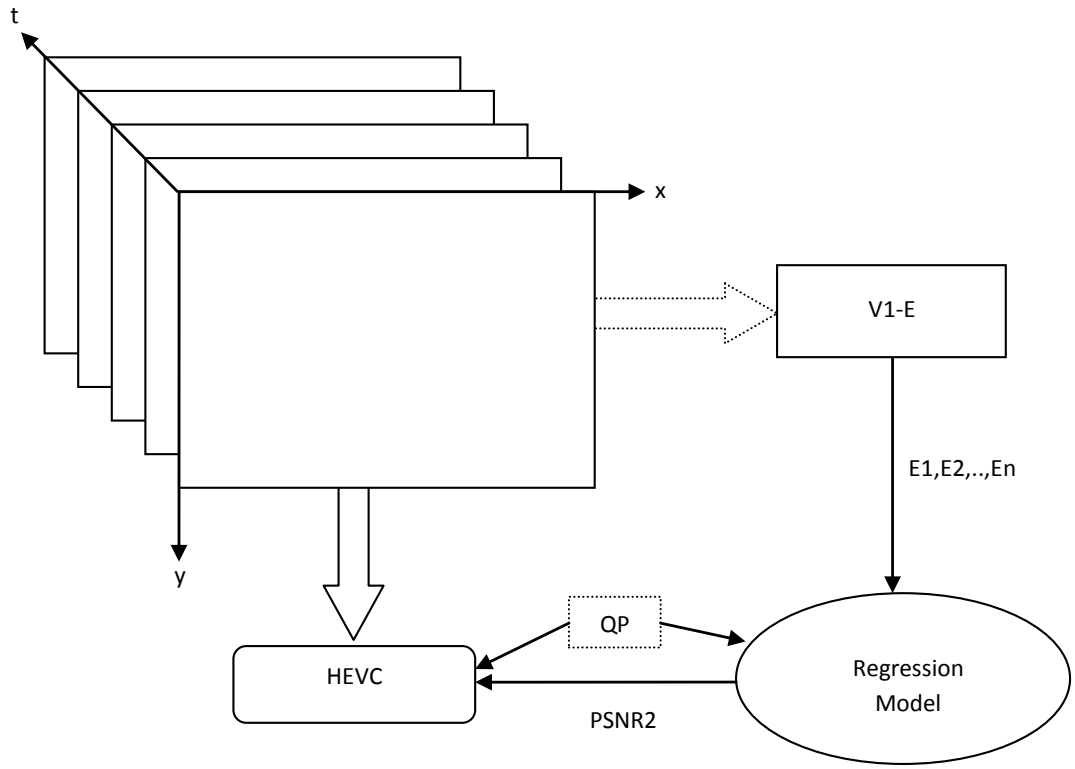


Figure 6.6 – Proposed perceptual optimization framework of HEVC Encoding.

Features set	Number of features	Mean Squared Error	Mean Absolute Error
V1-E	8	0.0067	0.25

Table 6.2 – Performance evaluation of perceptual tolerance predictors, using leave-one-out validation procedure.

Figure 6.7 – Images from sequences having different distortion sensitivity values, at three compression levels. From top to down: Lowest, middle and highest distortion tolerance. From left to right: HEVC compression with QP values of respectively 32,37 and 47.

6.4 Conclusion

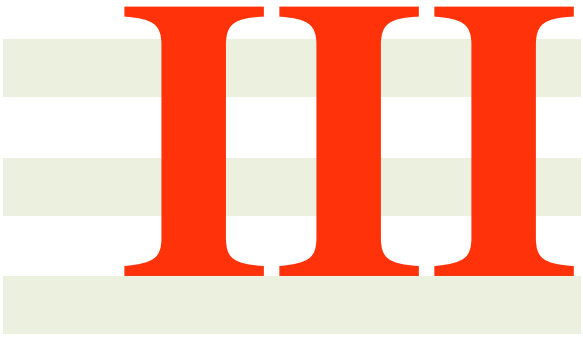
After examining the performance of V1-E as a similarity metric in both retrieval and recognition tests (Chapter 5), the objective here was to test V1-E as a features extractor. The features extracted by V1-E are tested in order to verify whether they can be used to predict visual properties of textures, in link with video compression, that are revealed with a psychophysical experiment.

Video compression is an important part of this thesis work. In this chapter, a first step in linking V1-E to video compression is made. First, a psychophysical procedure was employed to measure the amount of visual sensitivity of textures towards the perceived distortions due to HEVC compression. In other words, for a given texture with a certain compression level, the sensitivity tells us how much extra distortion can be allowed without altering the perceived quality.

From the experiment, two visual properties are defined: perceptual redundancies and perceptual distortion tolerance. In addition to the texture features, perceptual redundancies depend on the compression parameters, while the latter is independent of them.

The experimental results showed that V1-E, as a features extractor, can be used to predict both the visual redundancies as well as distortion tolerance. With a very small features set, it outperforms the set of features previously used in the published work. With this achievement, as well as the results of the retrieval and recognition tests, it is concluded that the proposed V1-E model is a proper model for textures, and it is achieving its design goals.

Exploiting the distortion sensitivity can provide a significant bitrate saving without altering the perceived quality. As can be seen in Fig. 6.5, up to 40% of the bitrate can be reduced. This is a first employment of V1-E model in the video compression system. This approach can be considered as a passive approach. This is because it is implemented outside the encoding processes. In the next part (Part III), we propose an active framework for perceptually optimizing the compression of textures, in which V1-E is employed for this purpose.



Model Application in Perceptually Optimized Video Compression

7

Proposed Optimization Framework

The main goal of the thesis work is to develop a model of visual texture perception, that can be used to drive the compression system in order to improve the rate-quality performance. The proposed model, V1-E, was shown to be a robust tool for expressing texture similarity, as well as a texture features extractor. The question in this chapter is how to use it inside the video compression system.

The chapter provides the general idea about the proposed perceptual optimization framework of the video compression standard. The framework utilizes the knowledge about perceptual texture similarity, for deriving a perceptual rate-distortion optimization. Then, initial experimenting with the proposed model is performed, accompanied by experimental results (visual and numerical).

The chapter is organized as follows: The introduction is given in section 7.1. The state of the art details about the rate-distortion optimization is provided in section 7.2. Section 7.3 explains the details of the proposed optimization frame-work. The distortion measure used for perceptual optimization is explained in section 7.4.1. The details of the rate-distortion implementation are provided in section 7.4.2. The experimental results are given in section 7.4.3, with the verification test in section 7.4.4 and the conclusion in section 7.5.

7.1 Introduction

Textures represent the perfect candidate for the optimization of the image/video compression system. This is due to their spatio-temporal homogeneity, which facilitate the characterization and analysis of their contents. Machine learning based approaches,

for predicting encoding parameters or enhancing the encoding system, are one of the successful solutions that utilize the analysis part of textures. Chapter 3 provides the state of the art methodologies for intelligent compression of textural contents.

Perceptual optimization, on the other hand, is another important approach for improving the compression system. It is generally concerned about improving the perceptual quality of the decoded image/video, for a given rate budget. It is thus mostly dependent on the definition and understanding of the perceptual quality, and how the compression distortions interact with it.

By developing V1-E, the goal was to use it as a perceptual optimization tool for the video compression system, specifically focusing on texture contents. V1-E has shown to be an excellent model of expressing texture similarity (Chapter 5), then, the natural way for utilizing this model is by improving the rate-similarity performance of the compression system. In other word, for a given rate budget, V1-E can be utilized to drive the encoder in such a manner that the perceptual similarity of the decoded video is higher compared to the default compression scenario. This is a part of a general framework of rate-quality optimization, which is extensively studied for perceptual optimization purposes.

7.2 State of the Art in Rate-Distortion Optimization

The rate-distortion optimization is a tool that is used inside the state of the art image and video encoder in order to achieve the best rate and distortion trade-off. For each encoder decision, the rate (expected number of bits) and distortion (expressed by a certain metric) is computed, and combined by a cost function. The cost value is then used to retain the best decision, which minimizes the computed cost.

The reference implementation of HEVC, realized by the HM software [8], follows a classical approach of rate distortion optimization, inherited by the older standard of Advanced Video Coding (AVC [120]). The cost function (J), involving the two variables: rate (R) and distortion (D), is defined as the following:

$$J = D + \lambda R \tag{7.1}$$

Where λ is the Lagrangian multiplier used for finding the minimum of the cost value J.

In the reference HEVC encoder, two distortion metrics are used. Namely, the Sum of Squared Differences (SSD), and the Sum of Absolute Transformed Differences (SATD), which is an extension of the Sum of Absolute Difference (SAD). For a given block of size NxM pixels, SSD is computed from the difference ($Diff(n, m)$) between the original block content and the reconstructed (decoded) version as follows:

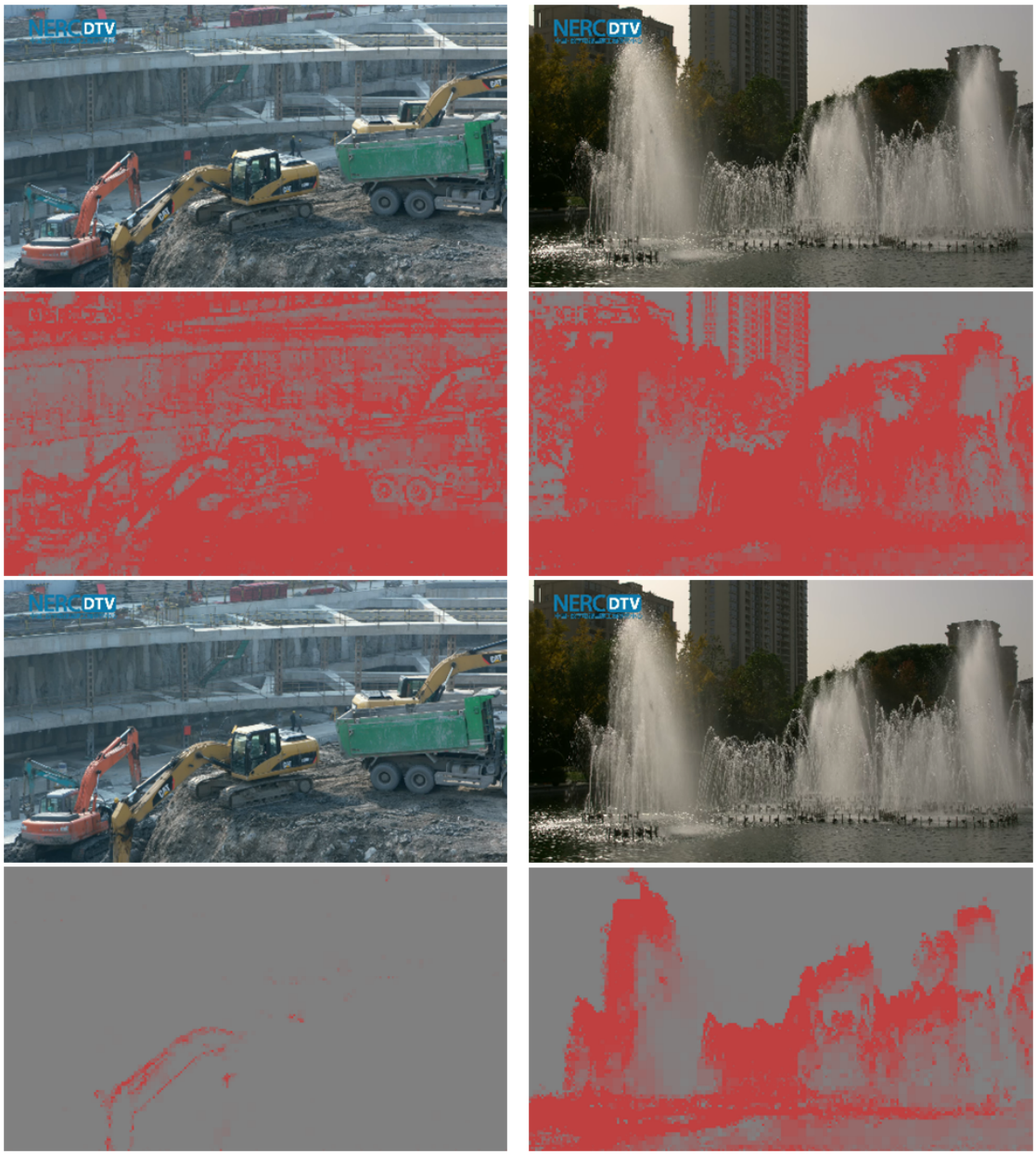


Figure 7.1 – Original frames (from the SJTU dataset [177]) and with their corresponding bitrate maps .

$$SSD(O,D) = \sum_{i=1}^{N}\sum_{j=1}^{M}(Diff(i,j))^2 \tag{7.2}$$

Similarly, SAD is defined as:

$$SAD(O,D) = \sum_{i=1}^{N}\sum_{j=1}^{M}|Diff(i,j)| \tag{7.3}$$

The difference between the two is the same as the difference between L1 and L2 norms. There are many applications that prefer L1 over L2 (vice versa). Beside SSD and SAD, SATD is computed by applying a transform to the difference before the sum, as follows:

$$SATD(O,D) = \sum_{i=1}^{N}\sum_{j=1}^{M}|T\ Diff(i,j)\ T'| \tag{7.4}$$

Where T is the transform matrix, and T' is the transpose of T. It should be noted that the motivation behind preferring SATD over SAD is that the residual signal to be encoded ($Diff(i,j)$) will be represented in the transform domain before compression, and the difference in that domain will correspond to the actual difference that will appear. However, for reducing the complexity, the reference HEVC implementation considers the simple transform of Hadamard, as an approximation to the discrete cosine transform that used for the actual compression, since Hadamard transform is much computationally simpler and requires only simple mathematical additions and subtractions.

In video compression scheme, the encoder has several options to encode each block. For example, in HEVC there are 35 intra-prediction directions. In addition, there is also a large number of candidates for inter-prediction, depending on the motion search range. For making the optimal choice, the cost function must be computed for each option, and the option showing minimum cost is then selected. Along with this, each block can be portioned into smaller blocks (both symmetrically and asymmetrically). The best decision then considers also the overall cost of the individual sub-blocks, as well as the partitioning cost.

Instead of performing exhaustive search for the best decision, HEVC reference implementation selects first a set of most probable candidates from which the best decision is drawn. This is done to reduce the large complexity involved in computing the rate term in the cost function, which requires performing the full compression steps (transform, quantization and entropy coding) which is equivalent to running the encoder as many times as the number of all candidates. Rather than this, an estimate of the rate, combined with SATD is used. After specifying the most probable candidates, including the ones with block partitioning, the accurate rate and SSD based distortions are used. In this manner, two cost functions can be defined:

$$J_1 = SATD + \lambda_1 R \tag{7.5}$$

$$J_2 = SSD + \lambda_2 R \tag{7.6}$$

Each of which has its own Lagrangian multiplier. Intuitively, SAD (and also SATD) can be approximately considered as the square root of SSD. Thus, the two λ are related in this equation:

$$\lambda_1 = \sqrt{\lambda_2} \tag{7.7}$$

The Lagrangian multiplier is inherited from older standards. Mathematically, it can be derived by searching for the value that minimizes the cost function. First taking the first derivative of the cost function in Eqn. 7.1 with respect to R yields:

$$\frac{\partial J}{\partial R} = \frac{\partial D}{\partial R} + \lambda$$

Then, set the derivative to zero for finding the minima:

$$0 = \frac{\partial D}{\partial R} + \lambda$$

Or

$$\lambda = -\frac{\partial D}{\partial R} \tag{7.8}$$

This means that λ is equal to the negative change of the distortion with respect to the rate. For this, the prior knowledge of the functional relation between the distortion and rate is needed, which is not feasible. For this reason, empirical solutions are usually employed, such as the one in [178].

In HEVC reference implementation, the value of λ used with SSD, i.e. λ_2 in Eqn. 7.6, is computed as follows [8]:

$$\lambda_2 = \alpha \times W_k \times 2^{((QP-12)/3)} \tag{7.9}$$

and the other λ for SATD (λ_1) is computed from Eqn. 7.5. α and W_k are multiplication factors depend on the encoder configuration, and QP is the quantization parameter.

Looking at Eqn. 7.9, one can see that for higher QP value, which corresponds to higher compression level, larger λ value is set. This means that encoder puts a higher weight for the rate, and thus seeks to reduce it as much as possible with loosely considering the distortion. The reverse is also true, for low compression (low QP), the rate is unimportant, and the encoder tries to provide the least possible distortion.

7.3 Proposed Framework

The distortion metrics used for the rate-distortion optimization, as explained in the section above, are very attractive from both mathematical and practical point of view. First, they represent the simplest and computationally most efficient metrics. Second, they exhibit a monotonic behavior with the signal distortion.

Perceptually speaking, this type of metrics is very poor. This is because they rely on the simple pixel difference, and do not consider any properties of the human visual system, such as the distortion sensitivity (Chapter 6), masking and others. The problem is even more severe in the case of textural contents. This is because textures similarity can highly deviate from the pixel-wise difference. Accordingly, the measured distortion cannot represent the perceived distortion.

For this reason, many studies proposed replacing these metrics by others that are more perceptually flavored. Examples of this are found in [179, 180, 181], where SSIM has been proposed as a replacement for these metrics.

On the other hand, the Lagrangian multiplier λ needs to be assigned depending on the visual importance of the region under consideration. In other words, higher λ can be assigned for visually unimportant areas, to spend fewer rate in comparison to other important areas with lower λ. This can even be done in combination with the distortion metrics, such that they are weighted in correspondence to the importance of each region. Examples of such approaches are the saliency-based coding [134], or Free-Energy principle-based coding [132] and others that provide heuristics based on blocks properties [182, 183, 184].

In the proposed perceptual optimization framework, the perceptual similarity (more precisely the dissimilarity) developed in this thesis is considered as replacement for the default metrics (SSD and SATD). In other words, the encoder will make its decisions based on perceptual criteria, rather than a simple mathematical one. The proposed framework utilizes V1-E for this purpose, in which two test scenarios are considered, where V1-E is either directly used as a similarity metric inside HEVC, or as a features extractor to predict the perceptual distortions for making the encoder decision.

7.4 Initial Experiment

The proposed framework, as explained in section 7.3, is a perceptual rate-distortion optimization model that considers replacing the default compression standard's distortion metrics by other metrics that are based on the properties of the human visual system. The main idea is to employ the texture similarity metrics for this purpose.

To do so, the similarity metric of V1-E, that is developed in this thesis (Chapter 4 for details), can be directly used. However, for the initial test, the work was limited to texture images, rather than texture videos. For this purpose, any perceptual texture-

image similarity metric could be used. We considered using an existing one, which is conceptually very similar to V1-E. The one that is selected for this task is a very recent metric, successfully applied for the purpose of textures retrieval, named as Structure-Texture Similarity Metric (STSIM). Its details are given in section 7.4.1.

7.4.1 Overview of STSIM

STSIM is a recent texture similarity metric, which has been successfully used for texture similarity assessment tasks like textures retrieval. It can be considered as an extension of the structure-similarity index (SSIM), mainly concerned with textures. In addition, it follows an asymptotic approach to the developed texture similarity metric (V1-E) within this work. For measuring texture similarity, STSIM performs the following steps [69]:

— **Subband Decomposition:** This is the first step in computing STSIM. The goal, similar to V1-E, is to model the neural processing in the human visual cortex. Instead of trying to resembling the exact neural processing, STSIM employs the steerable pyramid filter [68] as a tool for this decomposition.

— **Statistical Features Computation:** After the subband decomposition, STSIM defines some statistical features used for measuring texture similarity. Unlike V1-E, which only considers the subband energies, STSIM takes into account the mean, standard deviation and the horizontal and vertical auto-correlations coefficients. It should be noted that the subband signal is, by definition, a signal with zero mean value (no DC component). However, the mean only exists (not necessarily zero) in the case of windowed averaging. In this work, the implementation considered only global averaging. This essentially means that the considered STSIM implementation uses the root values of V1-E features, i.e. standard deviations instead of energies, accompanied by another features of correlations.

— **Pooling:** After computing the features on the subband domain, STSIM computes the similarity in the following manner: Assuming that the similarity score is to be computed between two textures (T_1 and T_2). Then, if ${F_1}^i$ and ${F_2}^i$ are certain features of the i^{th} subband from the first and second textures, STSIM computes the similarity of those two features in this way:

$$Sim(F_1, F_2) = \frac{2F_1F_2}{{F_1}^2 + {F_2}^2 + C} \tag{7.10}$$

which is exactly the same approach employed for V1-E (Eqn. 4.6). This similarity metric is computed for each feature, and average independently, over all the subands, for the standard deviations and the correlations. These three similarity values (Sim_{std}, Sim_{Hcorr}, Sim_{Vcorr}) are pooled as follows:

$$Sim_{std}{}^{\alpha} Sim_{Hcorr}{}^{\beta} Sim_{Vcorr}{}^{\gamma} \tag{7.11}$$

The performance of STSIM as a texture similarity metric has been partially discussed in section 2.7. It has shown an excellent performance for the common texture retrieval task, with up to 96% retrieval rate (Table 2.1).

7.4.2 STSIM for Rate-Distortion Optimization

STSIM, as shown in the previous section, is conceptually very close to the V1-E similarity metric, but limited to texture images. As explained in Chapter 7, the main goal is to use a certain perceptual metric to improve the rate-distortion performance of the video compression system, and STSIM is first tested in this initial experiment.

The similarity score obtained by STSIM cannot be directly used as a distortion measure inside the encoder. First, since the score is bounded between zero and one, where one is the highest similarity, it can be converted to a distortion D (dis-similarity) score by:

$$D_{STSIM} = 1 - STSIM$$

Second, the metric should consider also the block size in its computation. This is because it is used inside the rate-distortion loop of the encoder, in which encoder decisions over different blocks, due to block splitting, need to be taken into account. This is the same reason why the sum of squared differences (SSD), or the sum of absolute differences (SAD), is used instead of the mean squared difference or mean absolute difference. For this purpose, the distortion is scaled by the number of pixels that each block contains.

Finally, the rate-distortion function, as given in Eqn. 7.1, includes also a Lagrangian multiplier (λ), which is equal to the negative value of the derivative of distortion with respect to the rate (Eqn. 7.8). Because of the absence of information concerning the relationship between the STSIM and rate, no modification has been performed with respect to λ. Nevertheless, for the purpose of having the same range for the new metric D_{STSIM} as it is in SSD or SAD, D_{STSIM} is scaled by a proper value corresponding to each one.

In summary, STSIM was used to generate two distortion metrics (D_1 and D_2), which can be used to respectively replace SSD and SATD. Mathematically, they are expressed as:

$$D_1 = D_{STSIM} \times 255^2 \times N \tag{7.12}$$

$$D_2 = D_{STSIM} \times 255 \times N \tag{7.13}$$

Where 255 is the maximum pixel values for 8 bits representation, and N is the number of elements inside the block under consideration.

7.4.3 Experiments and Results

We have experimented the use of STSIM in HEVC for coding static textures. For this purpose, the Brodazt textures dataset, downloaded from USC-SIPI dataset [185], was used. This dataset contains 13 different gray scale textures which are extensively used in textures analysis for engineering and psychophysical experiments. The reference HEVC encoder used in this experiment was HM software version 9.0 [186]. In the following subsections, the details of each experiment are provided.

We experiment using STSIM inside the HM encoder as a distortion measure. Since the tested videos consist of one frame (one image), these metrics were only tested for Intra-Picture prediction scheme. The distortion metrics inside HM software that can be replaced by STSIM are: Sum of Absolute Transformed Difference (SATD) and Sum of Square Difference (SSD). For the details of each metric, the reader is referred to section 7.2.

For STSIM filter design, the simplified design of the steerable pyramid filter developed in [187], was used. The number of orientations we chose for the frequency decomposition is 2 while the number of scales was varying according to the patch size. We chose the number of scales equal to 2 for the HEVC prediction block sizes of 64x64, 32x32 and 16x16 and 1 for the size of 8x8 and 4x4. The experiments carried out are:

— Experiment 1: Replacing SATD alone: Eqn. 7.12
— Experiment 2: Replacing SSD alone: Eqn. 7.13
— Experiment 3: Replacing both SATD and SSD: Eqn. 7.12 and Eqn. 7.12

For all of the above experiments, the HM encoder was used to encode the texture videos with different Quantization Parameters (QP). The QPs which we chose were (22,26,32,36,43 and 51) to cover a wide range of compression ratios (from fine to coarse compression). We studied the effect of using these metrics on the decoded picture quality. We studied also the effect of using these metrics on the prediction mechanism of HEVC. The results of experiments are given in the next section.

Quality of the Decoded Textures

To show the result of the three experiments on the decoded pictures, we first provide one example of a decoded texture for each of the experiments explained before. The effect of using different distortion metrics is not very distinguishable for lightly compressed images. For this reason, we show here effect in a very high compression scenario.

Fig. 7.3 shows the effect of replacing SATD with STSIM (D_1) for QP value of 51. It can be seen that this has an effect of reducing the number of DC blocks and enhances

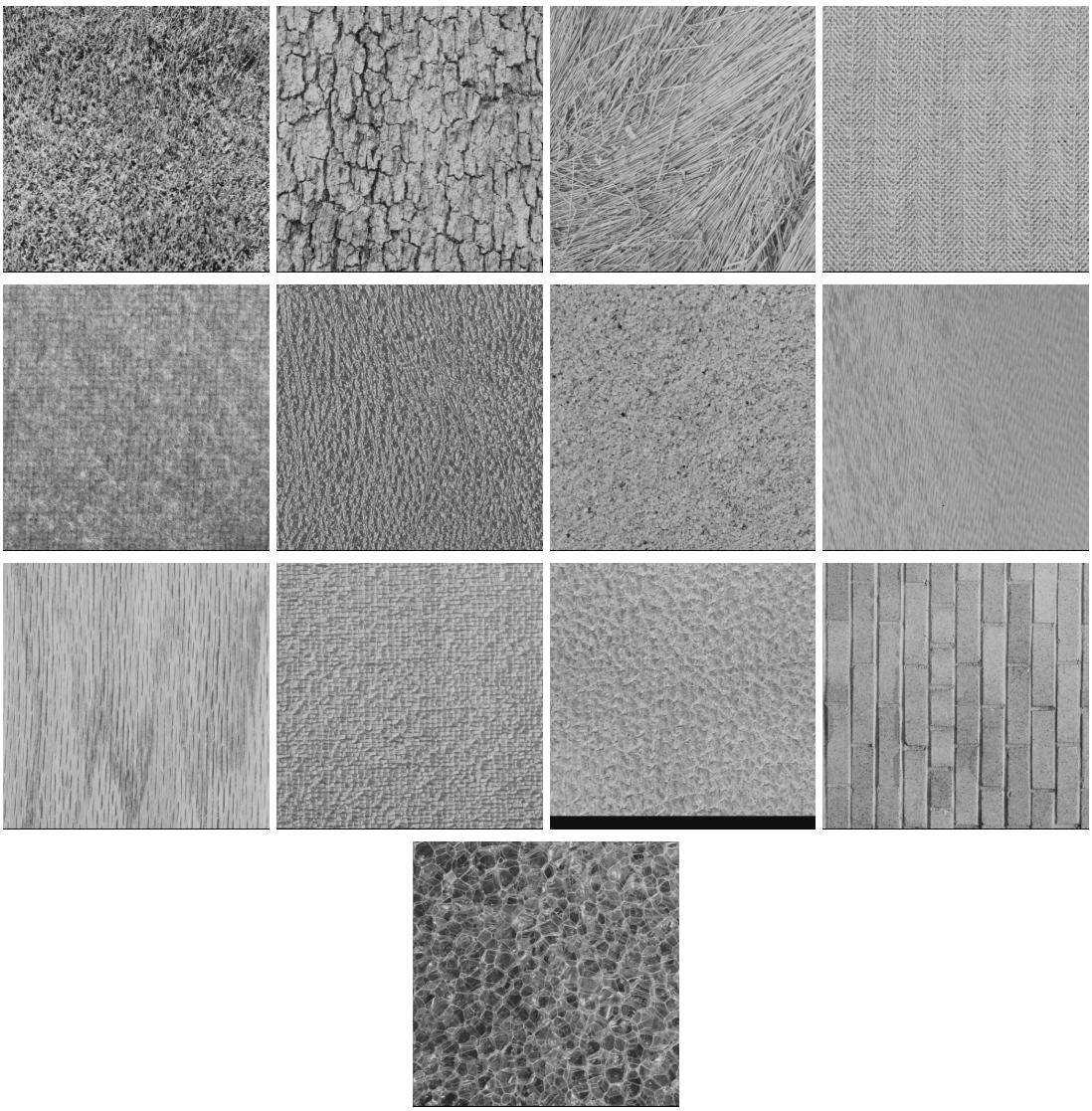


Figure 7.2 – Brodatz texture dataset (from USC-SIPI dataset [185]).

Figure 7.3 – Visual effect of replacing of SATD with STSIM (D_1) for QP value 51. Left: original texture, middle: encoded using default distortion function, right: STSIM instead of SATD.

Figure 7.4 – Visual effect of replacing SSD with D_2, QP value 51. Left: original Image, middle: encoded using default distortion measure, right: STSIM instead of SSD.

slightly the quality of the decoded image. This has very little improvement because in this approach, the new metrics affect only choosing the three most probable intra-prediction modes but does neither decide on the mode selection nor the block partition.

Replacing SSD with STSIM (D_2) improves significantly the quality of the decoded image. This can be seen in Fig. 7.4. If we examine carefully the decoded textures, it can be seen that STSIM introduces some artificial lines which were not present in the original image. The reason is that STSIM is less sensitive to rotational variations. With this property, the prediction signal generated using directional prediction may have little distortion computed by STSIM although it is in a wrong direction as compared with the original image.

The effect of replacing both SATD and SSD, as shown in Fig. 7.5, is not very different from replacing SSD only, since replacing SATD has a minor effect on the decoded picture as seen before.

Figure 7.5 – Visual effect of replacing both SATD and SSD with STSIM (D_1 and D_2), QP value 51. Left: original Image, middle: encoded using default distortion measure, right: STSIM instead of SATD and SSD.

One thing can be noticed is that the effect of producing artificial lines in STSIM is more obvious here as the all mode selections do not rely on pixel by pixel comparison. For the rest of the chapter, replacing both SATD and SSD is considered.

To elaborate more on the results, Fig. 7.6 provides other two examples of the dataset. We see an example of encoding a highly structured texture (First row). In high compression scenario, the texture looses most of its details when the default HEVC metrics are used. This is because many blocks are replaced by DC values. Using the similarity metrics, the overall structure of the texture can be retained. One can also notice that there exist many wrong directions, but the overall quality is much better.

The second example (second row of Fig. 7.6) of bubbles is a good example of high deviation from pixel fidelity when STSIM is used. We can see that the bubbles do not appear closed anymore and many directions appear which were not available in the original image. But overall, the decoded textures appear more pleasant when STSIM is used.

Rate Distortion Analysis

The rate distortion analysis is usually carried out using PSNR as a distortion measure. In our approach, PSNR is avoided as it is based on pixel difference, which is far away from the goal of this work. For this purpose, we sought another metric that is specifically designed for textures.

We used a texture similarity metric (Gabor Distance [65]) which is based on comparing features of textures in the Gaborian domain. These features correspond to the mean and standard deviation of the subband images obtained using Gabor filters. This metric often provides close by performance as compared to LRI and STSIM (Table 2.1) in terms of retrieval rate. The metric was downloaded from the author's website and

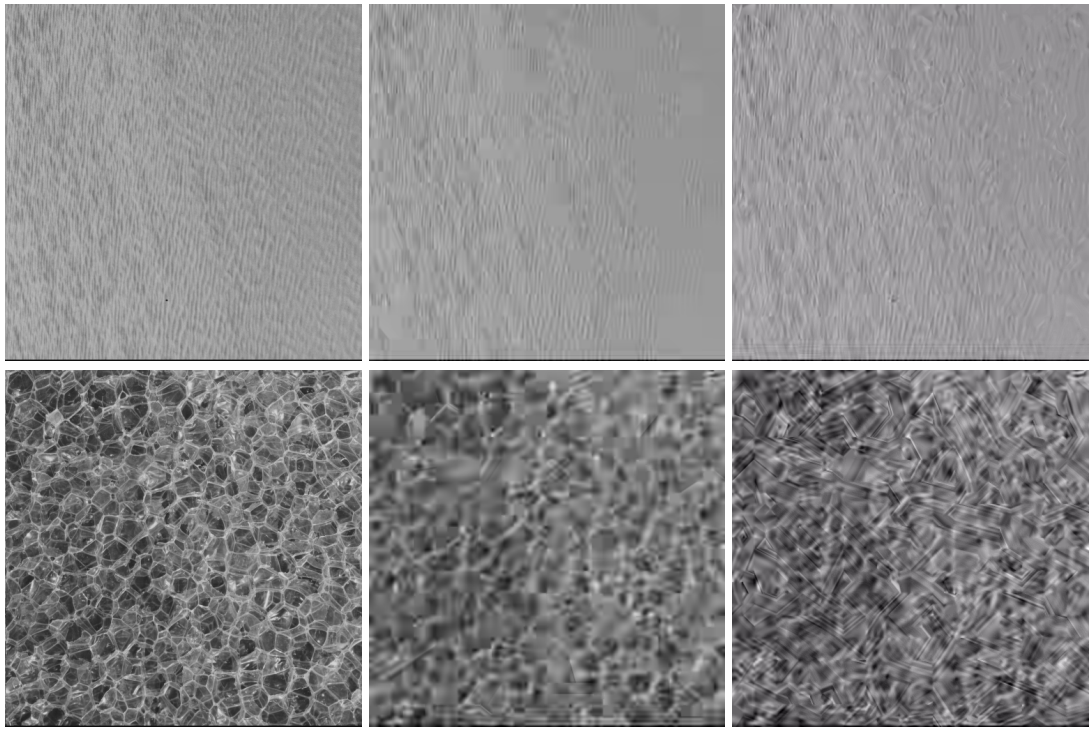


Figure 7.6 – Other examples of visual effects of replacing both SATD and SSD with STSIM (D_1 and D_2) for the same QP. From left to right: Original texture, compressed using HEVC default metrics and using STSIM.

Metric	PRCC	SROCC	KROCC
MSE	0.648826	0.633596	0.484619
PSNR	0.645703	0.633596	0.484619
SNR	0.705922	0.682557	0.536714
SSIM	0.763965	0.709877	0.577535
MSSIM	0.741126	0.687490	0.558827
VSNR	0.654254	0.611049	0.475622
VIFP	0.735588	0.681102	0.556914
UQI	0.736198	0.672940	0.533633
IFC	0.703233	0.655365	0.532372
NQM	0.666645	0.621154	0.472261
WSNR	0.709537	0.679295	0.558244
CWSSIM	0.778678	0.762760	0.606370
Gabor Distance[65]	**0.838151**	**0.807865**	**0.595272**

Table 7.1 – Statistical correlation measure of different quality metrics using QualTex dataset.

used as a distortion metrics in our work.

First of all, the performance of the Gabor distance metric as a texture quality metric is examined. For this purpose, QualTex dataset [188] is used. This dataset consists of 10 textures, with 5 types of distortions; each is subjectively evaluated, where the mean opinion scores are used as ground truth quality data. The metric score is then computed and compared to the ground truth data. Four statistical evaluation measures, recommended by the ITU-T P.1401 [189], are used to verify the performance, namely : Pearson Correlation Coefficient (PRCC), Root Mean Squared Error (RMSE) and Outlier Ratio (OR). In addition, two other measures: Spearman Rank Order Correlation Coefficient (SROCC) and Kendell Rank Order Correlation Coefficient (KROCC) were also considered. The results are shown in Tables 7.1 and 7.2. The result showed that this metric has a globally better correlation with the subjective evaluation.

By calculating the distance measured by this metric to the original texture for all the considered compression levels, we obtained the rate-distortion curves, or more precisely rate-dissimilarity curves, shown in Fig 7.7. We observe that in most cases, STSIM based rate-distortion optimization approach provides better score than the default metrics in the low rate region. For high rate region, no gain is achieved. It should be noted that these curves are associated with the third experiment, where both SATD and SSD are replaced.

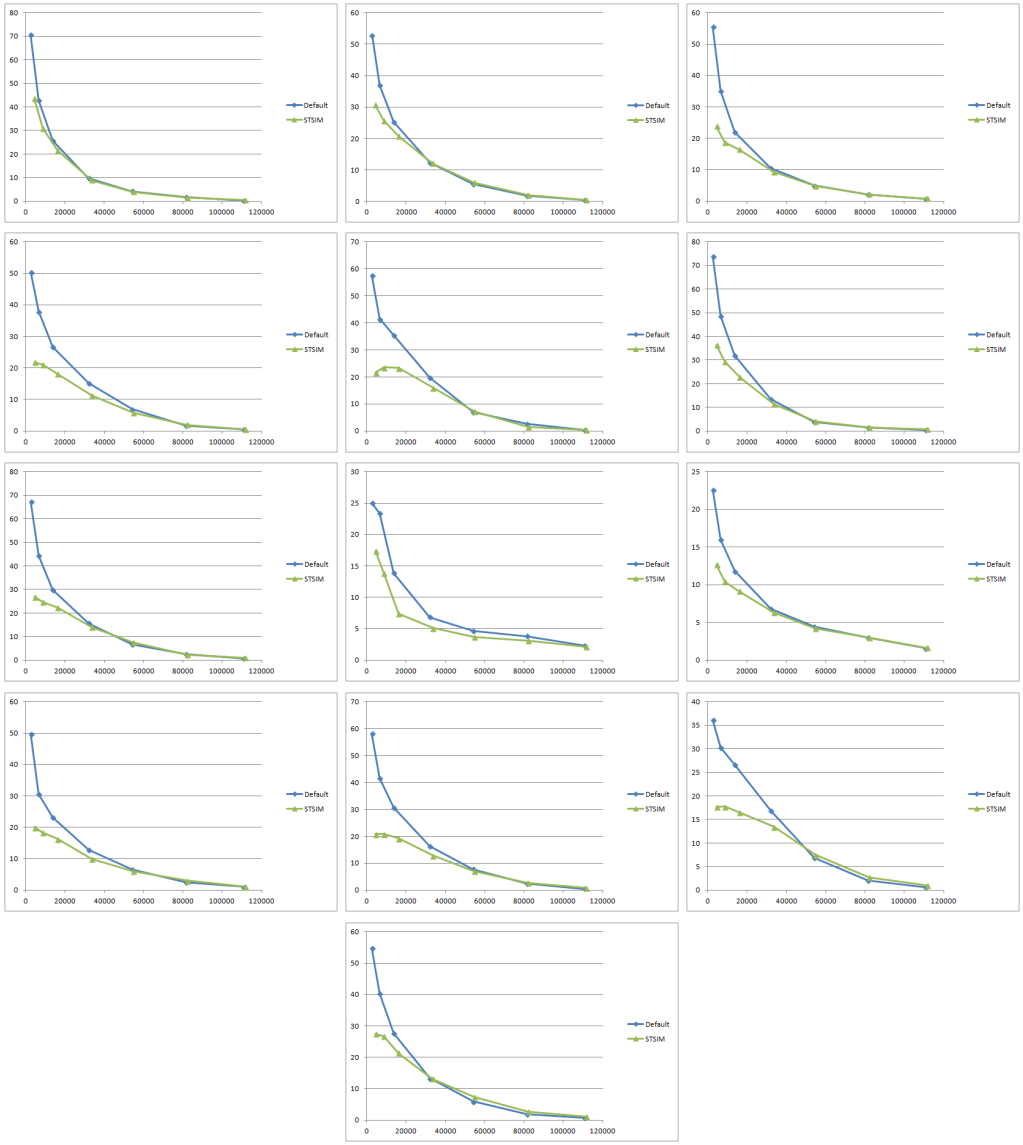


Figure 7.7 – Rate-distortion curves (using Gabor distance metric [65]) of the brodatz texture dataset shown in Fig. 7.6. x-axes: Bytes used to encode the texture, y-axes: distance to the original texture.

Metric	RMSE	OR
MSE	0.905398	0.638235
PSNR	0.908553	0.658824
SNR	0.842756	0.570588
SSIM	0.767757	0.532353
MSSIM	0.798822	0.526471
VSNR	0.899850	0.641176
VIFP	0.806037	0.567647
UQI	0.805247	0.558824
IFC	0.845932	0.594118
NQM	0.886882	0.679412
WSNR	0.838447	0.576471
CWSSIM	0.746541	0.547059
Gabor Distance[65]	**0.648988**	**0.505882**

Table 7.2 – Other statistical measure of different quality metrics on QualTex dataset.

Encoder Behavior Analysis

More analysis is carried out to understand the effects of using the perceptual metrics on the prediction mechanism. For this, we measured the frequency of splitting depths as a function of the quantization parameter. The corresponding histograms are shown in Fig. 7.8. The splitting depth of zero means that the prediction block has its maximal size (64x64). Increasing the splitting depth by one corresponds to partition the block into four sub-blocks. The histograms in Fig. 7.8 were scaled by the number of (4x4) blocks that each splitting has. This was done to have a fair comparison between splitting depths as each splitting occupies different areas of the frame. One can observe from these histograms that when the default metrics are used, the encoder uses small prediction blocks for low compression (low QP) to better approximate the input signal. For high compression, it tries to approximate large prediction blocks (mostly with DC values) to have better compression. The behavior totally changes when STSIM is used. The encoder behavior does not change much as the compression changes. It selects always large block sizes to approximate the input signal and small block sizes (less than 16x16) are rarely chosen. This can be explained by the fact that STSIM, in contrast to simple difference metrics, does not have the summability property. This is a very important property for the coding purpose. To understand this, let us take an example of what happens inside the rate-distortion loop. Imagine that the encoder is processing an NxN block, and obtained its best prediction mode, labeled by its cost value ($D + \lambda R$). Now, the encoder will try to split this block into smaller blocks, in quadtree manner, to check if better cost is achieved. The encoder, after finding the best modes for the four subblocks,

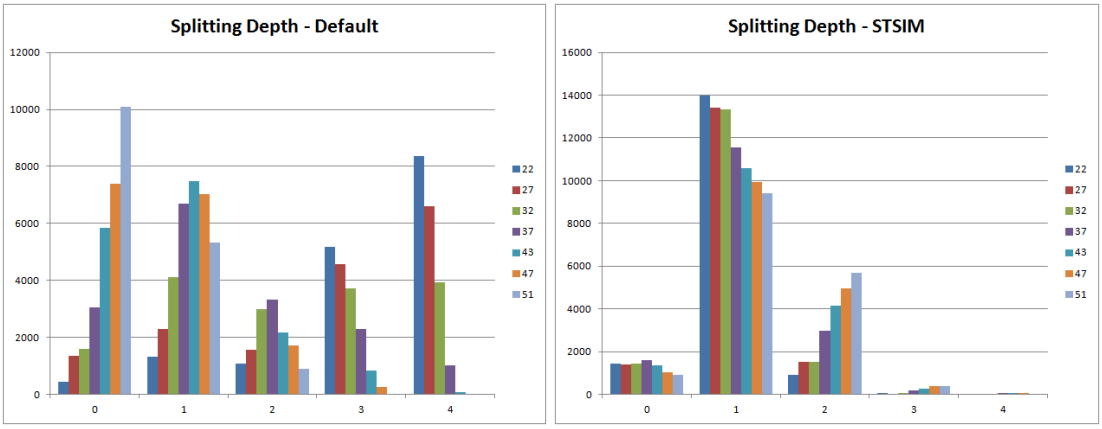


Figure 7.8 – Histograms of splitting depths vs QP. Each depth is scaled by the number of 4x4 block that it has.

will compare the sum of the individual costs to the bigger size cost. However, the sum of the four blocks does not equal to bigger block cost even if exactly the same distortion is present. This is because computing 4 times STSIM is not generally equal to computing 4 individual STSIMs. This would result in the case where the metric has a preferred block size. For example, in Fig. 7.8, the metric generally prefers the size of 32x32. This problem is not present in SSD or SATD, which makes them preferable inside the rate-distortion loop.

7.4.4 Verification of the results

To verify the results, we repeated the same experiments using a different dataset of textures. This time, we used some textures from QualTex texture dataset [188]. Examples of the decoded textures are shown in Fig. 7.10. We can clearly see that the fine details of the texture are better preserved when the texture similarity metrics are used, but when using the default metrics, all images look more blurry compared to the original ones.

The rate similarity curves are shown in Fig. 7.11. These curves are very much consistent with curves obtained using Brodatz textures (Fig. 7.7). This indicates clearly that these metrics perform better in low rate scenario.

The encoder behaves similarly in both datasets. As we see in Fig. 7.12, when STSIM is used, the encoder uses larger blocks independently from QP, this is for the same reason mentioned previously.

Figure 7.9 – QualTex textures.

Figure 7.10 – Examples of decoded textures using the same QP. From left to right: Original texture, compressed using HEVC default metrics and using STSIM.

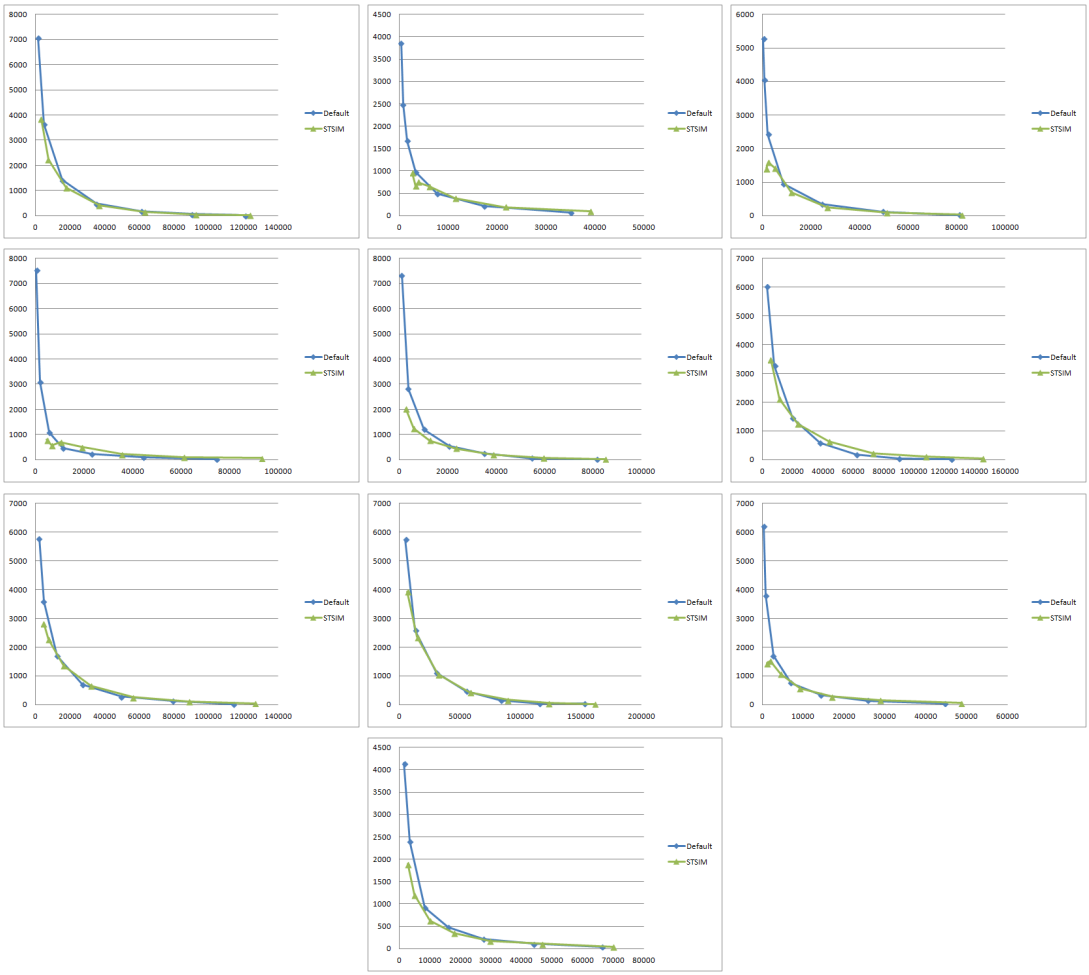


Figure 7.11 – Rate Distortion (using Gabor distance metric [65]) of the textures shown in Fig. 7.6. x-axes: Bytes used to encode the texture, y-axes: distance to the original texture. Indexes above each curve correspond to the same naming terminology in the dataset.

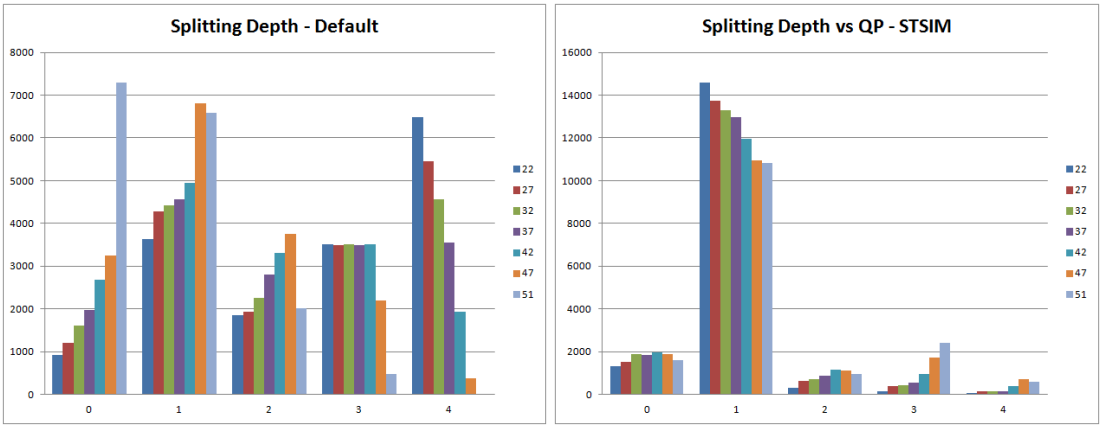


Figure 7.12 – Histograms of splitting depths vs QP. Each depth is scaled by the number of 4x4 block that it has.

7.5 Discussion and Conclusion

Research on encoder optimization for textures is often triggered; this is because textures are perfect candidates for this task. The reason is that textures represent homogeneous areas, which facilitate the learning based approaches. On the other hand, their details are relatively unimportant, which permits replacing them by synthetic ones that are easier to encode. Last, textures usually consume high bitrate, which mandates developing different coding tools for these contents.

In this chapter, a perceptual optimization framework has been proposed. This framework utilizes the developed perceptual model of texture similarity, with the aim of improving the rate-similarity behavior, rate-quality in parallel, of the state of the art video compression standard HEVC. Effectively, we tackled the classical framework of rate-distortion optimization, and considered replacing the default distortion metrics of HEVC (SATD and SSD) by perceptual ones, derived from perceptual texture similarity models, and modifying the corresponding Lagrangian multiplier. This has an effect of tuning the encoder decision in a perceptual manner. That is, the prediction mode selection, and the block splitting will be decided based on optimized rate and texture similarity, rather than rate and pixel similarity, which is assumed to improve the overall similarity of the compressed textures, and thus improving their quality.

After fully describing it, an initial experimentation with the proposed framework is performed to examine its performance. The test was limited to texture images, rather than texture videos. In this context, we considered a texture similarity metric that is conceptually very close to V1-E similarity metric. This metric, named STSIM, performs a subband decomposition, standard deviation and correlation computation, and pooling in a manner very similar to the V1-E similarity metric.

The initial test was conducted with texture images from Brodatz dataset. The visual effect is quite clear for high compression level, in which the overall similarity is higher when STSIM is employed, even though there exist large point-wise differences. In terms of rate-distortion behavior, the model is tested with a texture dissimilarity metrics based on Gabor decomposition. The results showed that the proposed model achieves more similarity to the original textures, as compared to the default HEVC, especially for high compression level.

The encoder behavior is has dramatically changed when STSIM is used. This is because STSIM is not a linear metric, and thus it does not possess the summability property. This has resulted in having a preferred metric size, and the encoder is thus biased to this value.

The results have been verified with another dataset, namely QualTex textures. The same experimental outcome is obtained. Thus, the conclusion can be generalized to any type of textured images.

8

Framework Implementation and Generalization

The proposed perceptual optimization framework, based on rate-distortion model, has shown an improved coding of texture images (Chapter 7). This was achieved by replacing the default distortion metrics in the video coding standard inside the rate-distortion loop, by perceptual ones obtained via texture similarity. The results, despite being promising in improving the rate-similarity behavior of the compression system, suffer from the non-linearity property inherited in texture similarity metrics, thus the summability does not hold, which made a dramatic change to the encoder behavior, in which the block sizes are not fairly used, but rather there is bias towards a metric preferred size.

In this chapter, we developed an indirect approach for employing texture similarity in the rate-distortion loop. In this approach, rather than relying on specific texture similarity metric, the dissimilarity is directly measured using a proper psycho-physical procedure. The main idea is to use the interesting properties of the pixel-wise difference metrics, such as MSE, by piece-wise linearly mapping its score to a perceptual one. V1-E, in turn, is used as a features extractor to predict this mapping for each texture.

The chapter is organized as follows: The general introduction is given in section 8.1. An overview of the existing psycho-physical tests for measuring the perceived distortions is provided in section 8.2, while the details of the conducted psychophysical experiment are provided in section 8.3. The implementation of the rate-distortion optimization, accompanied by the resulting rate improvement, is given in section 8.4. The generalization of the approach is discussed in section 8.5, while the conclusion is given

in section 8.6.

8.1 Introduction

In the initial experiments in Chapter 7, we have implemented the initial version of the proposed perceptual rate-distortion optimization framework. A texture similarity metric, namely STSIM, was used as a distortion measure inside the HEVC reference software to replace the default metrics of SSD and SATD. The initial results were promising in terms of improving the visual quality of the decoded images, and also enhancing the encoder rate-distortion behavior.

Nevertheless, there exist some issues with this initial implementation:

1. Complexity: STSIM is much more computationally complex, as compared to SSD or SATD. Since the metric is used for both pre-modes selection and full rate-distortion loop, it is invoked quite a lot of times inside each CTU, which results in much larger encoding time as compared to the default implementation.

2. Summability: This is an important issue. The encoder is biased to some extent to a certain block size. This results in preventing other possible block sizes from being selected.

For this reason, pixel differences based distortion metrics are extensively used for encoder optimization. Other than this, there are many other properties of pixel difference based metrics, especially SSD that are listed in [190]. Among them, convexity, symmetry and differentiability are present in this metric, facilitating its use for optimization purposes.

Accordingly, in this chapter, we present a different implementation of the proposed perceptual rate-distortion optimization framework, taking into account the importance of MSE and the issues associated with texture similarity metrics. To do so, we were looking into finding a mapping function, which takes as an input the computed distortion (SSD), and maps it to its perceptual value. In other words, instead of relying on the existing metrics of texture similarity for improving the compression quality, we directly measure the perceived distortion (or dissimilarity) due to HEVC compression on textures [191], and find a mathematical function that relates the computed distortions to the measured (perceptual) ones. This mathematical function is, by nature, sequence dependent. The role of V1-E, in contrast to previous approach, is not to be a similarity metric, but rather to use it as a features extractor, with which the mapping function can be learned.

8.2 Measuring the Perceived Distortions due to HEVC

To deduce the relationship between the computed and perceived distortions, we first need to conduct a psychophysical experiment to measure the perceived distortions. To do so, a proper subjective testing methodology must be selected, which well matches the target behind the experiment.

The standard subjective tests methodologies are listed in the BT-500 [192] and P.910 [193] documents of International Telecommunications Union Recommendations Center (ITU-T). There exist several methodologies that can be either single stimulus or double stimulus. In the single stimulus methodologies, the basic method is known as the Absolute Category Rating (ACR). In ACR, the observers are shown a visual stimulus (an image or video), with a certain amount of distortion, and the task is to assign a category to this stimulus (excellent, good, fair, poor or bad). Similar to this, in the Absolute Category Rating with Hidden Reference (ACR-HR), the reference (undistorted stimuli) is shown to the observers, without explicitly informing the observers. In this way, differential quality can be estimated in order to reduce the source bias. Instead of having discrete categories, the Single Stimulus Continuous Quality Scale Evaluation (SSCQSE) defines a continuous quality scale (from 0 to 100%), where the observers vote using a slider device.

Compared to single stimulus methodologies, the double stimulus ones allow the observers to compare the current stimulus to the reference one (undistorted). The Double Stimulus Impairment Scale (DSIS) is the double stimulus version of ACR. However, it uses the impairment scale instead of quality, that is, the used scale is (imperceptible, perceptible but not annoying, slightly annoying, and very annoying). Similarly, the Double Stimulus Continuous Quality Scale (DSCQS) is analogous to SSCQSE without informing the observers about the reference stimulus.

Instead of voting for each individual stimuli, the Subjective Assessment Methodology for Video Coding (SAMVIQ) [194] enables the observers to see all the possible distortions of stimulus, view them as many times as needed, and then vote for each of them. For this reason, SAMVIQ provides usually better precision than ACR [195].

Besides these methods, the classical method of Paired Comparison (PC) is highly preferred. This is because it requires the least effort from the observers to obtain the results, which makes it highly reliable. The observer votes are not directly used, but rather converted to quality scale using Thurstone or Bradley-Terry Model [196]. The PC test usually requires large number of trials because of comparing every possible pairs, which makes it unpractical when the number of degradations is big. For this reason, adaptive PC tests have been developed such as [197] [198]. On the other side, extensions of PC has been developed in [199, 200, 201].

Similar to PC, another recent binary decision subjective test was introduced by Maloney et al. [202], which is known as Maximum Likelihood Difference Scaling (MLDS). The methodology is suited to measure the suprathreshold distortion profile of a stimulus.

In MLDS, the observers are asked to compare two suprathreshold pairs having different amount of distortions, and the task is to select the pair which shows less (or higher) perceptual difference. The binary observation is then converted to continuous difference scale using maximum likelihood estimation. The method has shown high theoretical precision even in the case of using limited number of trails and/or observers. It has been also tested in the context of estimating the visual quality of compressed images [203]. The authors also provided a software package in [204]. This methodology was employed in this work to measure the perceived distortions, due to HEVC compression. The details of the psychophysical experiment are given in section 8.3.1.

8.3 Psycho-physical Measuring of the Perceived Difference

8.3.1 Method

As mentioned in section 8.2, we opted to use the Maximum Likelihood Difference Scaling methodology for measuring the perceptual distortions due to HEVC. In this methodology, specifically in the four-points protocol, observers are presented 2 pairs of stimuli. The amount of the physical quality, say distortion, is different in each stimulus. The observer task is to determine which of the pairs results in higher perceived difference. The individual differences are then converted, by the means of maximum likelihood estimation, to a difference scale that covers the range of the physical quantity.

Adapting MLDS in this work is pretty straightforward. For each texture video under consideration, the observers were presented 4 instances of that video, corresponding to 4 compression levels. The observers are asked to select one pair, out of two, that shows higher differences. Fig. 8.1 shows the constellation of the videos. Horizontally, the distance between the video pairs is kept at 1 degree of visual angle, while vertically it was 3 degrees. The short horizontal distance is selected in order to facilitate the comparison between the two videos in one pair, while the larger distance serves to avoid the influence of cross pair interaction. The selection of pairs was done via the keyboard arrows, and by pressing "enter" to validate the selection.

The subjective test was conducted in a professional room specifically designed for subjective testing. It complies with the ITU recommendations regarding the room lighting and screen brightness [173]. The used screen was a TVLogic LVM401 with a resolution of 1920x1080 at 60Hz. The viewing distance was 3H, where H is the screen height.

In total, 6 expert subjects participated in the test, where all of them have their research field involving image/video quality assessment and coding. Each observer made 150 comparisons, which took on average 20 minutes.

Figure 8.1 – Screen shot of the software used for MLDS

8.3.2 Material

There exist some constraints in selecting the material for the psychophysical experiment. First, as already discussed in section 6.2.2, the video encoder has very limited access to both spatial and temporal domains. This is because it deals with small block, and does not store the full video while encoding it. Accordingly, all the encoder decision are made on this limited access, which necessitate the development of spatio-temporally short-term distortion model. Second, we are interested in the foveal vision, for the reason that the human visual system is mostly attentive to this range of vision, this makes the distortion measured in this type of vision represent the upper case of tolerated distortions.

For this reason, we used the HomoTex dataset (section 5.2.1 and Appendix A). This dataset is developed in this work in order to provide a wide-range of homogenous texture videos. Spatially, the 128x128 (2 degrees of visual angle) spatial dimension were masked by a circular Gaussian window, such that the inner 1.5 degrees of visual angle is presented, and the rest is faded to the background level. Temporally, 500ms were considered to be a good compromise between the minimum fixation time, and the time for capturing the visual phenomena. To provide the observers with adequate time for decision making, the videos were continuously repeated. However, upon the end of one loop of video, the repetition was made with time reversal in order to avoid temporal

discontinuity.

In the subjective test, one can only afford limited number of sequences. Thus, we looked for sequences with distinguishable features. First, we selected 43 videos out of the 47 ones, by removing the ones with high visual similarity. Second, we considered the HEVC performance as an important feature for clustering the sequences. Using HEVC reference software (HM 16.2 [8]), the sequences were encoded to 4 levels of Quantization Parameters (QP's), that correspond to the common testing conditions. Namely, the QP's of (22,27,32 and 37) were used. The corresponding rate-distortion curves are shown in Fig. 8.2. The Bjontegaard delta PSNR (BD-PSNR [205]) was computed between all sequences. The sequence which has the minimum sum of BD-PSNR compared with all the other sequences is considered as the reference one, and the BD-PSNR with respect to this sequence is considered as the feature for clustering. Using this feature, the k-means clustering algorithms (k=8) was used to generate 8 classes of textures. One texture of each class, representing the class center, was used in the subjective experiment. The resulting 8 sequences are shown in Fig. 8.3. For clarity, each video was assigned to a SeqId from 1 to 8, which follows the same order as shown in the figure (from left to right, and top to bottom).

8.3.3 Subjective Test Results

The raw data of selected pairs from the subjective experiment were converted to a perceptual scale using the software package provided by the authors of MLDS in [204]. The resulting perceptual scales are shown in Fig. 8.4 for four sequences. The x-axes represents the overall average MSE of all the frames, whereas the y-axes represents the perceived difference. The confidence intervals are computed by learning the observers probability and repeating 10000 simulations using a boot-strapping procedure as explained in [204].

The four curves shown in Fig. 8.4 represent two different trends in the MSE vs. perceptual difference relationship. The first trend, as for SeqId 2, shows that there is a big deviation between the measured distortion (MSE) and the perceived one. On the other hand, the second trend, which is shown for SeqId 7, indicates that MSE is directly proportional to the perceived value of distortion.

The relationship cannot be easily fit with any mathematical function. We opted to use a piece-wise linear function to represent it. Thus, the mapping function between the computed distortion (MSE) and the perceived difference is a sequence dependent function, which is parameterized by a set of slopes and y-intersections for each sequence function period. This function is used after for perceptual optimization, and we show that it can be predicted by using the features extracted by V1-E perceptual texture model.

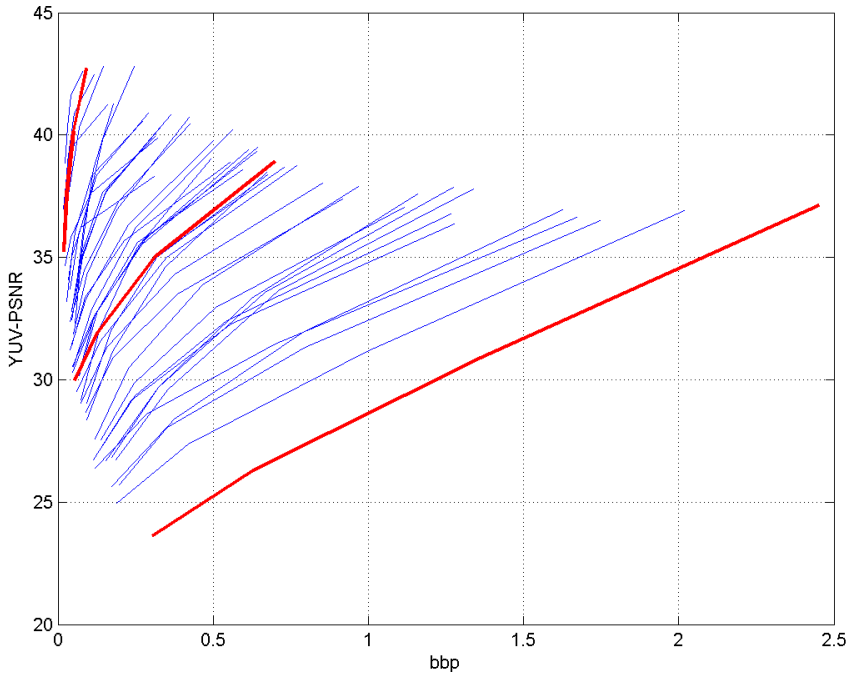


Figure 8.2 – Rate distortion curves of HomoTex dataset, using 4 QP values corresponding to the common testing conditions. Red curves represent the top, median and bottom curves.

8.4 Perceptual Optimization of HEVC

8.4.1 Optimization Process

Looking carefully at the curves in Fig. 8.4, we can see that sequences belonging to the first trend, where a large deviation between the computed and perceived distortions exists, are suitable candidates for perceptual optimization. This is because this large deviation can lead to wrong decisions inside the coding loop and thus would not lead to the optimal rate quality compromise. The SeqId's of the sequences belonging to this trend are given in the first row of Table 8.1.

A straightforward way to utilize the subjective test result in the video compression scenario (HEVC) is to map the distortion measure in HEVC to its perceptual value. To achieve this, we used linear piece-wise mapping functions, derived from the subjective test, to convert the measured MSE into a perceptual value (SSD_p) as follows:

Figure 8.3 – Sequences used for subjective test.

$$\begin{aligned} SSD_p &= (\alpha MSE + \beta) \times N \\ &= \alpha SSD + \beta N \end{aligned} \tag{8.1}$$

where N is the number of pixels belonging to the given block.

The Lagrangian multiplier (λ), as discussed in section 7.2 (Eqn. 7.8), is equal to the negative derivative of the distortion with respect to the rate. Thus, the new Lagrangian multiplier (λ_p) value can be also derived as follows:

$$\begin{aligned} \lambda_p &= -\frac{\partial SSD_p}{\partial r} \\ &= (\frac{\partial SSD_p}{\partial SSD}) \times (-\frac{\partial SSD}{\partial r}) \\ &= \alpha \times \lambda \end{aligned} \tag{8.2}$$

Thus, the λ_p is a scaled version of the previous λ.

According to this analysis, the perceptual optimization process, for each texture type, consists of piece-wise mapping function and scaling factor.

8.4.2 Estimating the Bitrate Saving

Instead of only testing whether the proposed framework is more preferred, we were interested in measuring the amount of bitrate saving that it can provide. For this purpose, one needs to compare the bitrate at equal subjective quality. However, finding the

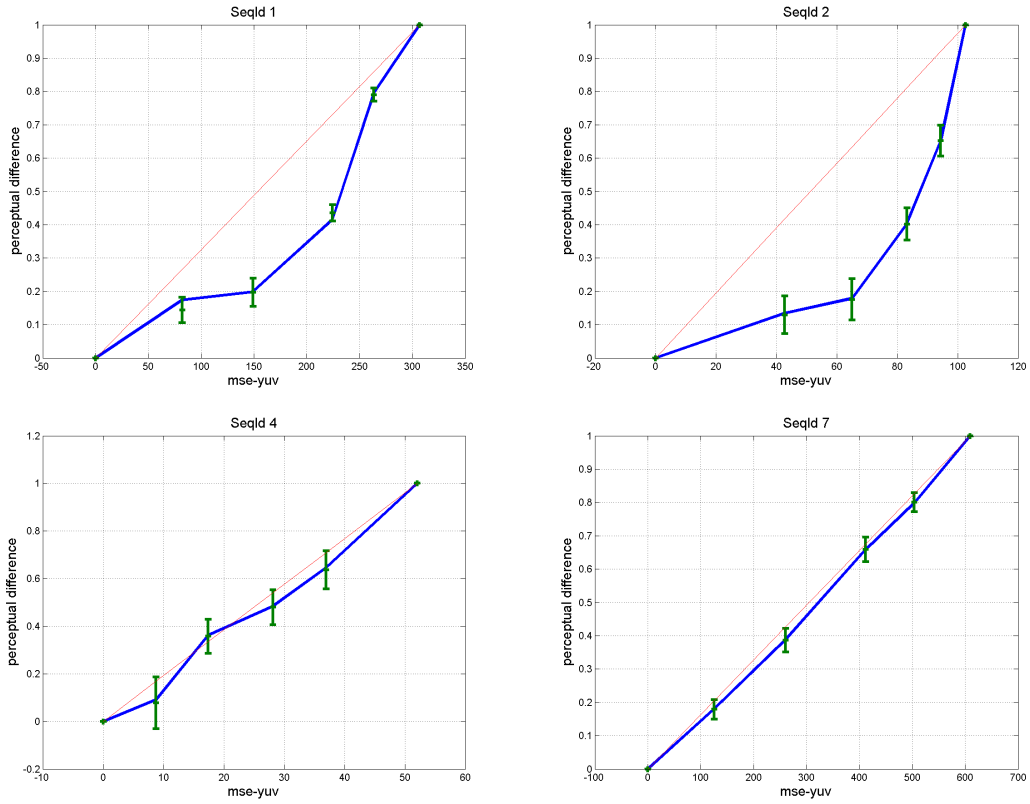


Figure 8.4 – Subjective test results of MLDS for 4 sequences.

subjective equality is not straightforward, but rather requires a specific psychophysical test known as threshold estimation. This threshold corresponds to the point of subjective equality. As discussed in section 6.2, we need to estimate the preference psychometric function, and find the 50% preference point.

Instead of using the UML procedure, in contrast to the case of estimating the distortion sensitivity (Chaper 6), we used the classical psychophysical test of forced choice, in which the observers select one sequence, out of two, that they prefer. We designed a specific subjective test to estimate this threshold, namely a forced choice (yes/no) method. We fixed the reference encoder bitrate (HM 16.2), and used the optimized encoder to produce 7 bitrate values around the reference rate. Each pair of dynamic patches, obtained from reference and optimized encoders, was shown to the observers 6 times. A screen shot of the used software is shown in Fig. 8.5.

Using the same subjective setup as in 8.3.1, the preference probability was computed. The preference probability is a psychometric function that can be generally fitted with an S shaped function. We used Weibull function (from the Matlab psychophysics toolbox [206]) as a fitting function using the maximum likelihood estimation. An ex-

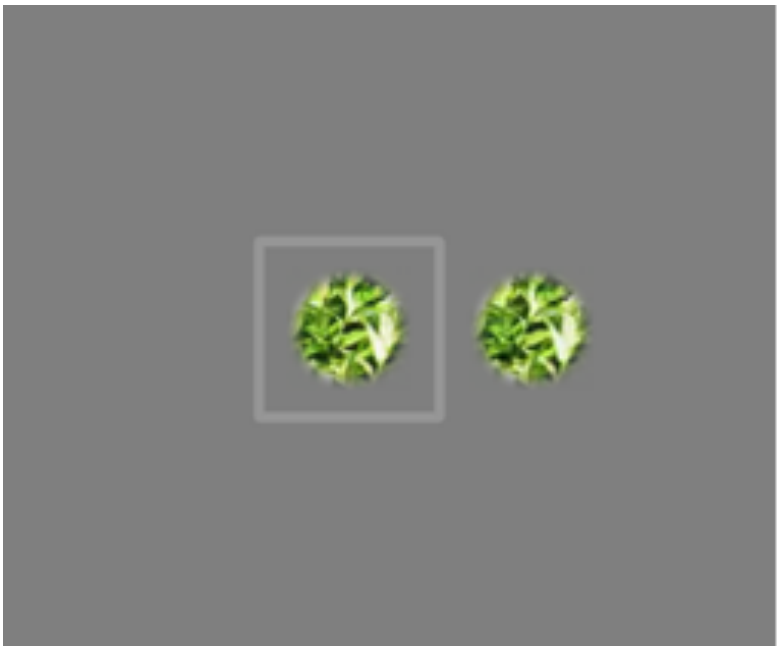


Figure 8.5 – Screen shot of the software used for forced choice experiment.

ample of the preference probability with its fitting function is shown Fig. 8.6. The fitted preference probability is then used to infer the iso-quality point, which corresponds to 50% value of probability of preference.

8.4.3 Perceptual Optimization Results

The optimization process in section 8.4.1 was used for the sequences shown in Table 8.1, as being sequences with possible perceptual optimization (see section 8.4.1). To investigate the amount of possible bitrate saving, we considered the points where the maximum deviation between the MSE and perceptual difference is assumed to occur. This corresponds to the QP values shown in Table 8.1. The bitrate saving is computed as follows:

$$Saving = (R_d - R_p)/R_d \tag{8.3}$$

where R_d and R_p represent the rate of the default HEVC and the perceptually optimized version respectively. We can clearly see in Table 8.1 that the proposed optimization process can highly reduce the bitrate (up to 17.7%).

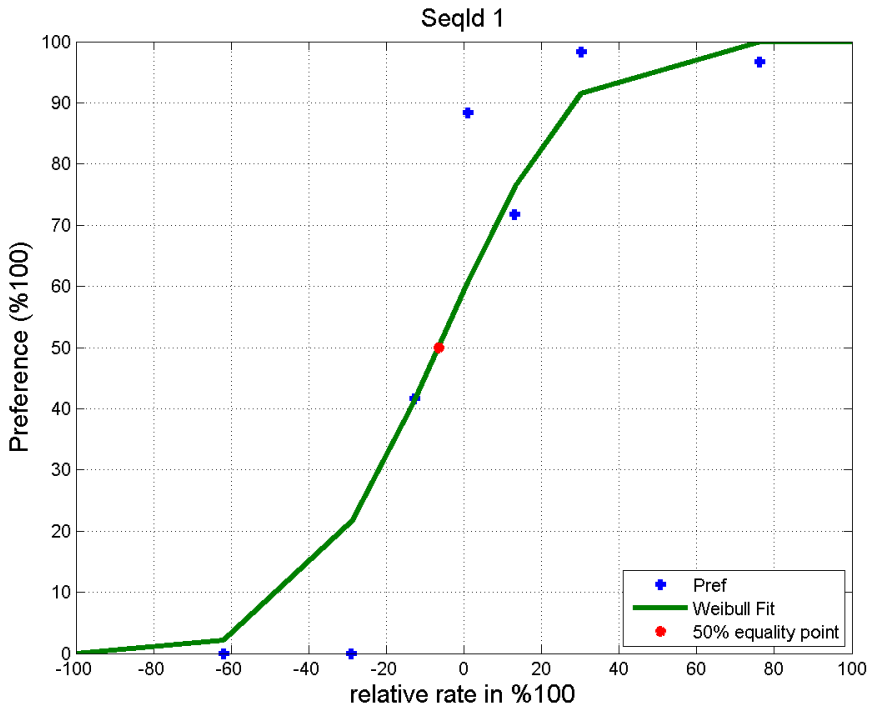


Figure 8.6 – Example of a psychometric preference function with Weibull fitting.

SeqId	1	2	3	8
QP (default)	43	47	42	42
Bitrate saving (%)	9.2	5.4	10.5	**17.7**

Table 8.1 – Relative bitrate saving.

8.4.4 Verification of the Proposed Approach

The results obtained so far, are specific for each texture type that was learned from the subjective experiment in section 8.3. In order to practically deploy such an approach, it must be verified that it works consistently with other textures belonging to same texture types. To do so, we sampled 4 new sequences (shown in Fig. 8.7), which are the most similar to ones in Table 8.2. Once more, the used feature to assess the similarity is the same as in section 8.3.2, namely HEVC rate-distortion behavior. These sequences were compressed using the perceptual optimization process used for the corresponding texture type. The corresponding bitrate saving, shown in Table 8.2, indicates clearly that the proposed approach is also valid for other sequences, sharing similar features.

Figure 8.7 – Sequences used for verification test.

SeqId	1	2	3	8
QP (default)	43	47	42	42
Bitrate saving (%)	6.5	12.3	**28.6**	7.5

Table 8.2 – Relative bitrate saving.

8.4.5 Testing Other Quality Levels

We were also interested in the performance of the proposed model on other quality points. After encoding the sequences with large set of compression levels, we manually selected three quality points corresponding to high, medium and low quality. This time, instead of using the forced choice method, we used UML to achieve faster and more accurate results.

The bitrate saving at the same subjective quality is shown in Table 8.3. One can clearly realize that the proposed perceptual optimization algorithm provides a significant bitrate saving, up to 37%.

SeqId	Q1	Q2	Q3	**Average(row)**
1	12.2+-7.4	6.8+-2.2	19.2+-1.4	12.7+-3,7
2	40.4+-1.3	34.9+-1.0	20.7+-0.9	32.0+-1.1
3	36.9+-4.6	**37.3**+-5.5	33.5+-6.02	35.9+-6.0
8	13.3+-5.9	26.9+-6.3	3.8-7.4	14.6+-6.5
Average(col)	25.7+-5.3	26.5+-3.7	19.3+-3.9	**23.8+-4.3**

Table 8.3 – Bitrate saving (%) due to perceptual optimization. +- refers to 95% confidence interval. Q1, Q2 and Q3 represent different quality points (High, medium and low resp.).

8.5 Generalization of the Proposed Approach

8.5.1 V1-E Features for Model Prediction

The results presented so far are based on optimizing sequences utilizing their measured perceived distortion profile. To generalize the perceptual optimization algorithm, the distortion model parameters need to be estimated from the sequences features. In other words, the piece-wise linear function parameters (values of α and β in section 8.3.3), must be learned from each sequence.

For this purpose, the proposed perceptual texture model (V1-E) was used as a features extractor (as in section 4.4). We used the same dimension as the one for sensitivity prediction (section 6.3), i.e. 8 energies from V1-E. This set of features was used in the form of linear regression. The performance has been evaluated by the mean squared error (normalized) of leave one out cross-validation test, which has a value of 0.01. This indicates that model prediction is reasonably good.

In the published work of [207], another set of features was used for this purpose. The set consists of the spatial and temporal information (SI and TI) [193] and the colorfulness (CF) [175], the homogeneity feature obtained from the gray-level co-occurrence matrix [176] and the curl and peakness of normal flow as defined in [94]. The prediction accuracy was less than the one obtained with V1-E, in the sense that the normalized mean squared error was 0.087, which is significantly higher than the one from V1-E (0.01).

8.5.2 Generalization Test

The trained linear regression model has been used to predict the perceptual distortion model parameters of novel sequences. As explained in section 8.3.2, we have an overall of 43 dynamic texture sequences, 8 of them where used in the first experiment. For the rest of sequences (33 sequences), the trained linear regression model was used to predict their perceptual distortion model.

Among these sequences, we selected the top 24 sequences having the highest deviation between the measured and the perceived (predicted) distortion. Examples of these sequences are shown in Fig. 8.8.

Using the same perceptual optimization algorithm described in section 8.4.1, we encoded these sequences also for 3 quality points (high, medium and low). The bitrate saving was measured subjectively, using the same psychophysical procedure as in section 8.4.2. The results are shown in Fig. 8.9, in which the average saving of the three quality points is plotted for each sequence, defined by its sequence id (SeqId). We can clearly see that the model can provide significant bitrate saving for the majority of the sequences. However, some exceptions are also present.

8.6 Conclusion

In this chapter, the proposed framework of perceptually optimizing the video coding system, based on rate-distortion optimization, was fully explored. To overcome the issues associated with the direct use of the perceptual similarity metrics, such as the one based on V1-E, this chapter presented a solution by mapping the simple metric, namely MSE, to a perceptual value obtained from psychophysical experiments.

First, a psycho-physical test, namely MLDS, was used to measure the perceived difference due to HEVC compression. This was used then to examine the relationship between the perceived difference and the computed difference (using MSE). For this purpose, a representative set of texture videos, belonging to different categories of rate-distortion behavior, of short temporal extent within the foveal vision, was used for the test. The results of this test reveal that there is generally two trends in the relation between perceived and computed difference, one with large deviation, and one with direct proportion. For first trend, a mapping between those differences is employed to yield a perceptual metric that is defined for each sequence, which is used inside the rate-distortion loop of the video coding system. The employment of such metric was verified with subjective testing and showed an improved rate-quality performance over the reference HEVC encoder (HM encoder).

The mapping between MSE and perceived difference is modeled as a piece-wise linear function. This function is unique for each texture. It has been shown that V1-E, as a features extractor, can be used to predict this function. Utilizing this, the function can be predicted for novel videos, and be used for the optimization purpose. Using 24 new videos, we were able to show a significant bitrate saving, at different quality points.

Figure 8.8 – Sequences used for the generalization test (from HomoTex dataset 5.2.1), with SeqId from 1 to 24 (from top left to bottom right).

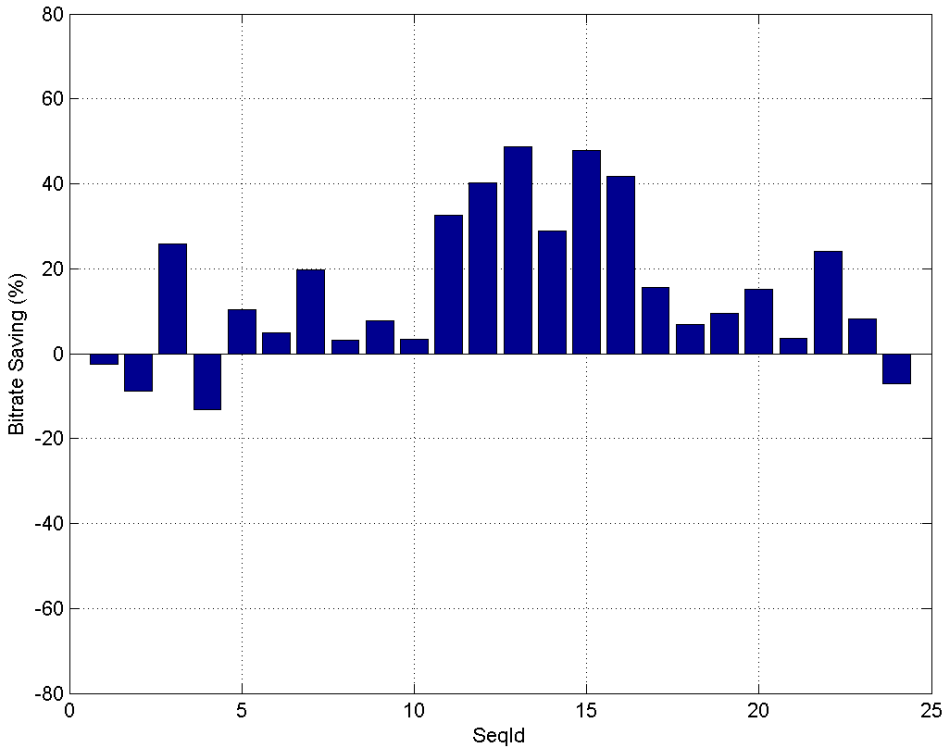


Figure 8.9 – Average bitrate saving for three quality points of the generalization test.

9

Conclusion and Outlook

9.1 Overall Summary

The thesis work has dealt with concept of visual similarity of textures. It reviewed most of studies on texture perception that has led to the development of texture similarity models. The importance of texture similarity is highlighted, and its use for the purpose of recognition and retrieval is thoroughly reviewed. As an important application, its employment for improving the image and video compression system is reviewed in details. By collecting all these studies, and analyzing them, this work has proposed a perceptual model of texture similarity that is generalized for both static and dynamic textures. The model has shown excellent performance in extensive testing scenarios. Utilizing this model for video compression has also shown a significant improvement over the state of the art video compression standard (HEVC).

In the first part of this thesis (Part I), a survey about texture perception was provided, and linked them to the existing models of texture similarity. The chapter focused on both static and dynamic textures. It was realized that there is a lack of universal definition of the textures, in the sense that static textures tends to be differently defined when compared to dynamic textures. In addition, a controversy was found in understanding what dynamic textures are. Accordingly, a proper definition of texture, inlcuding both static and dynamic textures, was given as:

A visual phenomenon, that covers both static and dynamic textures, where static textures refer to us as homogeneous regions of the scene that are typically composed of small elements (texels) arranged in a certain order, they might exhibit simple motion such as translation, rotation and zooming. On the other hand, dynamic textures

are textures that evolve over time, allowing both motion and deformation, with certain stationarity in space and time.

It was also observed that despite the extensive studies on static texture perception, dynamic textures are not yet well explored. This was due to the fact that most of the perceptual studies consider texture images rather than texture videos. This was the motivation to develop a perceptual model that is generalized for both types of textures.

The review about similarity models revealed that there exist generally 4 categories, with different degree of association to texture perception. The four categories are: Transform-, Auto-regressive-, Texton- and Motion-based modeling. The transform based modeling represents a direct link between texture perception and texture similarity. This is because of the fact that there is a similar transform occurring in the V1 cortical processing area. Texton based modeling is highly indirect, as it assumes that the similarity is due to the distribution of the texture element, and ignores the actual neural processing.

The use of texture similarity concepts in image and video compression is then overviewed. It was realized that most approaches consider indirect way of using texture similarity, in the sense that the similarity concepts are used for generating similar textures, by the means of texture synthesis techniques, instead of using the similarity models for improving the rate-similarity performance of the compression system. Three categories of synthesis based coding were identified, which are: texture removal, texture simplification and texture prediction. Texture removal approaches omit large part of the textures, and rely on texture synthesis for generating back the missing part. These approaches are commonly used as they can provide significant bitrate saving. This is because the omitted part is encoded only as a synthesis parameters, which are certainly much more easier to encode than original data. However, two obstacles preventing the deployment of such models were identified. First, significant changes in the coding standard are required which causes a change of the end-users' existing software/hardware. Second, high complexity would have to be added to the decoder side that might exceed the end-users' devices capacity.

To mitigate these problems, texture simplification approaches have been proposed. In these approaches, the synthesis is performed on the encoder side only, and the bitsteam can be decoded by any standard decoder. The local texture synthesis [10] was proposed. This algorithm replaces each block by a set of synthesized ones, when some statistical constraints are met. The encoder selects the synthetic block that minimizes the immediate rate-distortion cost. Typically, texture simplification approaches showed lesser bitrate saving, as compared to the texture removal approaches, because no omission of data is performed.

The lack of direct use of texture similarity was another motivation behind this work. We were interested in developing a perceptual model of texture similarity that is directly applicable to the optimization of image and video compression. Due to the several issues identified with violating the standard, we intended also to provide a solution that is fully

compatible with the current coding standard (HEVC).

In the second part (Part II), the proposed generalized model for texture similarity, named V1-E, is presented. The model covers both static and dynamic textures, and it can be considered as an extension of transform based similarity models with more in detailed modeling of the human visual system. Among the several stages of visual processing in the human visual system, the cortical processing stage is considered to be an active processing stage. This is because other levels exhibit a very marginal effect when compared to the cortical area. The proposed model relies on the argument that textures are neither defined by shapes nor structured motion, which led to the exclusion of higher level of analysis associated with shape perception (ventral stream) and motion perception (dorsal stream), and thus it only considers the initial processing unit, named as V1.

The neural processing in V1 can be generally modeled as bank of spatio-temporal band-pass filters. The exact parameters of these filters are not fully known. Thus, we relied on a recently introduced computational model in [60]. The proposed model considers only the energies, thus named V1-E, of sub-band signals as the features for texture similarity. This is because energies are actually assumed to be computed by the V1 complex cells to provide a shift invariant response. The computed energies are then fused to produce a similarity score for each subband, where the average is finally taken to yield the overall similarity.

Despite its simplicity, the model showed an excellent performance as both similarity metric and features extractor. Large scale evaluations are performed including both verification and validation tests, in which the verification test is mainly concerned about testing the performance within the design constraint of the model, while the validation test is a generalization test in order to assess the performance with general conditions. For the purpose of verification, a retrieval test is performed with a dataset (HomoTex) that is designed with the same constraints of the proposed model. In other words, it is compliant with the spatio-temporal resolution supported by V1-E. Excellent retrieval rate was achieved by this model, while it outperformed the well-known similarity model of LBP-TOP. The validation test was also performed in order to avoid the bias toward the model constraints, as well as to provide benchmarking results. In this test, the common recognition tasks were performed with the two datasets of UCLA and DynTex++. The results indicated that the V1-E has excellent performances, outperforming the state of the art method.

We also examined the performance of V1-E as a features extractor. The purpose is to verify whether the perceptual features provided by V1-E can be used to predict visual properties of textures, in link with video compression, that are revealed with a psychophysical experiment. As a first step in linking V1-E to video compression, a psychophysical procedure was employed to measure the amount of visual sensitivity of textures towards the perceived distortions due to HEVC compression. From the

experiment, two visual properties are defined: perceptual redundancies and perceptual distortion tolerance. The results showed that V1-E can be used to predict both the visual redundancies and distortion tolerance. It was also shown that exploiting the distortion sensitivity can provide a significant bitrate saving without altering the perceived quality. This approach was the first use of V1-E in a video compression task. However, it is a passive approach, because it is implemented outside the encoding processes. For this reason, we intended to propose an active framework for perceptually optimizing the compression of textures.

In the last part (Part III), the proposed perceptual optimization framework of video compression was described and implemented. It utilizes the developed the perceptual model of texture similarity, with the aim of improving the rate-similarity behavior of the state of the art video compression standard HEVC. The classical framework of rate-distortion optimization was tackled. The idea is to replace the default distortion metrics of HEVC (SATD and SSD) by perceptual ones, obtained from perceptual texture similarity models, and also modifying the corresponding Lagrangian multiplier. In other words, the prediction mode selection and the block splitting will be decided based on optimized rate and texture similarity, rather than rate and pixel similarity.

Initially, a test of the proposed framework with texture images was performed. For this, STSIM based dissimilarity metric was used. STSIM is conceptually very close to V1-E similarity metric. The visual effect is quite clear for high compression level, in which the overall similarity is higher when STSIM is employed, even though there exist large point-wise differences. The rate-distortion behavior was tested with another (dis-)similarity metrics based on Gabor decomposition, as typical PSNR metric is out of context for this type of distortions. The results showed that the proposed approach achieves higher similarity (as computed by the Gaussain similarity metric) to the original textures, as compared to the default HEVC, especially for high compression level. A side effect was observed, that the encoder behavior has dramatically changed. This is because STSIM is not a linear metric, and thus it does not possess the summability property. This resulted in having a metric preferred block size, and the encoder thus does not equally consider the other possible block sizes

To overcome this issue, a solution is developed that utilizes the interesting properties of pixel-wise similarity, and adds the perceptual aspect into it. This is done by mapping the simple metric, namely MSE, to a perceptual value obtained with psychophysical experiments. First, a proper psycho-physical test, named MLDS, was used to measure the perceived difference due to HEVC compression, which in turn used to reveal the relationship between the perceived difference, and the computed difference. The test was performed on a representative set of texture videos, belonging to different categories of rate-distortion behavior, of short temporal extent within the foveal vision. The results of this test showed that there is generally two trends in the relation perceived and computed difference, one with large deviation, and one with direct proportion. For first trend, a

mapping between those differences is employed to yield a perceptual metric, which is defined for each sequence, which is used inside the rate-distortion loop of the video coding system. The employment of such metric was verified with subjective testing and showed an improved rate-quality performance over the reference HEVC encoder (HM encoder). The mapping function MSE and perceived difference is modeled as a piece-wise linear function. It has been shown that V1-E, as a features extractor, can be used to predict this function. Utilizing this, the function can be predicted for novel videos, and be used for the optimization purpose. Using 24 new videos, we were able to show a significant bitrate saving, at different quality points.

9.2 Future work

9.2.1 Beyond V1 Energies

The proposed perceptual model of textures (V1-E) considers only the energies of the subbands as the features for computing texture similarity. This can be amended by other statistical measures. The straightforward enhancement could be obtained by considering the cross correlations between bands, in a way very similar to STSIM. STSIM considers the correlation between bands having similar spatial orientation or spatial frequency. In the case of V1-E, one needs to take into account the third dimension of time, which largely increases computational space, and thus it is currently avoided for the proposed model. Nevertheless, an investigation about the performance of this extension is left for possible future work.

9.2.2 Alternative Uses of Texture Similarity and Features in Video Coding

In this work, the texture similarity metric was used for quantifying the distortions inside the rate-distortion loop of the video encoder. Besides its successful performance, one can plausibly think of many other possibilities of utilizing it for better compression.

First, it can be used to decide on the amount of distortions within each texture region. This can be done in order to allow different amount of compression, depending on the region/block properties, such that the overall similarity level is maintained. An example of this is region-based quantization parameter (QP) assignment. Another way is perceptual preprocessing by simplifying regions, without passing below a given similarity level, such that they are simpler to be encoded.

Other than this, the texture similarity metrics can also be used in the synthesis based coding approaches. The main problem with these approaches is assessing the quality of the synthesized regions, in comparison to distortions due to compression, in order that the encoder can precisely choose whether to enable synthesis or not. Texture similarity

metrics could possibly be used in this purpose, as they are meant for measuring similarity in a way far from pixel-wise similarity. However, it is still challenging to have a metric that can faithfully measure different types of distortions, and yet tells which one is better. Other than that, texture synthesis is still far away from being developed for coding purposes, and most of the synthesis-based coding approaches utilize simple synthesis algorithms. A good synthesis algorithm, in turn, would usually require lots of parameters to be encoded, that can already cost more than the original data itself.

On the other hand, the features extracted by V1-E can also be utilized to improve the video compression system. As long as a strong set of texture features is provided, intelligent machine learning models can be used to enhance the coding performance. For example, the encoder parameters can be learning by analyzing these features to provide better coding results. Other than that, the encoder decisions can also be learned, such that no exhaustive search for the optimal decision is needed. This can lead to a huge reduction of the complexity of the coding system, which is significantly desired for many applications such as real time communications.

9.2.3 Beyond Texture Similarity

Understanding texture similarity can directly lead to revealing other visual mechanisms of the human visual system. It can also be considered as a tool for many other computer vision applications. V1-E, developed as a perceptual model for assessing texture similarity and extracting textural features, can be used for segmentation purposes. This can result in providing coherent regions, defined by their spatio-temporal properties. Such type of segmentation can be used to explain visual mechanisms, such as visual organization. Further possibilities would be to exploit this segmentation to explain the visual importance of regions, or to help in finding salient regions as well as following eye gazes.

Finally, a proper modeling of texture similarity should obviously result in proper synthesis of textures. This is clearly a reverse engineering problem, if we know why textures look similar; we can easily generate similar textures. However, this is not an easy problem. One issue is the reversibility of the signal processing operations, which is not achievable. For example, the spatio-temporal decomposition that is considered in V1-E cannot be inverted. In contrast, the human mental capability can synthesize textures from the set of examples seen in the daily life. This is of course within the cognitive level of vision, and out of the scope of this thesis.

HomoTex: The Complete Set of The Used Texture Videos

Figure A.1 – Thumbnails of HomoTex videos.

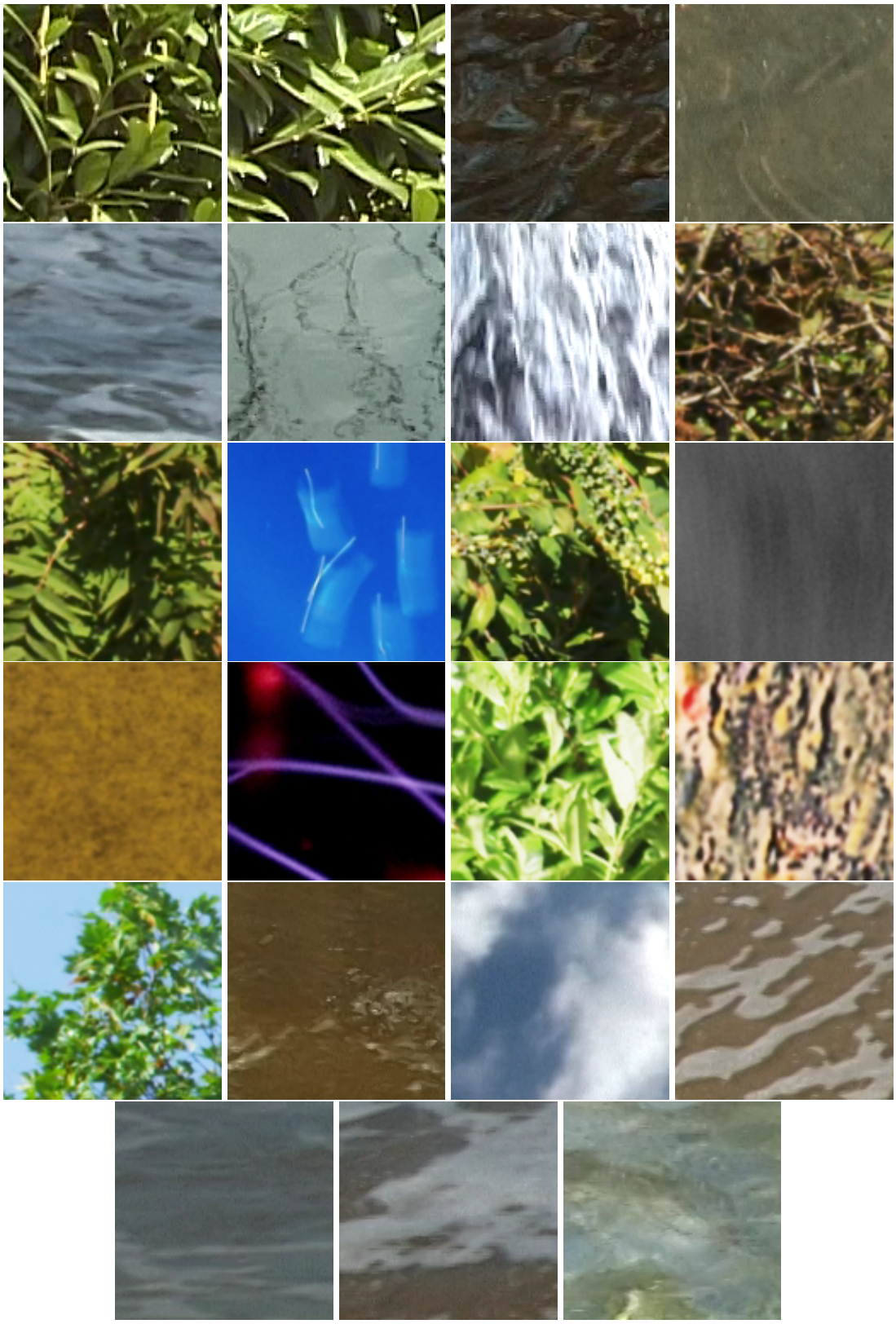


Figure A.1 – Thumbnails of HomoTex videos (cont.).

B

Publications and Dissemination

International Conferences

1. Naser, Karam, Vincent Ricordel, and Patrick Le Callet. *"Experimenting texture similarity metric STSIM for intra prediction mode selection and block partitioning in HEVC."* Digital Signal Processing (DSP), 2014 19th International Conference on. IEEE, 2014.
2. Naser, Karam, Vincent Ricordel, and Patrick Le Callet. *"Performance Analysis of Texture Similarity Metrics in HEVC Intra Prediction."* Video Processing and Quality Metrics for Consumer Electronics (VPQM). 2015.
3. Naser, Karam, Vincent Ricordel, and Patrick Le Callet. *Texture Similarity Metrics Applied to HEVC Intra Prediction."* The third Sino-French Workshop on Information and Communication Technologies, SIFWICT 2015.
4. Naser, Karam, Vincent Ricordel, and Patrick Le Callet. *"Local texture synthesis: A static texture coding algorithm fully compatible with HEVC."* Systems, Signals and Image Processing (IWSSIP), 2015 International Conference on. IEEE, 2015.
5. Naser, Karam, Vincent Ricordel, and Patrick Le Callet. *"Estimation of perceptual redundancies of HEVC encoded dynamic textures."* Quality of Multimedia Experience (QoMEX), 2016 Eighth International Conference on. IEEE, 2016.
6. Naser, Karam, Vincent Ricordel, and Patrick Le Callet. *"Modeling the perceptual distortion of dynamic textures and its application in HEVC."* Image Processing (ICIP), 2016 IEEE International Conference on. IEEE, 2016.

7. Naser, Karam, Vincent Ricordel, and Patrick Le Callet. *"A Foveated Short Term Distortion Model for Perceptually Optimized Dynamic Textures Compression in HEVC"*. Picture Coding Symposium (PCS) 2016
8. Thakur, Uday Singh, Karam Naser, and Mathias Wien. *"Dynamic Texture Synthesis Using Linear Phase Shift Interpolation."* Picture Coding Symposium (PCS) 2016
9. Ma, Chengyue, Karam Naser, Vincent Rircordel, Patrick Le Callet and Chunmei Qing *"An Adaptive Lagrange Multiplier Determination Method for Dynamic Texture in HEVC."* IEEE International Conference on Consumer Electronics China 2016

Book Chapters

1. Naser, Karam, Vincent Ricordel, and Patrick Le Callet. " Perceptual Texture Similarity for Machine Intelligence Applications." in Visual Content Indexing And Retrieval with Psycho-visual Models – Springer 2017

Patents

1. Naser, Karam, Vincent Ricordel, and Patrick Le Callet. "Method for Encoding Video Frames Based on Local Texture Synthesis and Corresponding Device" in France, Patent n° : 15306367.2 - 1908. 2015, European Patent Office (Munich, Germany)
2. Naser, Karam, Vincent Ricordel, and Patrick Le Callet. "Method of Compression of Dynamic Textures Based on the Exploitation of a Perceptual Distortion Model" US patent – on going

National Conferences

1. Journées Imagerie Optique Non Conventionnelle - 11 ème édition. GDR-ISIS in Nantes 2016.
2. Folle Journée imagerie Nantaise. Nantes 2017.

Dissemination

1. Thee Minute Thesis (3MT) Best Presentation Award. European Signal Processing Conference (EUSIPCO) 2016. Budapest, Hungary.

Bibliography

[1] Y. Zhai, D. L. Neuhoff, and T. N. Pappas, "Local radius index-a new texture similarity feature," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 1434–1438. 9, 35, 38

[2] J. Zujovic, T. N. Pappas, and D. L. Neuhoff, "Structural texture similarity metrics for image analysis and retrieval," *Image Processing, IEEE Transactions on*, vol. 22, no. 7, pp. 2545–2558, 2013. 9, 33, 38

[3] D. Tiwari and V. Tyagi, "Dynamic texture recognition based on completed volume local binary pattern," *Multidimensional Systems and Signal Processing*, vol. 27, no. 2, pp. 563–575, 2016. 9, 39, 78

[4] ——, "Dynamic texture recognition using multiresolution edge-weighted local structure pattern," *Computers & Electrical Engineering*, 2016. 9, 39, 77, 78, 79

[5] R. Péteri, S. Fazekas, and M. J. Huiskes, "Dyntex: A comprehensive database of dynamic textures," *Pattern Recognition Letters*, vol. 31, no. 12, pp. 1627–1632, 2010. 11, 23, 24, 38, 66, 69

[6] M. S. Landy, "Texture analysis and perception," *The new visual neurosciences, MIT Press, Cambridge, Mass*, pp. 639–652, 2013. 11, 22, 29

[7] C. J. Perry and M. Fallah, "Feature integration and object representations along the dorsal stream visual hierarchy," *Frontiers in computational neuroscience*, vol. 8, p. 84, 2014. 11, 32

[8] C. Rosewarne, B. Bross, M. Naccari, K. Sharman, and G. Sullivan, "High Efficiency Video Coding (HEVC) Test Model 16 (HM 16), Joint Collaborative Team on Video Coding (JCTVC) of ITU-T SG16 WP3 and ISO," IEC JTC1/SC29/WG11, Document JCTVC-P1002, Tech. Rep. 11, 43, 90, 102, 105, 128

[9] U. Thakur, K. Naser, and M. Wien, "Dynamic texture synthesis using linear phase shift interpolation," in *Proc. of International Picture Coding Symposium PCS '16*. Nuremberg, Germany: IEEE, Piscataway, Dec. 2016. 11, 44, 45, 46, 47

[10] K. Naser, V. Ricordel, and P. Le Callet, "Local texture synthesis: A static texture coding algorithm fully compatible with HEVC," in *Systems, Signals and Image*

Processing (IWSSIP), 2015 International Conference on. IEEE, 2015, pp. 37–40. 11, 45, 48, 50, 140

[11] M. N. Do and M. Vetterli, "Wavelet-based texture retrieval using generalized Gaussian density and Kullback-Leibler distance," *Image Processing, IEEE Transactions on*, vol. 11, no. 2, pp. 146–158, 2002. 12, 32, 68

[12] L. Song, X. Tang, W. Zhang, X. Yang, and P. Xia, "The sjtu 4k video sequence dataset," in *Quality of Multimedia Experience (QoMEX), 2013 Fifth International Workshop on*. IEEE, 2013, pp. 34–35. 13, 103

[13] "USC-SIPI Dataset," http://sipi.usc.edu/database, accessed: 2015-08-07. 13, 109, 110

[14] B. S. Manjunath and W.-Y. Ma, "Texture features for browsing and retrieval of image data," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 18, no. 8, pp. 837–842, 1996. 13, 32, 38, 59, 112, 114, 115, 116, 120

[15] M. S. Landy and N. Graham, "Visual perception of texture," *The visual neurosciences*, vol. 2, pp. 1106–1118, 2004. 22, 24, 29

[16] R. Rosenholtz, "Texture perception," *The Oxford handbook of perceptual organization*, 2014. 22, 29

[17] Tuceryan, Mihran and Jain, Anil K, "Texture analysis," *The handbook of pattern recognition and computer vision*, vol. 2, pp. 207–248, 1998. 22

[18] G. Doretto, A. Chiuso, Y. N. Wu, and S. Soatto, "Dynamic textures," *International Journal of Computer Vision*, vol. 51, no. 2, pp. 91–109, 2003. 22, 25, 34, 66, 72

[19] L. Liu, P. Fieguth, Y. Guo, X. Wang, and M. Pietikäinen, "Local binary features for texture classification: Taxonomy and experimental study," *Pattern Recognition*, vol. 62, pp. 135–160, 2017. 22

[20] "Oxford Dictionaries." [Online]. Available: http://www.oxforddictionaries.com 23

[21] H. Tamura, S. Mori, and T. Yamawaki, "Textural features corresponding to visual perception," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 8, no. 6, pp. 460–473, 1978. 24

[22] M. Amadasun and R. King, "Textural features corresponding to textural properties," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 19, no. 5, pp. 1264–1274, 1989. 24

[23] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 7, pp. 971–987, 2002. 24, 34

[24] G. Fan and X.-G. Xia, "Wavelet-based texture analysis and synthesis using hidden markov models," *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, vol. 50, no. 1, pp. 106–120, 2003. 24

[25] G. Caenen and L. Van Gool, "Maximum response filters for texture analysis," in *Computer Vision and Pattern Recognition Workshop, 2004. CVPRW'04. Conference on*. IEEE, 2004, pp. 58–58. 25

[26] J. A. Montoya-Zegarra, N. J. Leite, and R. da S Torres, "Rotation-invariant and scale-invariant steerable pyramid decomposition for texture image retrieval," in *Computer Graphics and Image Processing, 2007. SIBGRAPI 2007. XX Brazilian Symposium on*. IEEE, 2007, pp. 121–128. 25, 32

[27] J. Portilla and E. P. Simoncelli, "A parametric texture model based on joint statistics of complex wavelet coefficients," *International Journal of Computer Vision*, vol. 40, no. 1, pp. 49–70, 2000. 25, 32, 58

[28] V. Kwatra, I. Essa, A. Bobick, and N. Kwatra, "Texture optimization for example-based synthesis," in *ACM Transactions on Graphics (TOG)*, vol. 24, no. 3. ACM, 2005, pp. 795–802. 25

[29] L.-Y. Wei, S. Lefebvre, V. Kwatra, and G. Turk, "State of the art in example-based texture synthesis," in *Eurographics 2009, State of the Art Report, EG-STAR*. Eurographics Association, 2009, pp. 93–117. 25

[30] R. C. Nelson and R. Polana, "Qualitative recognition of motion using temporal texture," *CVGIP: Image understanding*, vol. 56, no. 1, pp. 78–89, 1992. 25, 36

[31] A. Rahman and M. Murshed, "A motion-based approach for temporal texture synthesis," in *TENCON 2005 2005 IEEE Region 10*. IEEE, 2005, pp. 1–4. 25

[32] W.-H. Chang, N.-C. Yang, C.-M. Kuo, Y.-J. Chen *et al.*, "An efficient temporal texture descriptor for video retrieval," in *Proceedings of the 6th WSEAS International Conference on Signal Processing, Computational Geometry & Artificial Vision*. World Scientific and Engineering Academy and Society (WSEAS), 2006, pp. 107–112. 25

[33] S. Soatto, G. Doretto, and Y. N. Wu, "Dynamic textures," in *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, vol. 2. IEEE, 2001, pp. 439–446. 25, 34, 72

[34] L. Wang, H. Liu, and F. Sun, "Dynamic texture classification using local fuzzy coding," in *Fuzzy Systems (FUZZ-IEEE), 2014 IEEE International Conference on*. IEEE, 2014, pp. 1559–1565. 25

[35] K. G. Derpanis and R. P. Wildes, "Dynamic texture recognition based on distributions of spacetime oriented structure," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 191–198. 25, 33

[36] D. Chetverikov and R. Péteri, "A brief survey of dynamic texture description and recognition," in *Computer Recognition Systems*. Springer, 2005, pp. 17–26. 25

[37] S. Dubois, R. Péteri, and M. Ménard, "A comparison of wavelet based spatio-temporal decomposition methods for dynamic texture recognition," in *Pattern Recognition and Image Analysis*. Springer, 2009, pp. 314–321. 25, 33

[38] K. G. Derpanis and R. P. Wildes, "Spacetime texture representation and recognition based on a spatiotemporal orientation analysis," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 34, no. 6, pp. 1193–1205, 2012. 25, 33, 77

[39] T. Crivelli, B. Cernuschi-Frias, P. Bouthemy, and J.-F. Yao, "Motion textures: Modeling, classification, and segmentation using mixed-state markov random fields," *SIAM Journal on Imaging Sciences*, vol. 6, no. 4, pp. 2484–2520, 2013. 26

[40] S. Valaeys, G. Menegaz, F. Ziliani, and J. Reichel, "Modeling of 2D+1 texture movies for video coding," *Image and Vision Computing*, vol. 21, no. 1, pp. 49–59, 2003. 26

[41] Y. Wang and S.-C. Zhu, "Modeling textured motion: Particle, wave and sketch," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*. IEEE, 2003, pp. 213–220. 26

[42] Y. Guo, G. Zhao, Z. Zhou, and M. Pietikainen, "Video texture synthesis with multi-frame lbp-top and diffeomorphic growth model," *Image Processing, IEEE Transactions on*, vol. 22, no. 10, pp. 3879–3891, 2013. 26, 38

[43] M. Bosch, F. Zhu, and E. J. Delp, "An overview of texture and motion based video coding at Purdue University," in *Picture Coding Symposium, 2009. PCS 2009*. IEEE, 2009, pp. 1–4. 26, 44

[44] X. Sun, B. Yin, and Y. Shi, "A low cost video coding scheme using texture synthesis," in *Image and Signal Processing, 2009. CISP'09. 2nd International Congress on*. IEEE, 2009, pp. 1–5. 26

[45] F. Zhang and D. R. Bull, "A parametric framework for video compression using region-based texture models," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 5, no. 7, pp. 1378–1392, 2011. 26, 44

[46] B. Julesz, "Visual pattern discrimination," *IRE transactions on Information Theory*, vol. 8, no. 2, pp. 84–92, 1962. 26

[47] B. Julész, E. Gilbert, L. Shepp, and H. Frisch, "Inability of humans to discriminate between visual textures that agree in second-order statistics-revisited," *Perception*, vol. 2, no. 4, pp. 391–405, 1973. 26, 27

[48] B. Julesz, E. Gilbert, and J. D. Victor, "Visual discrimination of textures with identical third-order statistics," *Biological Cybernetics*, vol. 31, no. 3, pp. 137–140, 1978. 27

[49] B. Julesz, "Textons, the elements of texture perception, and their interactions," *Nature*, vol. 290, no. 5802, pp. 91–97, 1981. 28

[50] J. Beck, "Textural segmentation, second-order statistics, and textural elements," *Biological Cybernetics*, vol. 48, no. 2, pp. 125–130, 1983. 28

[51] D. H. Hubel and T. N. Wiesel, "Receptive fields and functional architecture of monkey striate cortex," *The Journal of physiology*, vol. 195, no. 1, pp. 215–243, 1968. 28, 57, 59

[52] D. A. Pollen and S. F. Ronner, "Visual cortical neurons as localized spatial frequency filters," *Systems, Man and Cybernetics, IEEE Transactions on*, no. 5, pp. 907–916, 1983. 28

[53] M. R. Turner, "Texture discrimination by gabor functions," *Biological cybernetics*, vol. 55, no. 2-3, pp. 71–82, 1986. 28

[54] J. Ontrup, H. Wersing, and H. Ritter, "A computational feature binding model of human texture perception," *Cognitive Processing*, vol. 5, no. 1, pp. 31–44, 2004. 28

[55] J. Malik and P. Perona, "Preattentive texture discrimination with early vision mechanisms," *JOSA A*, vol. 7, no. 5, pp. 923–932, 1990. 28, 59

[56] K. Grill-Spector and R. Malach, "The human visual cortex," *Annu. Rev. Neurosci.*, vol. 27, pp. 649–677, 2004. 30, 31

[57] E. P. Simoncelli and D. J. Heeger, "A model of neuronal responses in visual area MT," *Vision research*, vol. 38, no. 5, pp. 743–761, 1998. 30, 31, 59

[58] S. Nishimoto and J. L. Gallant, "A three-dimensional spatiotemporal receptive field model explains responses of area MT neurons to naturalistic movies," *The Journal of Neuroscience*, vol. 31, no. 41, pp. 14 551–14 564, 2011. 30, 31, 59

[59] E. Tlapale, P. Kornprobst, G. S. Masson, and O. Faugeras, "A neural field model for motion estimation," in *Mathematical image processing*. Springer, 2011, pp. 159–179. 30

[60] S. V. David, W. E. Vinje, and J. L. Gallant, "Natural stimulus statistics alter the receptive field structure of V1 neurons," *The Journal of Neuroscience*, vol. 24, no. 31, pp. 6991–7006, 2004. 30

[61] J. A. Perrone, "A visual motion sensor based on the properties of V1 and MT neurons," *Vision research*, vol. 44, no. 15, pp. 1733–1755, 2004. 31

[62] N. C. Rust, V. Mante, E. P. Simoncelli, and J. A. Movshon, "How MT cells analyze the motion of visual patterns," *Nature neuroscience*, vol. 9, no. 11, pp. 1421–1431, 2006. 31

[63] D. C. Bradley and M. S. Goyal, "Velocity computation in the primate visual system," *Nature Reviews Neuroscience*, vol. 9, no. 9, pp. 686–695, 2008. 31

[64] F. Solari, M. Chessa, N. K. Medathati, and P. Kornprobst, "What can we expect from a V1-MT feedforward architecture for optical flow estimation?" *Signal Processing: Image Communication*, vol. 39, pp. 342–354, 2015. 31, 59, 64, 141

[65] N. K. Medathati, M. Chessa, G. Masson, P. Kornprobst, and F. Solari, "Decoding MT Motion Response for Optical Flow Estimation: An Experimental Evaluation," Ph.D. dissertation, INRIA Sophia-Antipolis, France; University of Genoa, Genoa, Italy; INT la Timone, Marseille, France; INRIA, 2015. 31

[66] M. Chessa, S. P. Sabatini, and F. Solari, "A systematic analysis of a V1–MT neural model for motion estimation," *Neurocomputing*, vol. 173, pp. 1811–1823, 2016. 31, 60

[67] C. Pack, S. Grossberg, and E. Mingolla, "A neural model of smooth pursuit control and motion perception by cortical area mst," *Journal of Cognitive Neuroscience*, vol. 13, no. 1, pp. 102–120, 2001. 31

[68] S. Grossberg, E. Mingolla, and C. Pack, "A neural model of motion processing and visual navigation by cortical area mst," *Cerebral Cortex*, vol. 9, no. 8, pp. 878–895, 1999. 31

[69] M. N. Do and M. Vetterli, "Texture similarity measurement using Kullback-Leibler distance on wavelet subbands," in *Image Processing, 2000. Proceedings. 2000 International Conference on*, vol. 3. IEEE, 2000, pp. 730–733. 32, 38

[70] E. P. Simoncelli, W. T. Freeman, E. H. Adelson, and D. J. Heeger, "Shiftable multiscale transforms," *Information Theory, IEEE Transactions on*, vol. 38, no. 2, pp. 587–607, 1992. 32, 59, 107

[71] X. Zhao, M. G. Reyes, T. N. Pappas, and D. L. Neuhoff, "Structural texture similarity metrics for retrieval applications," in *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*. IEEE, 2008, pp. 1196–1199. 33, 63, 107

[72] J. Zujovic, T. N. Pappas, and D. L. Neuhoff, "Structural similarity metrics for texture analysis and retrieval," in *Image Processing (ICIP), 2009 16th IEEE International Conference on*. IEEE, 2009, pp. 2225–2228. 33, 58

[73] M. Maggioni, G. Jin, A. Foi, and T. N. Pappas, "Structural texture similarity metric based on intra-class variances," in *2014 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2014, pp. 1992–1996. 33

[74] J. R. Smith, C.-Y. Lin, and M. Naphade, "Video texture indexing using spatio-temporal wavelets," in *Image Processing. 2002. Proceedings. 2002 International Conference on*, vol. 2. IEEE, 2002, pp. II–437. 33

[75] W. N. Gonçalves, B. B. Machado, and O. M. Bruno, "Spatiotemporal gabor filters: a new method for dynamic texture recognition," *arXiv preprint arXiv:1201.3612*, 2012. 33

[76] K. G. Derpanis, M. Sizintsev, K. J. Cannons, and R. P. Wildes, "Action spotting and recognition based on a spatiotemporal orientation analysis," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 35, no. 3, pp. 527–540, 2013. 33

[77] Z. Bao, C. Xu, and C. Wang, "Perceptual auto-regressive texture synthesis for video coding," *Multimedia tools and applications*, vol. 64, no. 3, pp. 535–547, 2013. 33

[78] N. Campbell, C. Dalton, D. Gibson, D. Oziem, and B. Thomas, "Practical generation of video textures using the auto-regressive process," *Image and Vision Computing*, vol. 22, no. 10, pp. 819–827, 2004. 33

[79] A. Khandelia, S. Gorecha, B. Lall, S. Chaudhury, and M. Mathur, "Parametric video compression scheme using AR based texture synthesis," in *Computer Vision, Graphics & Image Processing, 2008. ICVGIP'08. Sixth Indian Conference on*. IEEE, 2008, pp. 219–225. 33

[80] G. Doretto and S. Soatto, "Modeling dynamic scenes: An overview of dynamic textures," in *Handbook of Mathematical Models in Computer Vision*. Springer, 2006, pp. 341–355. 34

[81] ——, "Editable dynamic textures," in *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, vol. 2. IEEE, 2003, pp. II–137. 34, 72

[82] B. Abraham, O. I. Camps, and M. Sznaier, "Dynamic texture with fourier descriptors," in *Proceedings of the 4th nternational Workshop on Texture Analysisand Synthesis*, 2005, pp. 53–58. 34

[83] Y. Li, T. Wang, and H.-Y. Shum, "Motion texture: a two-level statistical model for character motion synthesis," in *ACM Transactions on Graphics (ToG)*, vol. 21, no. 3. ACM, 2002, pp. 465–472. 34

[84] R. Costantini, L. Sbaiz, and S. Süsstrunk, "Higher order SVD analysis for dynamic texture synthesis," *Image Processing, IEEE Transactions on*, vol. 17, no. 1, pp. 42–52, 2008. 34

[85] L. Yuan, F. Wen, C. Liu, and H.-Y. Shum, "Synthesizing dynamic texture with closed-loop linear dynamic system," in *Computer Vision-ECCV 2004*. Springer, 2004, pp. 603–616. 34

[86] P. Ghadekar and N. Chopade, "Nonlinear dynamic texture analysis and synthesis model," *Int. J. of Recent Trends in Engineering & Technology*, vol. 11, 2014. 34

[87] G. Zhao and M. Pietikainen, "Dynamic texture recognition using local binary patterns with an application to facial expressions," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 29, no. 6, pp. 915–928, 2007. 34, 70, 73, 78, 79

[88] Y. Zhai and D. L. Neuhoff, "Rotation-invariant local radius index: A compact texture similarity feature for classification," in *2014 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2014, pp. 5711–5715. 35, 70

[89] J. Chen, S. Shan, C. He, G. Zhao, M. Pietikainen, X. Chen, and W. Gao, "WLD: A robust local image descriptor," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 9, pp. 1705–1720, 2010. 35

[90] D.-C. He and L. Wang, "Texture unit, texture spectrum, and texture analysis," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 28, no. 4, pp. 509–512, 1990. 35

[91] A. Barcelo, E. Montseny, and P. Sobrevilla, "Fuzzy texture unit and fuzzy texture spectrum for texture characterization," *Fuzzy Sets and Systems*, vol. 158, no. 3, pp. 239–252, 2007. 35

[92] D.-C. He and L. Wang, "Simplified texture spectrum for texture analysis," *Journal of Communication and Computer*, vol. 7, no. 8, pp. 44–53, 2010. 35

[93] X. Liu and D. Wang, "A spectral histogram model for texton modeling and texture discrimination," *Vision Research*, vol. 42, no. 23, pp. 2617–2634, 2002. 35

[94] L. van der Maaten and E. Postma, "Texton-based texture classification," in *Proceedings of Belgium-Netherlands Artificial Intelligence Conference*, 2007. 35

[95] C.-H. Peh and L.-F. Cheong, "Synergizing spatial and temporal texture," *Image Processing, IEEE Transactions on*, vol. 11, no. 10, pp. 1179–1191, 2002. 36

[96] R. Péteri and D. Chetverikov, "Dynamic texture recognition using normal flow and texture regularity," in *Pattern Recognition and Image Analysis*. Springer, 2005, pp. 223–230. 36, 93, 135

[97] S. Fazekas and D. Chetverikov, "Dynamic texture recognition using optical flow features and temporal periodicity," in *Content-Based Multimedia Indexing, 2007. CBMI'07. International Workshop on*. IEEE, 2007, pp. 25–32. 36

[98] A. Rahman and M. Murshed, "Real-time temporal texture characterisation using block-based motion co-occurrence statistics," in *International Conference on Image Processing*, 2004. 36

[99] T. Amiaz, S. Fazekas, D. Chetverikov, and N. Kiryati, "Detecting regions of dynamic texture," in *Scale Space and Variational Methods in Computer Vision*. Springer, 2007, pp. 848–859. 36

[100] S. Fazekas, T. Amiaz, D. Chetverikov, and N. Kiryati, "Dynamic texture detection based on motion analysis," *International journal of computer vision*, vol. 82, no. 1, pp. 48–63, 2009. 36

[101] Y. Xu, Y. Quan, H. Ling, and H. Ji, "Dynamic texture classification using dynamic fractal analysis," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1219–1226. 37, 78, 79

[102] Y. Xu, Y. Quan, Z. Zhang, H. Ling, and H. Ji, "Classifying dynamic textures via spatiotemporal fractal analysis," *Pattern Recognition*, vol. 48, no. 10, pp. 3239–3248, 2015. 37

[103] Y. Xu, S. Huang, H. Ji, and C. Fermüller, "Scale-space texture description on sift-like textons," *Computer Vision and Image Understanding*, vol. 116, no. 9, pp. 999–1013, 2012. 37, 78, 79

[104] W. N. Goncalves and O. M. Bruno, "Dynamic texture analysis and segmentation using deterministic partially self-avoiding walks," *Expert Systems with Applications*, vol. 40, no. 11, pp. 4283–4300, 2013. 37

[105] ——, "Dynamic texture segmentation based on deterministic partially self-avoiding walks," *Computer Vision and Image Understanding*, vol. 117, no. 9, pp. 1163–1174, 2013. 37

[106] K. Dimitropoulos, P. Barmpoutis, and N. Grammalidis, "Spatio-temporal flame modeling and dynamic texture analysis for automatic video-based fire detection," 2014. 37

[107] P. Barmpoutis, K. Dimitropoulos, and N. Grammalidis, "Smoke detection using spatio-temporal analysis, motion modeling and dynamic texture recognition," in *Signal Processing Conference (EUSIPCO), 2013 Proceedings of the 22nd European*. IEEE, 2014, pp. 1078–1082. 37

[108] R. Narain, V. Kwatra, H.-P. Lee, T. Kim, M. Carlson, and M. C. Lin, "Feature-guided dynamic texture synthesis on continuous flows," in *Proceedings of the 18th Eurographics conference on Rendering Techniques*. Eurographics Association, 2007, pp. 361–370. 37

[109] K. J. B. S. S. H. D. Siddalinga Swamy, D. M. Chandler, "Parametric quality assessment of synthesized textures," *Proc. Human Vision and Electronic Imaging*, 2011. 38

[110] S. Varadarajan and L. J. Karam, "Adaptive texture synthesis based on perceived texture regularity," in *Quality of Multimedia Experience (QoMEX), 2014 Sixth International Workshop on*. IEEE, 2014, pp. 76–80. 38

[111] J. Ballé, "Subjective evaluation of texture similarity metrics for compression applications," in *Picture Coding Symposium (PCS), 2012*. IEEE, 2012, pp. 241–244. 38

[112] J. Zujovic, T. N. Pappas, D. L. Neuhoff, R. van Egmond, and H. de Ridder, "Subjective and objective texture similarity for image compression," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2012, pp. 1369–1372. 38

[113] P. Saisan, G. Doretto, Y. N. Wu, and S. Soatto, "Dynamic texture recognition," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of*

the 2001 IEEE Computer Society Conference on, vol. 2. IEEE, 2001, pp. II–58. 38, 77

[114] B. Ghanem and N. Ahuja, "Maximum margin distance learning for dynamic texture recognition," in *European Conference on Computer Vision*. Springer, 2010, pp. 223–236. 38, 66, 73, 79

[115] D. Tiwari and V. Tyagi, "Improved Weber's law based local binary pattern for dynamic texture recognition," *Multimedia Tools and Applications*, pp. 1–18, 2016. 39, 79

[116] G. J. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 22, no. 12, pp. 1649–1668, 2012. 42, 84

[117] J.-R. Ohm, G. J. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand, "Comparison of the coding efficiency of video coding standards—including high efficiency video coding (HEVC)," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 22, no. 12, pp. 1669–1684, 2012. 42

[118] D. Mukherjee, H. Su, J. Bankoski, A. Converse, J. Han, Z. Liu, and Y. Xu, "An overview of new video coding tools under consideration for VP10: the successor to VP9," in *SPIE Optical Engineering+ Applications*. International Society for Optics and Photonics, 2015, pp. 95 991E–95 991E. 42

[119] D. Grois, D. Marpe, A. Mulayoff, B. Itzhaky, and O. Hadar, "Performance comparison of H.265/MPEG-HEVC, VP9, and H.264/MPEG-AVC encoders," in *Picture Coding Symposium (PCS), 2013*. IEEE, 2013, pp. 394–397. 43

[120] P. Ndjiki-Nya, B. Makai, G. Blattermann, A. Smolic, H. Schwarz, and T. Wiegand, "Improved H.264/AVC coding using texture analysis and synthesis," in *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*, vol. 3. IEEE, 2003, pp. III–849. 44

[121] P. Ndjiki-Nya and T. Wiegand, "Video coding using texture analysis and synthesis," in *Proc Picture Coding Symp. Saint-Malo, France*, 2003. 44

[122] G. J. Sullivan, P. N. Topiwala, and A. Luthra, "The H.264/AVC advanced video coding standard: Overview and introduction to the fidelity range extensions," in *Optical Science and Technology, the SPIE 49th Annual Meeting*. International Society for Optics and Photonics, 2004, pp. 454–474. 44, 102

[123] P. Ndjiki-Nya, T. Hinz, A. Smolic, and T. Wiegand, "A generic and automatic content-based approach for improved H.264/MPEG4-AVC video coding," in *Image Processing, 2005. ICIP 2005. IEEE International Conference on*, vol. 2. IEEE, 2005, pp. II–874. 44

[124] P. Ndjiki-Nya, D. Bull, and T. Wiegand, "Perception-oriented video coding based on texture analysis and synthesis," in *Image Processing (ICIP), 2009 16th IEEE International Conference on*. IEEE, 2009, pp. 2273–2276. 44

[125] U. S. Thakur and B. Ray, "Image coding using parametric texture synthesis," in *2016 IEEE 18th International Workshop on Multimedia Signal Processing (MMSP)*, Sept 2016, pp. 1–6. 44

[126] O. Chubach, P. Garus, and M. Wien, "Motion-based analysis and synthesis of dynamic textures," in *Proc. of International Picture Coding Symposium PCS '16*. Nuremberg, Germany: IEEE, Piscataway, Dec. 2016. 44

[127] A. Dumitras and B. G. Haskell, "A texture replacement method at the encoder for bit-rate reduction of compressed video," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 13, no. 2, pp. 163–175, 2003. 45, 48

[128] J. Ballé and M. Wien, "Extended texture prediction for H.264/AVC intra coding," in *Image Processing, 2007. ICIP 2007. IEEE International Conference on*, vol. 6. IEEE, 2007, pp. VI–93. 46

[129] A. Stojanovic, M. Wien, and J.-R. Ohm, "Dynamic texture synthesis for H.264/AVC inter coding," in *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*. IEEE, 2008, pp. 1608–1611. 46

[130] C.-W. Wong, O. C. Au, B. Meng, and H.-K. Lam, "Perceptual rate control for low-delay video communications," in *Multimedia and Expo, 2003. ICME'03. Proceedings. 2003 International Conference on*, vol. 3. IEEE, 2003, pp. III–361. 47

[131] C. Sun, H.-J. Wang, H. Li, and T.-h. Kim, "Perceptually adaptive Lagrange multiplier for rate-distortion optimization in H.264," in *Future Generation Communication and Networking (FGCN 2007)*, vol. 1. IEEE, 2007, pp. 459–463. 47

[132] H. Yu, F. Pan, Z. Lin, and Y. Sun, "A perceptual bit allocation scheme for H.264," in *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*. IEEE, 2005, pp. 4–pp. 47

[133] M. Liu and L. Lu, "An Improved Rate Control Algorithm of H.264/AVC Based on Human Visual System," in *Computer, Informatics, Cybernetics and Applications*. Springer, 2012, pp. 1145–1151. 47

[134] L. Xu, W. Lin, L. Ma, Y. Zhang, Y. Fang, K. N. Ngan, S. Li, and Y. Yan, "Free-energy principle inspired video quality metric and its use in video coding," 2016. 47, 106

[135] H. Hadizadeh, "Visual saliency in video compression and transmission," Ph.D. dissertation, Applied Sciences: School of Engineering Science, 2013. 47

[136] H. Hadizadeh and I. V. Bajic, "Saliency-aware video compression," *Image Processing, IEEE Transactions on*, vol. 23, no. 1, pp. 19–33, 2014. 47, 106

[137] K. Naser, V. Ricordel, and P. Le Callet, "Estimation of perceptual redundancies of HEVC encoded dynamic textures," in *Quality of Multimedia Experience*

(QoMEX), 2016 Eighth International Conference on. IEEE, 2016, pp. 1–5. 47, 93, 94

[138] C. Ma, K. Naser, V. Ricordel, P. L. Callet, and C. Qing1, "An Adaptive Lagrange Multiplier Determination Method for Dynamic Texture in HEVC," in *IEEE International Conference on Consumer Electronics China*. IEEE, 2016. 47

[139] V. Bruce, P. R. Green, and M. A. Georgeson, *Visual perception: Physiology, psychology, & ecology*. Psychology Press, 2003. 56

[140] G. Mather, *Foundations of sensation and perception*. Psychology Press, 2009. 56

[141] J. Hérault, *Vision: Images, Signals and Neural Networks: Models of Neural Processing in Visual Perception*. World Scientific, 2010, vol. 19. 56

[142] R. Young, "The gaussian derivative theory of spatial vision: Analysis of cortical cell receptive field line-weighting profiles. publication gmr-4920, general motors research labs," *Computer Science Dept*, vol. 30500, pp. 48 090–9055. 59

[143] A. Parker and M. Hawken, "Two-dimensional spatial structure of receptive fields in monkey striate cortex," *JOSA A*, vol. 5, no. 4, pp. 598–605, 1988. 59

[144] A. B. Watson, "The cortex transform: rapid computation of simulated neural images," *Computer vision, graphics, and image processing*, vol. 39, no. 3, pp. 311–327, 1987. 59

[145] K. Foster, J. P. Gaska, M. Nagler, and D. Pollen, "Spatial and temporal frequency selectivity of neurones in visual cortical areas V1 and V2 of the macaque monkey." *The Journal of physiology*, vol. 365, no. 1, pp. 331–363, 1985. 59

[146] M. Hawken, R. Shapley, and D. Grosof, "Temporal-frequency selectivity in monkey visual cortex," *Visual neuroscience*, vol. 13, no. 3, pp. 477–492, 1996. 59

[147] C. C. Pack and R. T. Born, "Temporal dynamics of a neural solution to the aperture problem in visual area MT of macaque brain," *Nature*, vol. 409, no. 6823, pp. 1040–1042, 2001. 62

[148] A. B. Chan and N. Vasconcelos, "Probabilistic kernels for the classification of auto-regressive visual processes," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 846–851. 66, 73

[149] M. A. Papadopoulos, F. Zhang, D. Agrafiotis, and D. Bull, "A video texture database for perceptual compression and quality assessment," in *Image Processing (ICIP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 2781–2785. 69

[150] S. Zhang, H. Yao, and S. Liu, "Dynamic background modeling and subtraction using spatio-temporal local binary patterns," in *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*. IEEE, 2008, pp. 1556–1559. 70

[151] D. Tiwari and V. Tyagi, "A novel scheme based on local binary pattern for dynamic texture recognition," *Comput. Vis. Image Underst.*, vol. 150, no. C, pp. 58–65, Sep. 2016. [Online]. Available: https://doi.org/10.1016/j.cviu.2016.04.010 77, 78, 79

[152] A. B. Chan and N. Vasconcelos, "Classifying video with kernel dynamic textures," in *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*. IEEE, 2007, pp. 1–6. 78

[153] J. Ren, X. Jiang, and J. Yuan, "Dynamic texture recognition using enhanced lbp features." in *ICASSP*, 2013, pp. 2400–2404. 79

[154] J. Ren, X. Jiang, J. Yuan, and G. Wang, "Optimizing lbp structure for visual recognition using binary quadratic programming," *IEEE Signal Processing Letters*, vol. 21, no. 11, pp. 1346–1350, 2014. 79

[155] M. Baktashmotlagh, M. Harandi, B. C. Lovell, and M. Salzmann, "Discriminative non-linear stationary subspace analysis for video classification," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 12, pp. 2353–2366, 2014. 79

[156] Y. Wang and S. Hu, "Chaotic features for dynamic textures recognition," *Soft Computing*, vol. 20, no. 5, pp. 1977–1989, 2016. 79

[157] S. R. Arashloo and J. Kittler, "Dynamic texture recognition using multiscale binarized statistical image features," *IEEE Transactions on Multimedia*, vol. 16, no. 8, pp. 2099–2109, 2014. 79

[158] Y. Wang and S. Hu, "Exploiting high level feature for dynamic textures recognition," *Neurocomputing*, vol. 154, pp. 217–224, 2015. 79

[159] A. R. Rivera and O. Chae, "Spatiotemporal directional number transitional graph for dynamic texture recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 10, pp. 2146–2152, 2015. 79

[160] A. B. Watson, "DCTune: A technique for visual optimization of DCT quantization matrices for individual images," in *Sid International Symposium Digest of Technical Papers*, vol. 24. SOCIETY FOR INFORMATION DISPLAY, 1993, pp. 946–946. 84

[161] Z. Wei and K. N. Ngan, "Spatio-temporal just noticeable distortion profile for grey scale image/video in DCT domain," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 19, no. 3, pp. 337–346, 2009. 84

[162] H. Wu and D. Tan, "Subjective and objective picture assessment at suprathreshold levels," in *Picture Coding Symposium (PCS), 2015*. IEEE, 2015, pp. 312–316. 84

[163] S. A. Klein, "Measuring, estimating, and understanding the psychometric function: A commentary," *Perception & psychophysics*, vol. 63, no. 8, pp. 1421–1455, 2001. 85

[164] F. A. Wichmann and N. J. Hill, "The psychometric function: I. fitting, sampling, and goodness of fit," *Perception & psychophysics*, vol. 63, no. 8, pp. 1293–1313, 2001. 85

[165] P. G. Engeldrum, *Psychometric scaling: a toolkit for imaging systems development*. Imcotek press, 2000. 85

[166] M. Taylor and C. D. Creelman, "PEST: Efficient estimates on probability functions," *The Journal of the Acoustical Society of America*, vol. 41, no. 4A, pp. 782–787, 1967. 86

[167] J. Findlay, "Estimates on probability functions: A more virulent PEST," *Attention, Perception, & Psychophysics*, vol. 23, no. 2, pp. 181–185, 1978. 86

[168] A. Pentland, "Maximum likelihood estimation: The best PEST," *Attention, Perception, & Psychophysics*, vol. 28, no. 4, pp. 377–379, 1980. 86

[169] A. B. Watson and D. G. Pelli, "Quest: A bayesian adaptive psychometric method," *Perception & psychophysics*, vol. 33, no. 2, pp. 113–120, 1983. 86

[170] P. E. King-Smith, S. S. Grigsby, A. J. Vingrys, S. C. Benes, and A. Supowit, "Efficient and unbiased modifications of the quest threshold method: theory, simulations, experimental evaluation and practical implementation," *Vision research*, vol. 34, no. 7, pp. 885–912, 1994. 86

[171] M. R. Leek, "Adaptive procedures in psychophysical research," *Perception & psychophysics*, vol. 63, no. 8, pp. 1279–1292, 2001. 86

[172] B. Treutwein, "Adaptive psychophysical procedures," *Vision research*, vol. 35, no. 17, pp. 2503–2522, 1995. 86

[173] Y. Shen and V. M. Richards, "A maximum-likelihood procedure for estimating psychometric functions: Thresholds, slopes, and lapses of attention," *The Journal of the Acoustical Society of America*, vol. 132, no. 2, pp. 957–967, 2012. 86

[174] Y. Shen, W. Dai, and V. M. Richards, "A MATLAB toolbox for the efficient estimation of the psychometric function using the updated maximum-likelihood adaptive procedure," *Behavior research methods*, vol. 47, no. 1, pp. 13–26, 2015. 86

[175] I. Rec, "Bt. 500-11,"," *Methodology for the subjective assessment of the quality of television pictures*, vol. 22, pp. 25–34, 2002. 87, 126

[176] T. Installations and L. Line, "Subjective video quality assessment methods for multimedia applications," *Networks*, vol. 910, p. 37, 1999. 93

[177] S. Winkler, "Analysis of public image and video databases for quality assessment," *IEEE Journal of Selected Topics in Signal Processing*, vol. 6, no. 6, pp. 616–625, 2012. 93, 135

[178] R. M. Haralick, K. Shanmugam, and I. H. Dinstein, "Textural features for image classification," *Systems, Man and Cybernetics, IEEE Transactions on*, no. 6, pp. 610–621, 1973. 93, 135

[179] T. Wiegand and B. Girod, "Lagrange multiplier selection in hybrid video coder control," in *Image Processing, 2001. Proceedings. 2001 International Conference on*, vol. 3. IEEE, 2001, pp. 542–545. 105

[180] Y.-H. Huang, T.-S. Ou, P.-Y. Su, and H. H. Chen, "Perceptual rate-distortion optimization using structural similarity index as quality metric," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 11, pp. 1614–1624, 2010. 106

[181] S. Wang, A. Rehman, Z. Wang, S. Ma, and W. Gao, "SSIM-motivated rate-distortion optimization for video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 4, pp. 516–529, 2012. 106

[182] H. H. Chen, Y.-H. Huang, P.-Y. Su, and T.-S. Ou, "Improving video coding quality by perceptual rate-distortion optimization," in *Multimedia and Expo (ICME), 2010 IEEE International Conference on*. IEEE, 2010, pp. 1287–1292. 106

[183] C.-W. Tang, C.-H. Chen, Y.-H. Yu, and C.-J. Tsai, "Visual sensitivity guided bit allocation for video coding," *Multimedia, IEEE Transactions on*, vol. 8, no. 1, pp. 11–18, 2006. 106

[184] L. Shen, Z. Liu, and Z. Zhang, "A novel H.264 rate control algorithm with consideration of visual attention," *Multimedia tools and applications*, vol. 63, no. 3, pp. 709–727, 2013. 106

[185] G.-x. Lin and S.-b. Zheng, "Perceptual importance analysis for H.264/AVC bit allocation," *Journal of Zhejiang University SCIENCE A*, vol. 9, no. 2, pp. 225–231, 2008. 106

[186] Joint Collaborative Team on Video Coding (JCT-VC) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG, "HEVC test model 9.0," Tech. Rep. 109

[187] K. Castleman, M. Schulze, and Q. Wu, "Simplified design of steerable pyramid filters," in *Circuits and Systems, 1998. ISCAS'98. Proceedings of the 1998 IEEE International Symposium on*, vol. 5. IEEE, 1998, pp. 329–332. 109

[188] M. S. Gide and L. J. Karam, "On the assessment of the quality of textures in visual media," in *Information Sciences and Systems (CISS), 2010 44th Annual Conference on*. IEEE, 2010, pp. 1–5. 114, 117

[189] "P. 1401, methods, metrics and procedures for statistical evaluation, qualification and comparison of objective quality prediction models," *International Telecommunication Union, Geneva, Switzerland*, 2012. 114

[190] Z. Wan and A. Bovik, "Mean squared error: Love it or leave it?" *IEEE Signal Processing Magazine*, pp. 98–117, 2009. 124

[191] K. Naser, V. Ricordel, and P. Le Callet, "Modeling the perceptual distortion of dynamic textures and its application in HEVC," in *Image Processing (ICIP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 3787–3791. 124

[192] I. R. Assembly, *Methodology for the subjective assessment of the quality of television pictures*. International Telecommunication Union, 2003. 125

[193] P. ITU-T RECOMMENDATION, "Subjective video quality assessment methods for multimedia applications," 1999. 125, 135

[194] J.-L. Blin, "SAMVIQ–Subjective assessment methodology for video quality," *Rapport technique BPN*, vol. 56, p. 24, 2003. 125

[195] S. Péchard, R. Pépion, and P. Le Callet, "Suitable methodology in subjective video quality assessment: a resolution dependent paradigm," in *International Workshop on Image Media Quality and its Applications, IMQA2008*, 2008, p. 6. 125

[196] K. Tsukida and M. R. Gupta, "How to analyze paired comparison data," DTIC Document, Tech. Rep., 2011. 125

[197] J. Li, M. Barkowsky, and P. Le Callet, "Boosting Paired Comparison methodology in measuring visual discomfort of 3DTV: performances of three different designs," *Proc. SPIE Electronic Imaging-Stereoscopic Displays and Applications XXIV*, 2013. 125

[198] A. R. Reibman, K. Shirley, and C. Tian, "A probabilistic pairwise-preference predictor for image quality," in *Image Processing (ICIP), 2013 20th IEEE International Conference on*. IEEE, 2013, pp. 413–417. 125

[199] Q. Xu, Q. Huang, T. Jiang, B. Yan, W. Lin, and Y. Yao, "Hodgerank on random graphs for subjective video quality assessment," *Multimedia, IEEE Transactions on*, vol. 14, no. 3, pp. 844–857, 2012. 125

[200] Z. Wang and E. P. Simoncelli, "Maximum differentiation (MAD) competition: A methodology for comparing computational models of perceptual quantities," *Journal of Vision*, vol. 8, no. 12, pp. 8–8, 2008. 125

[201] M. Nuutinen, T. Virtanen, T. Leisti, T. Mustonen, J. Radun, and J. Häkkinen, "A new method for evaluating the subjective image quality of photographs: Dynamic reference," *Multimedia Tools and Applications*, pp. 1–25, 2014. 125

[202] L. T. Maloney and J. N. Yang, "Maximum likelihood difference scaling," *Journal of Vision*, vol. 3, no. 8, p. 5, 2003. 125

[203] C. Charrier, L. T. Maloney, H. Cherifi, and K. Knoblauch, "Maximum likelihood difference scaling of image quality in compression-degraded images," *JOSA A*, vol. 24, no. 11, pp. 3418–3426, 2007. 126

[204] K. Knoblauch, L. T. Maloney *et al.*, "MLDS: Maximum likelihood difference scaling in R," *Journal of Statistical Software*, vol. 25, no. 2, pp. 1–26, 2008. 126, 128

[205] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," *Doc. VCEG-M33 ITU-T Q6/16, Austin, TX, USA, 2-4 April 2001*, 2001. 128

[206] D. H. Brainard, "The psychophysics toolbox," *Spatial vision*, vol. 10, pp. 433–436, 1997. 131

[207] K. Naser, V. Ricordel, and P. L. Callet, "A Foveated Short Term Distortion Model For Perceptually Optimized Dynamic Textures Compression In HEVC," in *32nd Picture Coding Symposium (PCS)*. IEEE, 2016. 135

UNIVERSITE BRETAGNE LOIRE

UNIVERSITÉ DE NANTES

Thèse de Doctorat

Karam NASER

Modélisation de la Similarité Perceptuelle de Textures Visuelles Statiques et Dynamiques - Application à l'Optimisation Perceptuelle de la Compression Vidéo

Modeling the Perceptual Similarity of Static and Dynamic Visual Textures - Application to the Perceptual Optimization of Video Compression

Résumé

Les textures sont des signaux particuliers dans la scène visuelle, où elles peuvent couvrir de vastes zones. Elles peuvent être classées en deux catégories : statique et dynamique, où les textures dynamiques impliquent des variations temporelles. Plusieurs travaux sur la perception des textures statiques ont permis de définir des mesures de similarité visuelle pour des applications comme la reconnaissance ou la classification de textures. Ces mesures utilisent souvent une représentation inspirée du traitement neuronal du système visuel humain. Cependant de telles approches ont été peu explorées dans le cas de textures dynamiques. Dans cette thèse, un modèle perceptuel généralisé pour la mesure de similarité applicable aux textures statiques et dynamiques, a été développé. Ce modèle est inspiré du traitement effectué dans le cortex visuel primaire. Il s'avère très efficace pour des applications de classification et de reconnaissance de textures. L'application du modèle dans le cadre de l'optimisation perceptuelle de la compression vidéo, a été également étudiée. En particulier, l'intégration de la mesure de similarité entre textures, a été utilisée pour l'optimisation débit-distorsion de l'encodeur. Les résultats expérimentaux avec observateurs humains montrent une qualité visuelle améliorée des vidéos ainsi codés/décodées, avec une réduction significative du débit par rapport aux approches traditionnelles.

Mots clés

Similarité visuelle, Analyse de texture, Modélisation du système visuel humain, Compression vidéo perceptuelle

Abstract

Textures are special signals in the visual scene, where they can cover large areas. They can be classified into two categories: static and dynamic, where dynamic textures involve temporal variations. Several works on the perception of static textures made it possible to define visual similarity measurements for applications such as the recognition or classification of textures. These measures often use a representation inspired by the neural processing of the human visual system. However, such approaches have been little explored in the case of dynamic textures. In this thesis, a generalized perceptual model for the measurement of similarity applicable to static and dynamic textures has been developed. This model is inspired by the processing performed in the primary visual cortex. It is very effective for texture classification and recognition applications. The application of the model in the context of the perceptual optimization of video compression, was also studied. In particular, the integration of the similarity measure between textures, was used for the rate-distortion optimization of the encoder. Experimental results with human observers showed an improved visual quality of the decoded videos, with a significant reduction in the bitrate compared to the traditional approaches.

Key Words

Visual similarity, Texture analysis, Human visual system modeling,Perceptual video compression