UNIVERSITY OF CALIFORNIA, SAN DIEGO

# Perceived Blur in Stereoscopic Video: Experiments and Applications

A dissertation submitted in partial satisfaction of the
requirements for the degree
Doctor of Philosophy

in

Electrical Engineering (Signal and Image Processing)

by

Ankit K. Jain

Committee in charge:

Professor Truong Q. Nguyen, Chair
Professor Donald I. A. MacLeod, Co-Chair
Professor Stuart M. Anstis
Professor Pamela C. Cosman
Professor Kenneth Kreutz-Delgado

2014

The dissertation of Ankit K. Jain is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

Co-Chair

Chair

University of California, San Diego

2014

# DEDICATION

To the memory of my father, *for your inspiration*;<br>
To Mukul, Malini, and Samir, *for your constant faith and support*;<br>
To Aarav and Shaan, *for all the great things you will become*;

And, above all,<br>
To Mom, *for your incredible strength and love.*

# TABLE OF CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# ACKNOWLEDGMENTS

I would like to express my gratitude and respect for my advisor, Prof. Truong Nguyen, for his leadership and faith throughout my graduate career. The freedom he afforded me in my research and his student-first mentality have made my graduate experience a pleasurable one.

I feel very fortunate to have had Prof. Donald MacLeod as my co-advisor. I am indebted to him for inspiring and nurturing my interest in vision science, and have benefited tremendously from his consummate knowledge of the field and countless helpful discussions that aided my research.

I am also grateful to my committee, Profs. Stuart Anstis, Pamela Cosman, and Kenneth Kreutz-Delgado, for valuable feedback, proofreading, and encouragement.

I would like to thank Dr. Alan Robinson, whose collaboration gave focus to my research and bore fruit in the form of several publications. Alan's patience and direction were formative as I caught up on psychophysical methods and analysis, and his meticulous research and writing skills stretched my own. I also thank Can Bal for his contributions to Chapter 3 of this dissertation and other projects.

Many current and former members of the Video Processing Lab and the Vision Lab have provided support and enthusiasm over the course of my graduate studies, for which I am truly grateful. I would also like to thank the people at Pelican Imaging for taking an interest in my education and career, and their understanding in working around my schedule.

Today, I am remembering fondly the many teachers, professors, and colleagues who have shaped my education and identity throughout the different stages of my life: growing up in Davis, college at Stanford, and working at Lincoln Lab in Boston. I am grateful to my friends, particularly my roommates over the years, who have provided much-needed distractions and merriment. To my family around the world and especially those to whom this work is dedicated, nothing I do will ever rival what you have done for me, but I will always try.

The text of Chapter 2 is, in part, a reprint of a paper that has been published as: A. K. Jain, A. E. Robinson, and T. Q. Nguyen, "Comparing perceived quality

and fatigue for two methods of mixed resolution stereoscopic coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 3, pp. 418–429, Mar. 2014. The dissertation author was the primary investigator and author of this paper.

The text of Chapter 3 is, in part, a reprint of a paper that has been published as: A. K. Jain, C. Bal, and T. Q. Nguyen, "Tally: a web-based subjective testing tool," in *Fifth Int'l. Workshop on Qual. of Multim. Exp. (QoMEX)*, Jul. 2013, pp. 128–129. The dissertation author was the primary investigator and author of this paper.

The text of Chapter 4 is, in part, a reprint of a paper that is being prepared for publication as: A. K. Jain and T. Q. Nguyen, "Video super-resolution for mixed resolution stereoscopic Coding," *IEEE Trans. Circuits Syst. Video Technol.*, in preparation. The dissertation author was the primary investigator and author of this paper.

The text of Chapter 5 is, in part, a reprint of a paper that has been published as: A. K. Jain and T. Q. Nguyen, "Discriminability limits in spatio-temporal stereo block matching," *IEEE Trans. Image Process.*, vol. 23, no. 5, pp. 2328-2342, May 2014. The dissertation author was the primary investigator and author of this paper.

The text of Chapter 6 is, in part, a reprint of a paper that is being prepared for publication as: A. K. Jain, A. E. Robinson, D. I. A. MacLeod, and T. Q. Nguyen, "Anisotropic spatial integration in the sensing of horizontal disparity," *Vision Research*, in preparation. The dissertation author was the primary investigator and author of this paper.

Chapter 1 contains some material reprinted from all of the above papers.

# VITA

| | |
|---|---|
| 2005 | B.S., Electrical Engineering, Stanford University |
| 2005-2008 | Assistant Technical Staff, MIT Lincoln Laboratory |
| 2010 | M.S., Electrical Engineering (Signal and Image Processing), University of California, San Diego |
| 2011-2014 | Technical Consultant, Pelican Imaging Corporation |
| 2014 | Ph.D., Electrical Engineering (Signal and Image Processing), University of California, San Diego |

# PUBLICATIONS

A. K. Jain, A. E. Robinson, D. I. A. MacLeod, and T. Q. Nguyen, “Anisotropic spatial integration in the sensing of horizontal disparity,” *Vision Research*, in preparation.

A. K. Jain and T. Q. Nguyen, “Video super-resolution for mixed resolution stereoscopic Coding,” *IEEE Trans. Circuits Syst. Video Technol.*, in preparation.

A. K. Jain and T. Q. Nguyen, “Discriminability limits in spatio-temporal stereo block matching,” *IEEE Trans. Image Process.*, vol. 23, no. 5, pp. 2328-2342, May 2014.

A. K. Jain, A. E. Robinson, and T. Q. Nguyen, “Comparing perceived quality and fatigue for two methods of mixed resolution stereoscopic coding,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 3, pp. 418–429, Mar. 2014.

A. E. Robinson, A. K. Jain, M. Scott, D. I. A. MacLeod, and T. Q. Nguyen, “Apparent sharpness of 3D video when one eye’s view is more blurry,” *i-Perception*, vol. 4, no. 6, pp. 456–467, 2013.

A. K. Jain, C. Bal, and T. Q. Nguyen, “Tally: a web-based subjective testing tool,” in *Fifth Int’l. Workshop on Qual. of Multim. Exp. (QoMEX)*, Jul. 2013, pp. 128–129.

A. K. Jain and T. Q. Nguyen, “Video super-resolution for mixed resolution stereo,” in *IEEE Int’l. Conf. Image Proc. (ICIP)*, Sep. 2013, pp. 962–966.

A. K. Jain, C. Bal, A. Robinson, D. I. A. MacLeod, and T. Q. Nguyen, “Temporal aspects of binocular suppression in 3D video,” in *Sixth Int’l. Workshop Video Proc. Qual. Metrics Consum. Elec. (VPQM)*, Jan. 2012, pp. 93–98.

A. K. Jain, L. C. Tran, R. Khoshabeh, and T. Q. Nguyen, "Efficient stereo-to-multiview synthesis," in *IEEE Int'l. Conf. Acoust., Speech, Sig. Proc. (ICASSP)*, May 2011, pp. 889–892.

C. Bal, A. K. Jain, and T. Q. Nguyen, "Detection and removal of binocular luster in compressed 3D images," in *IEEE Int'l. Conf. Acoust., Speech, Sig. Proc. (ICASSP)*, May 2011, pp. 1345–1348.

L. C. Tran, R. Khoshabeh, A. K. Jain, C. Pal, and T. Q. Nguyen, "Spatially consistent view synthesis with coordinate alignment," in *IEEE Int'l. Conf. Acoust., Speech, Sig. Proc. (ICASSP)*, May 2011, pp. 905–908.

S. Chan, A. K. Jain, T. Q. Nguyen, and E. Y. Lam, "Bounds on condition numbers of spatially variant convolution matrices," in *Signal Recovery and Synthesis.* Optical Society of America, 2011, p. SMA4.

ABSTRACT OF THE DISSERTATION

## Perceived Blur in Stereoscopic Video: Experiments and Applications

by

Ankit K. Jain

Doctor of Philosophy in Electrical Engineering (Signal and Image Processing)

University of California, San Diego, 2014

Professor Truong Q. Nguyen, Chair
Professor Donald I. A. MacLeod, Co-Chair

Nominally, the stereoscopic representation of a scene requires twice as much bandwidth as the corresponding monocular representation, since a separate view is presented to each eye. This increase in data rate is one of the most imposing challenges for stereoscopic media transmission and consumption. Mixed resolution stereoscopic coding (MRSC) has been proposed to mitigate the bandwidth requirements by transmitting a stereo pair comprised of one full resolution image and one lower resolution image. MRSC preserves the overall video quality by relying on the theory of binocular suppression, a perceptual phenomenon where if one eye's view of the world is blurry while the other eye's view is sharp, then the fused 3D percept of the scene will appear relatively sharp and faithfully represented in depth.

This dissertation examines the binocular perception of blur and its application to 3D video processing. We begin by investigating the temporal aspects of MRSC—the effects of frame rate and blur on perceived quality and viewer fatigue—and whether balancing blur between both eyes is necessary. Subsequently, we develop a super-resolution method for MRSC that restores high frequency content to the low resolution half of the stereo pair. Our method results in sharper images and temporally consistent video. As part of our analysis, we derive expressions that quantify how much motion can aid the stereo matching process as a function of image features, noise, and motion distribution. These limits are relevant to the design of algorithms for spatio-temporal disparity estimation. Finally, in consideration of the visual systems's natural preference for horizontal disparity, we examine the influence of anisotropic spatial filtering on the perception depth. We obtain contrast sensitivity functions for depth detection as a function of filter orientation, and develop a computational model of sensitivity based on the power spectrum of the stimulus. These results have implications for theories of stereopsis mechanisms, and can also be used in rate distortion decisions for MRSC. Taken together, this work establishes MRSC as a perceptually plausible coding technique for stereoscopic video, and more generally, can be applied to the development of stereo algorithms and metrics.

# Chapter 1

# Introduction

Stereo vision in humans and other animals is characterized by having two eyes laterally displaced from one another with largely overlapping fields of view. Each eye, owing to the mutual displacement, receives a slightly different perspective of the world. A point in the scene is imaged at a different point in each retina, where the difference in retinal position is inversely proportional to the square of the distance from the observer. Therefore, closer objects exhibit a larger relative shift between corresponding retinal images in each eye, and farther objects exhibit less of a shift. These shifts and their relationship to scene depth form what is known as the disparity cue, the basis of stereo depth perception. The brain is able to match corresponding points between the two eye's images, determine the disparity between them, and use that information to determine the relative depth ordering of objects in the scene.

First discovered by Charles Wheatstone in 1838 [1], the disparity cue can be produced artificially by capturing images of a scene from cameras with a horizontal displacement (or carefully drawing two perspective images) and separately delivering the appropriate view to each eye. Thus, captured imagery stands in for real-world input, but the rest of the retinal and neural processing remains the same. Wheatstone's original stereoscope, a device used for dichoptic display of stereo imagery, consisted of a pair of mirrors oriented oblique to two lateral stands that held the left and right images. The observer would look into the mirrors so that the images would be reflected separately into each eye. This concept is the basis of 3D display technology today. Devices that display 3D imagery must orthogonally multiplex the two views along a certain dimension: space, time, color, or polarization.

Since Wheatstone's discovery, 3D imagery has enjoyed periods of intense consumer interest. From the mid to late 1800's, people in both Europe and America regularly viewed stereograms as part of news and entertainment, and the stereoscope became a common household item [2]. There was a resurgence of interest in stereoscopy with the advent of film, and in the 1950's, anaglyph and polarized stereo movies became very popular. Once again, in the last five to ten years, the film industry has revived stereo as a common entertainment medium. The current stereo epoch has been longer lived and more profitable than its predecessor of last century. Many of the issues with 3D that led to its decline in previous eras—low-quality capture and production, mediocre display technology, and its naive use in storytelling—have been vastly improved.

However, modern interest in 3D technology is not limited to Hollywood. The enhanced sense of realism afforded by stereopsis makes 3D viewing a centerpiece in applications such as robotically assisted surgery, flight simulators for pilots, virtual reality, gaming, and remote vehicle navigation. Further, the ubiquity of technology such as consumer-level stereo cameras, online videos, TVs, and computer graphics enables the widespread capture, display, and distribution of stereo content. Whereas consumption of stereo media of the past two centuries was confined to print and movie theaters, 3D viewing today is demanded on nearly any device in any location. Coupled with expectations of high quality and fast service, a major challenge for modern 3D media is video compression. Delivering a separate view to each eye nominally doubles the amount of data to be transmitted; hence, efficient representations of 3D data is an active research topic.

The left and right views constituting a stereo pair must typically be very well-matched in order for comfortable stereopsis to occur. Our two eyes are virtually identical; equally sensitive to the same color bands, intensity levels, distribution of rods and cones, and processing. Similarly, extensive consideration is made to match the characteristics of the two cameras in a stereo rig. Inconsistencies between the views can induce a variety of perceptual effects, depending on the nature of the asymmetry. If each eye sees a different scene, (i.e. that does not result in stereopsis) binocular rivalry can occur: the perceived image will alternate between what each eye sees [3]. When one view is slightly magnified relative to the other, the pronounced vertical disparities can create a perception of tilt [4]. If an object is darker than the background in one view but lighter than the background in the other view, an entirely different effect is perceived; the object is said to exhibit binocular luster, and appears to shimmer or sparkle [3, 5]. In other cases when an object has a different luminance in each view but same polarity relative to the background, the object luminance appears to be averaged. Similar but more complex mechanisms of chromatic luster and averaging are observed when relative color contrasts between an object and the background are observed.

Asymmetric blur among a stereo pair is a unique case. If one eye's view of the world is blurry while the other eye's view is sharp, then the fused 3D percept of the scene will appear relatively sharp and faithfully represented in depth. First noted by Julesz in [6], he writes in his later book "...whichever of the two views contains the high frequencies in a given area will dominate in the final percept." [7]. The perceived 3D image will not be quite as sharp as the sharpest image of the pair, but will be significantly sharper than the blurry view [8]. Though psychophysical studies find stereoacuity thresholds degraded as a result of dichoptic blur [9, 10], overall depth impressions from natural scenes are preserved [11, 12]. The fact that humans are sensitive to frequencies in depth of only a few cycles per degree of visual angle, several times lower than our sensitivity to spatial frequencies [13], suggests that stereo depth can be well-represented by low resolution luminance. The phenomenon of the sharper image being weighted more heavily in the fused percept is often referred to as *binocular suppression*, and an asymmetrically blurred stereo pair is said to be of *mixed resolution*. An example stereo pair from [14, 15] processed to exhibit binocular suppression is shown in Fig. 1.1.

Thus, while other asymmetries can be quite salient, blur is relatively inconspicuous. The ability to discard frequency content from one view while hardly sacrificing overall quality has clear implications for compression of stereo imagery.

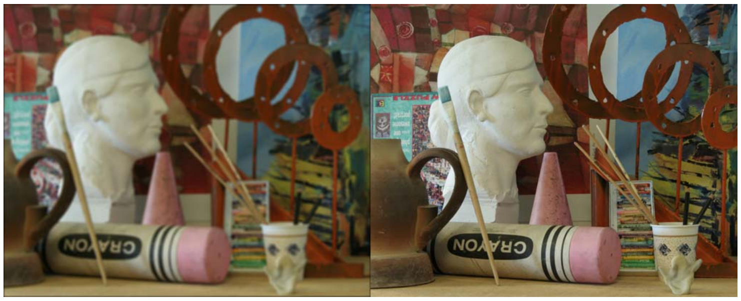


**Figure 1.1**: A stereo pair in which the right image has been blurred. When fused, the 3D percept is relatively sharp. Stereo pair set up for cross-eyed viewing.

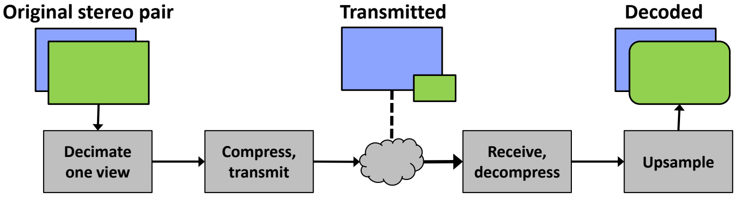


**Figure 1.2**: Structure of a mixed resolution codec.

Indeed, mixed resolution stereoscopic coding is a method that exploits the principle of binocular suppression by downsampling one of the two views, compressing the stereo pair, then transmitting the data. Upon receipt at the decoder, the stereo pair is decoded and the lower resolution view is upsampled to full size, but remains blurry relative to its companion full resolution view. This procedure is illustrated in Fig. 1.2.

Other techniques for 3D compression, such as the MVC extension to H.264 and 2D + Depth, have also been explored; however, mixed resolution coding is unique in that it exploits properties of human visual stereo perception. Therefore, the development of mixed resolution as a coding technique requires an interdisciplinary approach between the fields of video processing and vision science. Accordingly, there is potential value to both disciplines in the understanding of binocularly perceived blur, a topic that warrants closer study.

## 1.1 Dissertation Contributions

This dissertation examines the binocular perception of blur and its application to stereoscopic video compression and processing. We investigate temporal aspects of mixed resolution compression and restoration, as well as how spatial blur affects depth perception. Specifically, our contributions are:

1. We test whether mixed resolution stereo is comfortable to view, and if the blur needs to be temporally balanced between the two eyes. This was an open question before that has important consequences for the coding of mixed resolution video.

2. We introduce a free, open source software package to simplify data collection for video quality experiments. This is the first such tool that is freely and widely available, customizable, and possesses features exclusive to its web-based design, such as simultaneous testing of multiple subjects and collaboration between remote labs.

3. We develop a method to restore quality to the low resolution half of a compressed mixed resolution stereo pair that enforces temporal consistency. Our algorithm outperforms competing methods and is designed for video, whereas prior work generally neglected temporal information.

4. We derive theoretical limits on how much motion can aid stereo matching. Such a theoretical examination had not been conducted before, and it gives insight into video disparity estimation algorithms.

5. We obtain contrast sensitivity functions for sensing depth as a function of texture orientation and spatial frequency, as well as a computational model to calculate depth sensitivity. Our model and data can be applied for mixed resolution subsampling and stereo quality metrics, and are also informative about depth processing in the visual system.

## 1.2 Dissertation Outline

The topics discussed in Section 1.1 are the subject of the rest of this dissertation, organized as follows.

Chapter 2 addresses the question of whether mixed resolution video is fatiguing to watch, and if blur needs to be temporally balanced between the two stereo views. Many investigations have shown that a large reduction in bandwidth can be gained for a relatively small compromise in image quality by a mixed resolution representation. However, the viability of such an encoding method depends on the subjective response to viewing such videos at length. While methods for mixed resolution coding have been developed on the presumption of a certain visual fatigue response, none have actually examined it. We address this shortcoming in three experiments comparing two methods of binocular suppression processing. The first two experiments reveal subjects' preferences in terms of overall quality between the two methods for short exposures, and the third experiment examines the fatigue resulting from 10-minute exposures to mixed resolution encoded videos.

The third experiment in Chapter 2 would have been impossible without special software for data recording. In fact, no available data collection mechanism existed for 3D video, other than manual data recording, even for simpler testing methodologies. In Chapter 3, we introduce Tally, a modern web-based tool to help automate data collection for subjective video quality experiments. We created Tally in response to the lack of suitable testing software, and designed it with decoupled voting and viewing interfaces so that it can be used for both 2D and 3D displays. Its web-based design provides ubiquitous data access, the ability to share and collaborate on projects, and compatibility with mobile devices. Standard ITU test methods are supported, and we have released the software as free and open source for continued development.

Though mixed resolution video is generally comfortable to view as shown in Chapter 2, certain applications would benefit from having both views at full resolution. In Chapter 4, we present an example-based super-resolution method to recover the lost frequency information in compressed mixed resolution stereo video. High frequency information is projected from one view onto the other via stereo matching, and spatial, temporal, and spectral consistency is enforced within a 3D Markov network. Our algorithm is distinguished from prior art by considering the temporal dimension in its formulation, allowing for more accurate stereo matching, efficient computation by exploiting temporal correlation, and temporal consistency. We operate solely on the stereo pair at the decoder, requiring no depth or side information, and report high objective measures of quality and temporal smoothness. Further, the extensive evaluation of our method is performed on real compressed video data, which has implications for the viability and operating range of mixed resolution stereo compression.

Video disparity estimation methods, as in Chapter 4, use motion as a matching criterion to help disambiguate spatially similar candidates. In Chapter 5, we examine the validity of the underlying assumptions of spatio-temporal disparity estimation, and determine the extent to which motion aids the matching process. By analyzing the error signal for spatio-temporal block matching under the sum of squared differences criterion and treating motion as a stochastic process, we determine the probability of a false match as a function of image features, motion distribution, image noise, and number of frames in the spatio-temporal patch. This performance quantification provides insight into when spatio-temporal matching is most beneficial in terms of the scene and motion, and can be used as a guide to select parameters for stereo matching algorithms. We validate our results through simulation and experiments on stereo video.

In seeking an efficient representation of depth information, as mixed resolution aims to do, it makes sense to consider the method by which downsampling is performed. Due to the lateral separation of the eyes, there is a natural preference for vertical versus horizontal contours in stereopsis. In Chapter 6, we investigate this anisotropy by measuring contrast thresholds for depth detection from bandpass filtered random dot stereograms for different spatial center frequencies, disparities, and five filter orientations: horizontal, oblique, vertical, and two isotropic configurations. Sensitivities were much lower for horizontally oriented textures than for other conditions, except when a frequency-disparity combination results in a large binocular phase, and hence an ambiguous stimulus. The allocation of spectral components to the vertical direction is generally the most efficient for depth detection, but again depends on binocular phase. We develop a disparity energy model based on the power spectrum of the stimulus and the phase encoding model of disparity that adequately explains the data. We discuss the implications of our results for models of stereopsis, anisotropic integration prior to stereo matching, and how this anisotropy relates to the opposite anisotropy found in the cyclopean domain. Our model can be used in rate distortion characteristics for video compression, or metrics for depth sensation.

Finally, we conclude in Chapter 7 with a summary of our work and directions for future research.

## Acknowledgments

The text of Chapter 1 is, in part, a reprint of the following papers that have been published or are being prepared for publication as:

A. K. Jain, A. E. Robinson, and T. Q. Nguyen, “Comparing perceived quality and fatigue for two methods of mixed resolution stereoscopic coding,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 3, pp. 418–429, Mar. 2014.

A. K. Jain, C. Bal, and T. Q. Nguyen, “Tally: a web-based subjective testing tool,” in *Fifth Int’l. Workshop on Qual. of Multim. Exp. (QoMEX)*, Jul. 2013, pp. 128–129.

A. K. Jain and T. Q. Nguyen, “Video super-resolution for mixed resolution stereoscopic Coding,” *IEEE Trans. Circuits Syst. Video Technol.*, in preparation.

A. K. Jain and T. Q. Nguyen, “Discriminability limits in spatio-temporal stereo block matching,” *IEEE Trans. Image Process.*, vol. 23, no. 5, pp. 2328-2342, May 2014.

A. K. Jain, A. E. Robinson, D. I. A. MacLeod, and T. Q. Nguyen, “Anisotropic spatial integration in the sensing of horizontal disparity,” *Vision Research*, in preparation.

The dissertation author was the primary investigator and author on all of these papers.

# Chapter 2

# Comparing Perceived Quality and Fatigue for Two Methods of Mixed Resolution Stereoscopic Coding

Many studies have investigated perceptual limits of asymmetric blur or mixed resolution downsampling ratios [11,16,17] and codec design [18–20] for the traditional method of mixed resolution coding, in which one view is always sharp and the other is always blurry. These works have shown the viability of the mixed resolution coding scheme from a visual quality standpoint, in that the sharpness and depth quality are hardly compromised for a significant reduction in bandwidth. However, they assume that binocular suppression content will be comfortable to view despite having one eye continually out of focus. Conversely, the motivation behind several other works is to balance the blur level across both eyes in order to avoid eye strain or fatigue over prolonged exposure to asymmetric content. This assumption is likewise unverified.

In this chapter, we present a trio of experiments that compares the traditional method of mixed resolution coding to a balanced method where the blurry view alternates between the left and right views. The first two experiments build on our work in [12] and focus on the perceptual quality of the two processing methods for short video clips. Our third experiment answers the question as to whether exposure to asymmetrically blurred 3D videos induces eye strain or fatigue to the viewer. Taken together, these experiments have implications for mixed resolution as a viable coding scheme.

This chapter is organized as follows. In Section 2.1, we review previous studies and findings relevant to the present work. In Sections 2.2–2.4, we discuss the design and results of the quality and fatigue experiments. Section 2.5 contains a discussion of the results and potential applications of the study, and we conclude the chapter in Section 2.6 with a summary and observations for future work.

## 2.1 Related Work

The original and simplest method of mixed resolution coding is to downsample the view corresponding to one eye while leaving the other eye's view intact. Early work in applying the binocular suppression property to image coding is found in [21–23]. The study in [11] reports little quality and depth sensation degradations even when the spatial frequency was reduced to half of its original bandwidth, while similar reductions along the temporal axis were immediately noticeable.

Several methods have been proposed to balance the blur across both eyes. Blurring spatially alternating horizontal slices of each view was proposed in [24], whereas a coding structure that switches the blurry view temporally at every GOP (group of pictures) is proposed in [25]. In [26, 27], the visibility of switching the blurry view from one eye to the other was investigated for natural images and video, as well as for random dot stereograms. It was found that viewers noticed the switch, with detection rates increasing with degree of blur, although detection was well-masked by scene cuts. The motivation behind several works [24–28] is to balance the blur level across both eyes in order to avoid eye strain or fatigue over prolonged exposure to asymmetric content. Our work investigates this reasoning.

In our previous work [12], we examined the overall subjective quality of the traditional method of mixed resolution coding in which one of the two stereo views of a video is spatially downsampled, as well as alternating the blurry view at every frame. Our results show that both methods perform about equally at 60 Hz, where the alternation of blur between the eyes is relatively imperceptible. The present study extends this examination and contrasts the results with other blur balancing methods in Section 2.5.

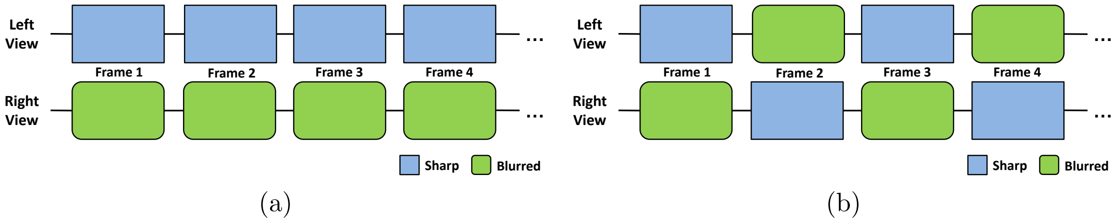
Previous work in assessing viewer discomfort or fatigue for 3D video has focused on accommodation and vergence [29–31], viewing zone [32, 33], or display type [34]. Our work expands this literature by considering the effects of processing methods on viewer fatigue for mixed resolution coding.

## 2.2 Experiment Design

We conduct 3 experiments described in Sections 2.3 and 2.4. The first two experiments measure quality or sharpness differences between the two processing methods described in Section 2.2.1 for short video clips, while the third experiment compares the visual fatigue of the same two methods over a longer exposure. To simulate a lower resolution image in the experiments, a blur filter of the type described in Section 2.2.2 is applied with various diameters.

### 2.2.1 Methods of Binocular Suppression

In this study, we compare two methods of processing 3D video to exploit the visual suppression of blurred images. The first method is to blur all frames of the video corresponding to one of the eyes. This is the traditional method of binocular suppression, and has been well-explored. The choice of which view to blur is arbitrary, and here we chose to blur the right view. The second method is to blur alternate frames of each view, such that there is one blurry and one sharp frame at each time instance, and the view that is blurred alternates with each frame. A depiction of the two processing schemes is given in Fig. 2.1.



(a) (b)

**Figure 2.1**: Two different blurring schemes for mixed resolution stereo video coding. (a) Single-eye blur. (b) Alternating-eye blur.

### 2.2.2 Filter Design

Mixed resolution coding is accomplished by downsampling one of the two stereo views at the encoder, transmitting the stereo pair, then upsampling the lower resolution view at the decoder for playback. In our experiments, we approximate this process by lowpass filtering certain frames of the video. This produces fewer visual artifacts than would downsampling an image to a smaller size, and then upsampling back to the original resolution. There exist many different methods that could be used in the upsampling process, and we did not want to limit our results to a particular method and the artifacts it produces.

We chose to use a disk filter of varying diameters as the blur kernel for these experiments. This kernel was chosen for its radial symmetry as well as its flat response. These properties ensure that no directional bias is introduced in the spectrum of the filtered image, and that pixels are averaged equally across the region of support. The positive half of a cross section of the filter frequency response used in the fatigue experiment is shown in Fig. 2.2.

## 2.3 Quality Experiments

In our previous work [12], we found the perceived quality differences between single-eye blur and alternating-eye blur to be small for presentations at 60 Hz. Here we extend that work by testing a wider range of refresh rates and greater amounts of blur. Rather than a numeric rating scale which can require a large number of trials to get meaningful results, we use the more direct paradigm of asking subjects

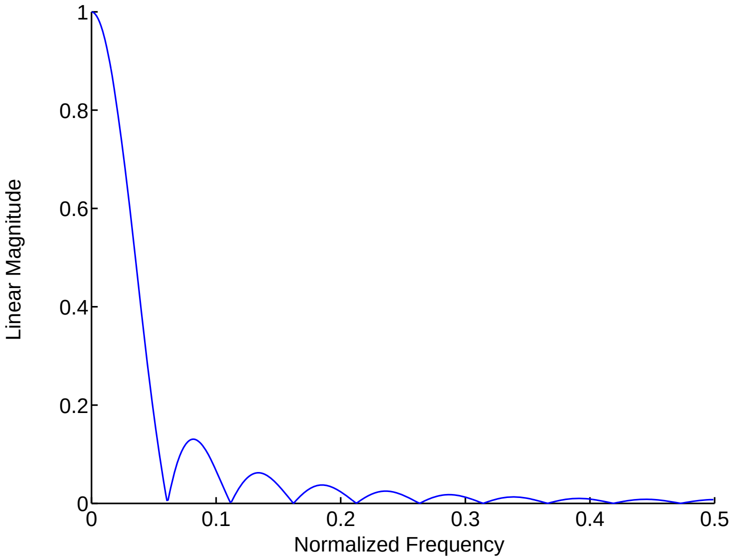


**Figure 2.2**: Frequency response of blur kernel used in fatigue experiment.

which of the pair of videos, each processed according to one of the two methods, they preferred. For the experiments described in this section, the sole difference between the two videos in each paired comparison was the processing method (single-eye versus alternating-eye blur). By showing the two presentation types in a row with a short delay, the differences between them are highlighted, and there is less risk of changing standards over the duration of the experiment producing random drift in the numeric rating.

Stimuli were presented on a 22” LaCie electron22blueIV Diamondtron CRT driven by an NVIDIA GeForce GT 545 GT video card running at 1280 × 1024 resolution with a refresh rate of 120 Hz, in an otherwise unlit room. A chinrest was used to maintain a viewing distance of 1.95 m. A mirror stereoscope presented a separate image to each eye. Each image subtended 6° × 9.6° (WxH) with a resolution of 640×1024 pixels. Video playback was controlled using Matlab running the Psychophysics Toolbox, version 3 [35–37] on a Windows XP computer.

The long viewing distance made it possible to present very high resolution content (107 pixels/degree). The use of a CRT and mirrors resulted in zero crosstalk between eyes, zero inherent flicker, and a high refresh rate, a combined set of features that cannot yet be achieved with LCD shutter glasses or passive polarized 3D LCDs. This apparatus was used for the first two experiments.

We used four high quality sports-themed stereoscopic video clips taken from the *LG 2012 Demo* disk, which was originally encoded at 1280x720, side by side, at 30 Hz with a bitrate of 13.6 Mb/s using H.264 AVC compression in the YUV color space with 4:2:0 subsampling. Most of the video consists of foreground objects moving slowly against a fairly static background, thus the content was quite sharp at this bit rate. Played at 30 Hz, the clips all depicted various amounts of slow-motion. Played back at 120 Hz, the videos were a mixture of faster and slower than real time, with the majority of frames somewhat slower than real time and a range of visual motion speeds from slow to quite fast. To create 60 Hz and 30 Hz videos we took the 120 Hz sequences and dropped frames; thus, each version was exactly the same duration and had the same speed of visual motion, independent of frame rate.

The four sequences (and durations) were *Karate Kick* (2 s), *Karate Board Split* (1.27 s), *Pole Vault* (1.37 s), and *Baseball* (1.63 s)[1]. Our choice of content was limited by our need for video that could be played back at 120 Hz without appearing drastically sped up, which is why the video lengths differed. The videos were chosen because they were visually varied and we hoped to avoid too much subject boredom from repetition. The differences between videos were not intended to test any particular hypothesis. All clips filled the horizontal dimension of the screen (6° per eye), but only about two-thirds of the vertical dimension (6.7°); the rest of the screen was black. While our videos were shorter than is typical in video quality rating experiments where subjects use a subjective scale (e.g. 0-10), the stimuli durations we used are actually quite common for psychophysics experiments where subjects indicate a binary preference between two stimuli, such as which one appears more blurry [38]. After preparing the videos for our experiment (changing playback rate and applying blur) the content was saved in uncompressed form to prevent adding compression artifacts.

[1] These videos started at frames 1528, 1874, 2673, and 3224, respectively.

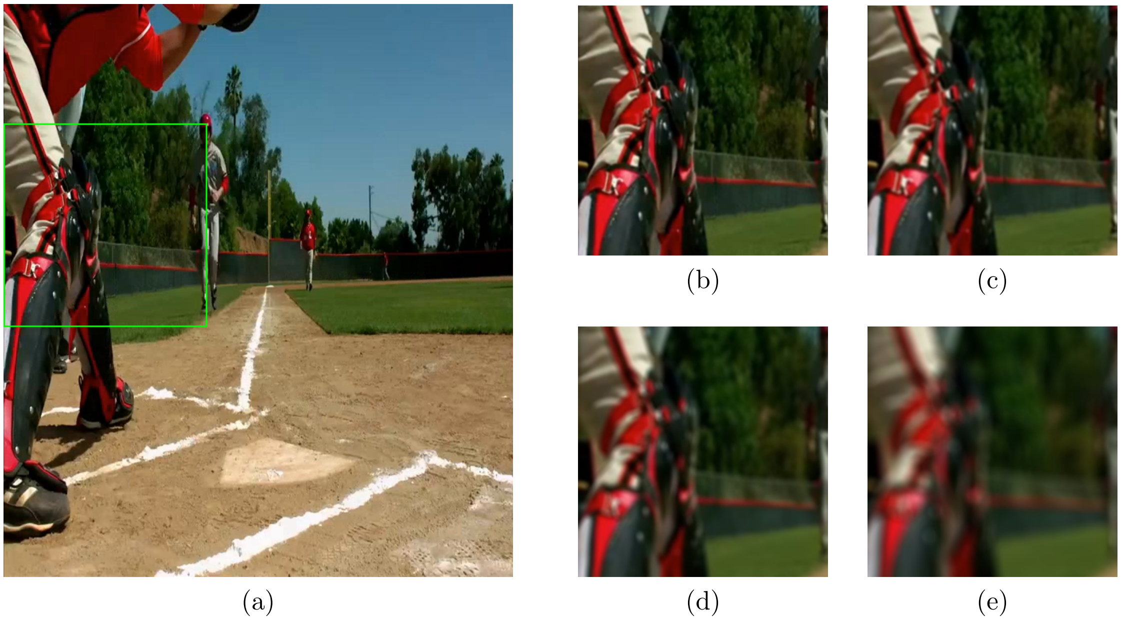


**Figure 2.3**: Example stimuli from the *Baseball* clip used in Experiments 1–2. (a) Original frame. (b)–(e) Frame in (a) filtered with a 2, 4, 8, and 16-pixel diameter disk filter (boxed area shown). A viewing distance of 1.6 m will approximate the spatial resolution of stimuli displayed to subjects during the experiments.

### 2.3.1 Experiment 1

We tested 4 blur levels corresponding to a diameter of 2, 4, 8 or 16 pixels of the kernel in Section 2.2.2. These ranged from subtle to very obvious levels of blur when viewed monocularly (see Fig. 2.3 for an example of the stimuli). To give an idea of how much the bandwidth of a video is reduced by the application of these filters, we estimate the equivalent downsampling ratio for each filter in Table 2.1. This ratio is the factor by which the video can be spatially downsampled in each direction without aliasing after applying the blur filter. We calculate this quantity by inverting the normalized cutoff frequency of the filter (expressed as a percentage of the Nyquist rate), where we consider the cutoff to be the location of the first zero in the filter frequency response. Note that a disk filter with diameter 2 induces very little blur, and does not have a zero in its frequency response. The perceived angular size of the filter for the viewing distance for these experiments is also listed in Table 2.1.

**Table 2.1**: Blur filter sizes and bandwidth measurements for Experiments 1–2.

| Blur<br>diameter<br>(pixels) | Angular<br>size<br>(arcmin) | Cutoff<br>frequency<br>(% Nyquist) | Equivalent<br>downsampling<br>ratio |
|---|---|---|---|
| 2 | 1.13 | 100 | 1.00 |
| 4 | 2.25 | 63.1 | 1.58 |
| 8 | 4.5 | 30.7 | 3.26 |
| 16 | 9.0 | 15.2 | 6.56 |

We used 3 frame rates: 30, 60, and 120 Hz. The 4 source videos, 4 blur levels, and 3 playback rates correspond to 48 unique combinations. Each test condition was repeated 4 times for a total of 192 trials, which were tested in random order. To give subjects some experience with the task, the experiment began with 8 additional training trials which were not included in the data analysis.

On each trial, subjects viewed a pair of video clips where one was processed for single-eye blur and the other for alternating-eye blur, but were identical in their source video, frame rate, and blur diameter. Their task was to indicate which of the two clips was more pleasant to watch. If they felt there was absolutely no difference they were also allowed to indicate that instead. We did not tell them how the videos differed, but told them they might notice "differences in sharpness, flicker, smoothness of motion, and visual fatigue". We also stressed to them the importance of using both eyes, "so that they could see the videos in 3D".

The sequence timing was Movie A, 0.5 s gray screen, Movie B, 1.5 s gray screen; repeat until subject responds A or B. Subjects were encouraged to watch the full sequence at most 3 times per trial, but there was no actual limit imposed. The type of blur played first was varied randomly between trials.

Twenty-three naive subjects participated, all with normal or corrected-to-normal acuity, and the ability to perceive stereoscopically defined depth. Depth perception was tested by asking subjects to report the depth ordering of rectangles that differed only in disparity.

Fig. 2.4 shows the proportion of trials where subjects preferred single-eye blur over alternating blur as a function of blur diameter, for the three different

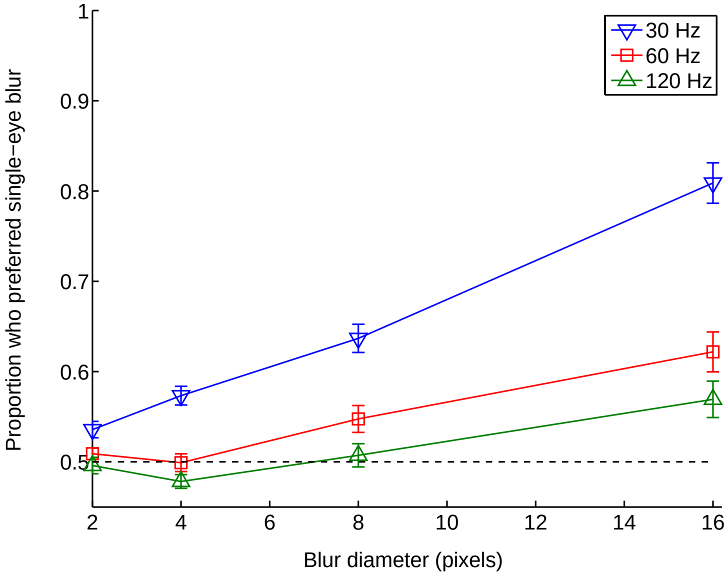


**Figure 2.4**: Experiment 1 results. Proportion of trials in which single-eye blur was preferred over alternating-eye blur as a function of blur diameter. Error bars denote 95% confidence intervals.

frame rates. Calculating this proportion is complicated by the fact that subjects were allowed to indicate “no preference”. This was handled by coding each trial as 0 (preferred alternating-eye blur), 1 (preferred single-eye blur), or 0.5 (no preference) and then taking the average. This weighted proportion represents a coding of a no-preference trial as two trials, one preferring single-eye and one preferring alternating-eye blur, with each trial receiving half of the regular weight. We did not see any consistent difference in preferences for blur type among the four different source videos, so we have combined data across that factor.

A repeated measures ANOVA found a significant effect of refresh rate ($F(2, 42) = 31.6$, $p < 0.000001$), with slower refresh rates resulting in a greater preference for single-eye blur. There was a significant effect of blur diameter ($F(3, 63) = 9.18$, $p < 0.00004$), with greater blur also leading to a stronger preference for single-eye blur. Finally, there was a significant interaction between these factors ($F(6, 126) = 6.38$, $p < 0.000007$), because the 30 Hz condition was much more influenced by blur diameter than the other refresh rates were.

For a refresh rate of 30 Hz it is clear that single-eye-blur is better, though the effect is small for the smaller blur diameters. For 30 Hz video, alternating-eye blur is probably always a poor choice. For 60 Hz and 120 Hz, there is no real evidence of a preference below blur diameters of 8 pixels. For blur diameters of 16 there is some preference for single eye-blur for both 60 Hz and 120 Hz, though the effect is not very large.

The data suggest there are a range of blur sizes (2 pixels and smaller) and refresh rates (60 Hz and 120 Hz) for which either blur approach is a reasonable choice, and should be dictated by technical concerns. Outside of that range, it appears that single-eye blur is best. There was no evidence of alternating-eye blur ever being preferred for the durations tested.

We were surprised by the general lack of preference in the 60 and 120 Hz conditions. While we did not ask on a trial-by-trial basis what subjects found objectionable about the content, we found flicker to be quite visible in the 30 Hz condition when piloting the experiment on ourselves, much more so than any other artifact in any other condition. Perhaps subjects learned to base their decision on the most conspicuous artifact (flicker), and tended to respond "no preference" or randomly when there was little flicker difference. We tested this hypothesis in Experiment 2.

### 2.3.2 Experiment 2

The method, stimuli, and paradigm for this experiment were identical to the first experiment except for two modifications. First, only a subset of the conditions from Experiment 1 were used: 4, 8, and 16 pixel blur diameters, and 60 Hz and 120 Hz frame rates. We removed the 30 Hz condition over concerns that it sensitized subjects to flicker, and the conditions with blur diameter 2 since there was little evidence of a preference with that small of a blur diameter. Second, because blur was the most obvious difference between methods to us, we instructed subjects to "check for differences in sharpness between videos" but that they "may use other differences in addition to sharpness to decide which video is preferable." To help subjects learn to pick up small differences in blur a new training sequence was created. The pairs of clips only differed in how much blur was applied, and the

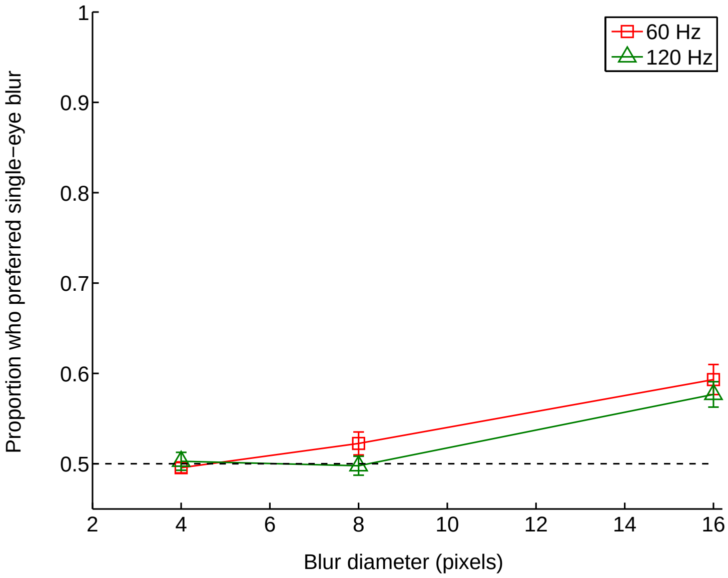


**Figure 2.5**: Experiment 2 results for naive viewers. Proportion of trials in which single-eye blur was preferred over alternating-eye blur as a function of blur diameter. Error bars denote 95% confidence intervals.

blur was applied equally to each eye, so there was an objectively correct response for each trial. Thus, we could verify that subjects were able to make fine blur discriminations. They were given up to 45 trials to demonstrate that they could discriminate between 2 pixel diameter blur and no blur; 2 subjects failed this test and did not proceed to the rest of the experiment.

Twenty new subjects completed the experiment. All of them had normal or corrected-to-normal acuity, the ability to perceive stereoscopically defined depth, and were naive to the experiment's purpose.

Fig. 2.5 shows the proportion of trials where subjects preferred single-eye blur over alternating-eye blur, as a function of blur diameter, for the two different frame rates. We did not see any consistent difference in preferences for blur type among the four different source videos, so we have combined data across that factor.

A repeated measures ANOVA found a significant effect of blur diameter ($F(2, 34) = 4.35$, $p < 0.02$), with larger blur diameters leading to more preference for single-eye blur. There was no significant effect of refresh rate ($F(1, 17) = 0.811$,

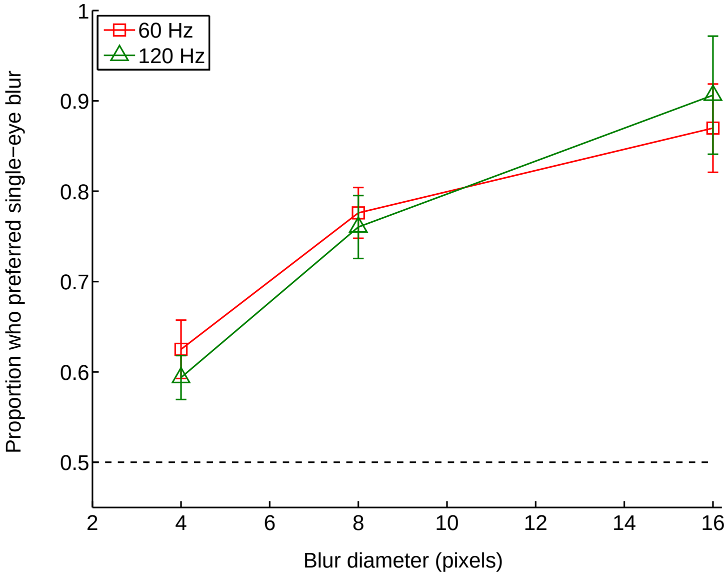


**Figure 2.6**: Experiment 2 results for expert viewers. Proportion of trials in which single-eye blur was preferred over alternating-eye blur as a function of blur diameter. Error bars denote 95% confidence intervals.

$p < 0.38$), nor was the interaction between factors significant ($F(2, 34) = 1.12$, $p < 0.34$). It appears that single-eye blur is preferable for large blur diameters, independent of refresh rate, but the effect is not very large. On the whole, the results from Experiment 2 were quite similar to the same conditions tested in Experiment 1. Thus, we can reasonably rule out the concern that the lack of preference in Experiment 1 was due to over-sensitization to flicker.

Again, these results surprised us, since during pilot work we had seen clear differences in sharpness in some of the conditions. To quantify this effect, we ran Experiment 2 again with four psychophysical experts: the three authors of [39] (including this dissertation's author) and another member of the lab who was heavily involved in creating the stimuli. Only in a few rare cases were we able to identify on each trial which movie contained which blur type, so our results should reflect the perceptual sharpness and not our expectations or biases.

The results from these four subjects are shown in Fig. 2.6. A repeated measures ANOVA again found a significant effect of blur diameter ($F(2, 4) = 103$, $p < 0.0004$), but no effect of refresh rate ($F(1, 2) = 0.06$, $p < 0.83$) or interaction

($F(2,4) = 1.48$, $p < 0.33$). These data show that single-eye blur appeared much sharper, especially for large blur diameters, and this effect was much larger than for our naive subjects.

Why would alternating-eye blur look more blurry than single-eye blur? We suggest that it is because at high frame rates the visual system tends to respond to the time-averaged luminance of multiple frames because the visual system cannot keep up with the rapid change in contrast between frames. Thus, each eye sees a relatively blurry view at the same time, leaving less opportunity for binocular suppression to allow the sharper view to dominate.

The difference between the experts and naive subjects is potentially troubling, though it is important to keep in mind that our naive subjects showed the same trend as our experts, just at a much smaller rate. The difference is almost certainly due to the expert's greater familiarity with blur. In particular, there are spatial locations in each video that are more diagnostic to blur (such as high contrast edges against a sparse background), and all of our experts independently chose to use these regions. Unfortunately, we did not ask our subjects where specifically they looked during the experiment, but it seems likely that many of them looked at other regions, perhaps those with higher personal interest (faces, people, etc). On the other hand, normal viewing behavior is more likely to be approximated by the behavior of our naive subjects than our experts. This is not to say the experts results should be ignored; instead, it probably represents a worst-case scenario that will only be experienced for specific video content where the blur-diagnostic regions happen to be the most likely to receive fixations. It would be interesting to run a follow-up experiment with an eye tracker to see how blur preference correlates with the spatial frequency content near fixation, and to compare where naive versus expert viewers choose to fixate.

An open question is if naive viewers who are exposed to these stimuli passively would eventually begin to notice the degradation in sharpness, which we address in the next experiment. Independent of that, however, Experiments 1 and 2 suggest that single-eye blur is the best choice, at least when shown over short durations. Either subjects do not care particularly, or they prefer single-eye blur. Thus, all things being equal, single-eye blur should used unless a lower bit rate can be achieved with alternating blur.

## 2.4 Fatigue Experiment

Our third and final experiment tests whether mixed resolution coded videos are tiring to watch over an extended duration. We continue our comparison of the two processing methods, but with a different technique better suited to 10-minute long viewing durations. Subjects provide a numerical rating of each video independent of the other, and we compare the average scores at the end.

Videos were shown on a full HD resolution (1920 × 1080 pixels) 47" LG 3D TV (model LW6500) using polarized glasses. Subjects were seated at a distance of five times the height of the display (about 2.9 m) as recommended in [40] for our screen size. At this distance, the screen subtends a visual angle of 20.4° × 11.5° at about 94 pixels/degree of visual angle horizontally and vertically.

### 2.4.1 Scoring

Since we are interested in the subject's temporal response and evolving comfort level throughout the video, we chose to use the Single Stimulus Continuous Quality Evaluation (SSCQE) method for this study [40]. Subjects evaluated each video solely on the basis of visual comfort level on the following scale: 5 - Excellent (very comfortable), 4 - Good, 3 - Fair, 2 - Poor, 1 - Bad (very uncomfortable). Using the system presented in Chapter 3 and [41], subjects recorded their scores by adjusting a slider on a mobile device that was sampled two times per second. An auxiliary monitor next to the TV displayed the scoring scale and the subjects' currently selected score. This setup was designed so that the subjects could keep their eyes on the TV for the entire duration of the video, as was emphasized in the test instructions.

In addition to the SSCQE evaluation, at the end of each test video, we asked subjects to indicate which eye felt more fatigued: left, right, or neither (same fatigue level in both eyes). The purpose of this final question was to see if subjects

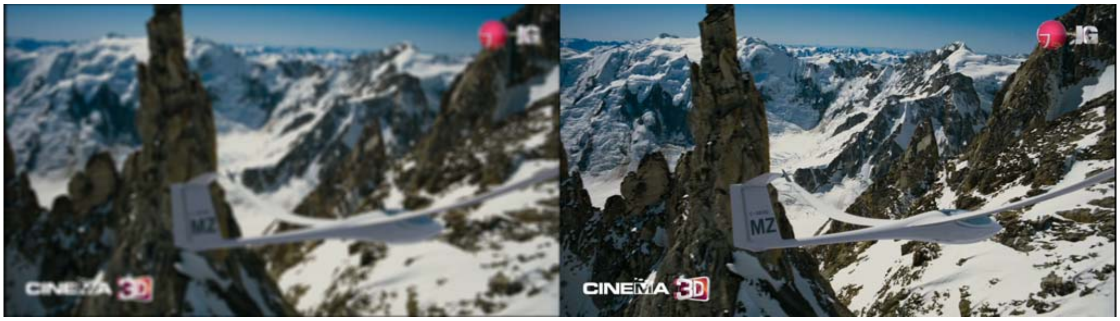


**Figure 2.7**: One frame from fatigue experiment test video, set up for cross-eyed viewing. A viewing distance of about 21 centimeters will provide the approximate equivalent spatial resolution as displayed during the experiment.

were able to notice any difference in strain between their eyes, an important factor in evaluating the two methods of binocular suppression.

### 2.4.2 Preparing the Test Videos

A 20-pixel diameter was chosen for the disk filter to introduce an extreme condition for binocular suppression. Previous work has shown that a stereo image pair comprised of one sharp image and one image bandlimited to half of the Nyquist rate (downsampling by a factor of 2 in each direction) shows virtually no sharpness loss as compared to the full resolution stereo pair [11]. Our recent study also showed similar results for stereo video and even more severe blur, down to 20% of the Nyquist rate [12], possibly owing to a motion sharpening effect [42]. While these studies do not address the comfort level of viewing such images or videos, we use them as guide with the underlying assumption that if subjects do not notice the difference between mixed resolution and full resolution content, then the former will be mostly comfortable to view.

The blur kernel in this study has a cutoff frequency of 12% of Nyquist (a downsampling factor of more than 8 in each direction), again taking the bandlimit as the location of the first zero in the frequency response as in Section 2.3. The 20-pixel diameter corresponds to 12.7 arcmin, larger than the kernels used in Section 2.3. The results of those experiments suggest that this blur diameter would be sufficient for subjects to notice a difference between the two blur types. This larger kernel is necessary for the slightly greater viewing distance than in [12], and is meant to induce some discomfort so that the subject response scores are not overly saturated. A sample frame is shown in Fig. 2.7.

Based on the results of [12] and Section 2.3, we choose to encode and present the videos at 60 Hz to minimize flicker artifacts. Since the test videos need to be high quality and 60 Hz, we chose source videos that were high resolution, high bit rate, and either high frame rate or slow motion. We used the *LG Demo*, *Grand Canyon*, *Looney Tunes*, and *Rome* sequences from [43], which contain a mix of animation and real footage, slow and fast motion, colors and brightness, and depth and texture (see Table 2.2). Each of the source videos were compressed with H.264 in the YUV colorspace with 4:2:0 subsampling at a bit rate of at least 19.8 Mb/s. Clips from these sequences were selected such that the motion did not look jerky or unnaturally fast when played at 60 Hz. These clips were compiled into a single source video that was 5 minutes long[2].

All of the source videos were encoded in full HD side-by-side format. To prepare the test videos, the left and right frames were extracted from each clip and upsampled to full HD using bicubic interpolation. Next, the frames were blurred according to the two methods of binocular suppression as in Section 2.2.1 using the blur kernel described in Section 2.2.2 with the diameter described above. Due to the side-by-side encoding and upsampling, the perceived spatial frequency difference between the filtered and unfiltered views would not be as large as if each source view was full HD from the start. However, the blur kernel was so large that the blur difference is very perceptible (see Fig. 2.7).

Since the intended display device was a polarized 3D TV, the content would be vertically subsampled so that the two views could be multiplexed on the screen with alternate lines having different polarizations. Thus, the left and right frames of the sequence were then subsampled vertically and combined into a single frame in up-down format. The sequence of frames was encoded into a single MP4 file at a bit rate much higher than that of any of the source videos. No additional

[2] Clips were taken from each source video with the following (start time, duration) in frames: *LG Demo*: (480, 661), (2566, 313), (2998, 312), (4244, 2781), (13860, 1631); *Grand Canyon*: (1217, 500), (2600, 418), (3956, 951); *Looney Tunes*: (232, 8333); *Rome*: (6593, 2100).

**Table 2.2**: Source clips used in the fatigue test video and their characteristics.

| Clip | Duration (MM:SS) | Description |
|---|---|---|
| *LG Demo* | 01:35 | 4 separate scenes containing a mix of real and CGI imagery; diverse in color, depth, motion |
| *Grand Canyon* | 00:31 | Panning views of natural scenery, little motion; predominantly tan and light blue colors for the canyon and sky; moderate depth |
| *Looney Tunes* | 02:19 | Animated sequence, some fast motion; high contrast, bright colors, varied depth |
| *Rome* | 00:35 | City scene of street entertainers in a plaza; vivid depth, bright colors, moderate motion |

compression artifacts were introduced into the test videos.

### 2.4.3 Test Procedure

Each 5-minute test video was looped twice to produce a 10-minute exposure time per video. In between test videos, subjects were asked to look at a gray screen for 2 minutes to normalize their vision before beginning the next test. Because there were only two test videos, the order in which they were played was alternated for each subject. Before the test, subjects were given written instructions describing the scoring and test procedure. After reading the instructions, they were given a chance to ask any clarifying questions they wanted.

To familiarize subjects with the scoring system and test procedure, a practice test was given before the actual test. The two practice test videos (one of each blur method) were prepared in exactly the same way as the actual test videos, but were only 90 seconds long and contained different content. After the practice test, subjects were given a chance to ask questions about the procedure. They were then asked to accommodate to a gray screen for one minute before beginning the actual exam. A diagram showing the overall test procedure is given in Fig. 2.8.

The full test lasted about 30-35 minutes. Subjects were drawn from the general university population and were unaware of the types of processing performed

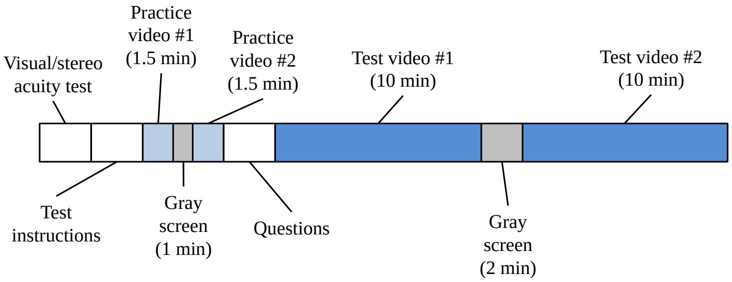


**Figure 2.8**: Diagram of procedure for fatigue experiment.

on the videos. Subjects were screened for natural or corrected 20/20 visual acuity in each eye. Stereoacuity was tested by asking subjects to identify rectangles of positive and negative disparaties in a random dot stereogram, as well as the disparate circle among groups of four down to a 2-pixel disparity. 22 subjects, 9 male and 13 female, ranging in age from 16 to 35 years, participated in the experiment.

### 2.4.4 Results and Discussion

The mean scores across subjects over the duration of the test videos are shown in the upper plot of Fig. 2.9. Outlier detection was performed as specified in [40] for the SSCQE method, but no subject data had to be discarded. For both of the tested methods, the mean scores generally range from 3 to 4 (fair to good comfort). The alternating blur method is rated higher, with an overall mean of 3.86 compared to 3.74 for the single-eye blur. However, the results of a paired sample t-test yield a $p$-value of 0.09, greater than the threshold of 0.05, showing that the difference in means between the two methods is not statistically significant. Here, the temporal mean was computed for each subject and test method across the full 10-minute duration of the video, and the two groups of means were compared. A permutation test [44] that uses all of the time samples for all subjects, then computes the $p$-value from the probability distribution of individual time sample test statistics, yields similar $p$-values. The 95% confidence intervals on a per-sample basis (not shown, to simplify the plots) mostly ranged between 0.25 and 0.45, with a mean of about 0.35, for both the single-eye and alternating blur cases. Note that scores for the first 5-10 seconds of exposure may be biased as subjects adjusted the slider from its initial position at the "3" mark.

The bottom half of Fig. 2.9 shows the mean score computed across subjects and time for each clip of the video, along with the 95% confidence intervals for each estimate. The dashed lines in the figure indicate the ends of each of the four clips, and the dotted lines indicate a scene change within the *LG Demo* clip (not all scene changes within this clip or the others are shown). In addition to the general preference for alternating blur over time, there is some dependence of preference on the type of content as certain clips are less straining to watch than others. For instance, there is a marked decrease in scores for both blur methods over the course of the second scene in the first clip for both loops of the video (between the dotted lines, from about 0:10–0:30 and 5:10–5:30). As soon as this scene is done playing at about 0:30 and 5:30, the scores escalate for the remainder of the *LG Demo* clip.

The *Looney Tunes* clip from about 2:05–4:25 and again from 7:05–9:25 reflects the largest difference in scores between the two methods, with the alternating blur being preferred. In fact, for the first presentation of this clip, the t-test $p$-value is 0.04, showing that the difference in means is statistically significant. The animation contains high contrast, flat textures, and sharp edges. All of these features produce artifacts under asymmetric blur, which are quite salient in the single-eye blur case. Since each eye sees a somewhat time-averaged video in the alternating blur case, the blur is more symmetric across the eyes and the effects of the artifacts are mitigated. Presentation at 60 Hz greatly masks the flickering of temporally changing spatial frequencies in the alternating blur case. Whatever flicker remains, particularly in high frequency, high contrast regions, appears to be less bothersome then the persistence of a spatial frequency discrepancy between the two eyes, as in the single-eye blur case. Although flicker would be more noticeable in these regions as well, it is possible that people accommodate to flicker easier than the spatial artifacts.

Since each 10-minute test video was actually a 5-minute video shown twice, we can compare the scores for the two halves of each test to see trends over the duration of the video. The mean scores for each half-presentation are shown in

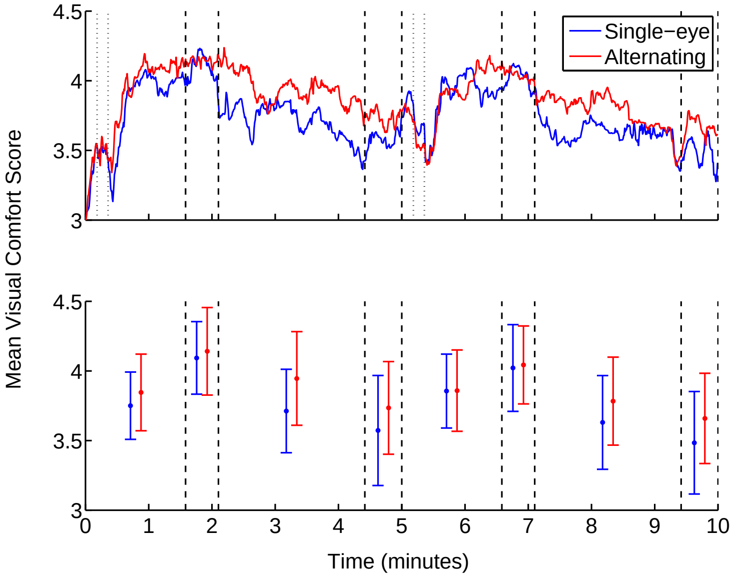


**Figure 2.9**: Mean scores across subjects over duration of video for each processing method (top), and mean scores across subjects and clip duration with 95% confidence intervals (bottom). Dashed lines separate each video clip in the sequence.

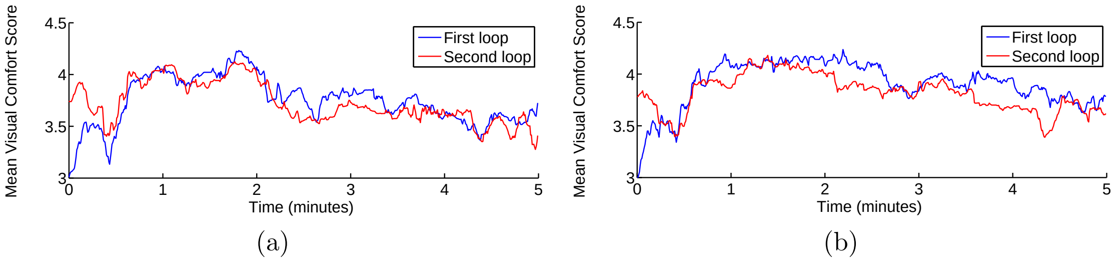


**Figure 2.10**: Mean scores across subjects separated by loops of the stimulus video, for each coding method. (a) Single-eye blur video. (b) Alternating-eye blur video.

Fig. 2.10. The overall means for each half are listed in Table 2.3 as well as the percentage change in scores from the first half to the second half. This table also lists the $p$-value resulting from a t-test between the first and second loop within each method. Although the mean scores for the second loop of the video were lower than those for the first loop in each method, the difference was small and not statistically significant. Thus, over this length of exposure, there is no degradation in comfort over time.

The results for the three-alternative forced choice question are shown in

**Table 2.3**: Statistics for the two methods by halves.

| Clip | First Half Mean | Second Half Mean | Percent Change | t-test $p$-value |
|---|---|---|---|---|
| Single-eye | 3.75 | 3.72 | -0.60 | 0.73 |
| Alternating | 3.91 | 3.82 | -2.3 | 0.34 |

Fig. 2.11. The majority (14 of 22 subjects) felt that their eyes were equally fatigued after watching the alternating blur video, while the rest felt that one eye was more fatigued, in equal proportion. This makes sense, since there is no reason to expect one eye to be more fatigued than the other using this blur method. For the single-eye blur case, 7 subjects reported equal fatigue in each eye, which suggests that they may not have noticed the blur or at least were not bothered by it. Of the remaining 15 subjects, 9 felt that their right eye was more fatigued (the eye shown the blurry video) and 6 felt their left eye was more fatigued.

It is surprising that a fair portion of subjects (27%) felt that their left eye was more fatigued, which represents 40% of the subjects who felt asymmetric fatigue. This data contradicts our hypothesis, and that of other researchers [26,28], that the eye receiving the blurry view would be more fatigued. Although 3 more subjects felt fatigue in their right versus left eye, suggesting that the blurry view may cause some fatigue, a difference of 3 subjects is hardly significant. A possible explanation for the data is as follows. For mixed resolution stereo video, each eye receives an image at a different focus. While the visual system weights its fused perception of the world toward the sharper view, it is not able to determine which eye has the “correct” focus. Visual fatigue can have many causes [45], and one eye may tire more than the other as muscles controlled by the visual system try to adjust focus.

The main point the histogram data conveys is that for alternating blur, there is a clear balance of fatigue between the two eyes. For single-eye blur, one of the two eyes becomes more fatigued than the other, but the fatigued eye is not correlated with the blurry view. Comparing equal fatigue in both eyes to more fatigue in one eye, the alternating blur method produced equal fatigue in both eyes about 64% of the time and asymmetric fatigue the remaining 36% of the time. The

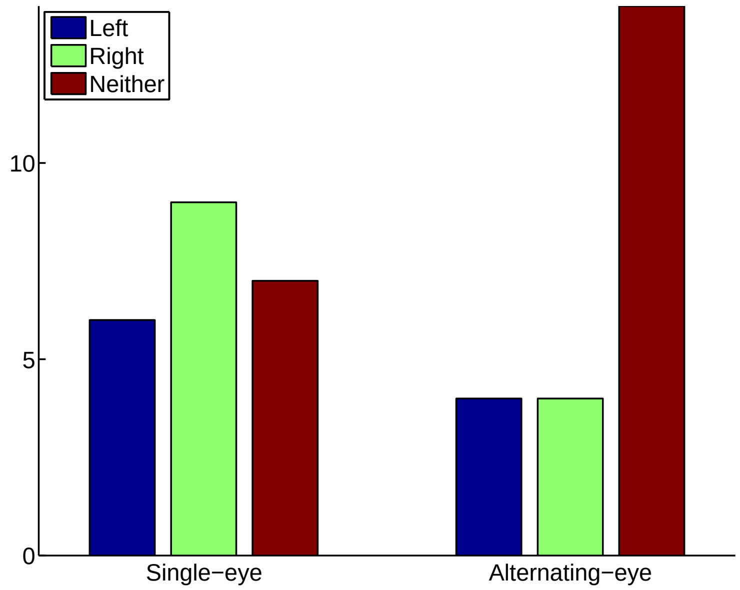


**Figure 2.11**: Histogram of responses to the question "Which eye is more fatigued?", asked to subjects following each test video.

proportions for the single-eye blur method are almost equal but flipped, with equal and asymmetric fatigue occurring about 32% and 68% of the time, respectively.

Note that the mean scores for both test cases were between 3 and 4 (fair to good comfort). These are relatively high scores considering the level of blur applied to the sequence. As noted in Section 2.2.2, the filter bandwidth is about 12% of Nyquist, corresponding to a downsampling ratio of 8.3 in each dimension or 69.4 overall. This downsampling ratio corresponds to an image that nominally contains only 1.4% of the data of the original. For full HD imagery, the downsampled image would have a resolution of $230 \times 130$ pixels as compared to the $1920 \times 1080$ original.

As intended, the chosen filter simulates quite an extreme level of downsampling; in practice, a less severe ratio would likely be used. Less blurring means that the two views would be more symmetric in the single-eye blur case, making it more difficult to sense the difference between the two views. For the alternating blur case, there would be less flicker. For both methods, a smaller downsampling ratio would lead to even higher comfort scores.

## 2.5 Discussion

Our results show that in terms of quality, there is a preference for single-eye blur but in terms of comfort, either method is acceptable. Generally then, for images and short videos, single-eye blur will be the better option. For longer videos, however, the proper choice of method depends on the content, other parameters of the video, and the artifacts introduced by each encoding method. The flicker artifact inherent in the alternating blur method is due to the changing amplitudes of high spatial frequency components from frame to frame. The severity of this artifact will be inversely related to the bandwidth of the blur filter as well as the frame rate of the video. As our results show, the perceived quality of videos produced in this manner remains quite high for relatively large blurs at 60 Hz and beyond. Owing to the persistent blur in one eye, the single-eye blur method runs the risk of viewers becoming aware of the asymmetric quality. This awareness might degrade the overall viewing experience in a way that is not reflected in a controlled experiment where subjects are asked to focus solely on the experiment variable. However, even relative experts, such as this dissertation's author and colleagues, are unable to tell whether they are viewing single-eye blurred videos versus full resolution videos for modest blur levels and durations without resorting to the trick of covering one eye at a time. In any case, the choice of method will also depend on blur level and frame rate, and the length of presentation should perhaps be more heavily weighted than these results can show.

Compared with the traditional method of single-view blur for binocular suppression, the primary advantage of the alternating blur method is that the blur is balanced equally across both eyes. For viewers who have particularly bad vision in one eye, probability dictates that single-eye blur will deliver the blurrier view to the good eye for half of these viewers. For scenarios that commonly have a single viewer, such as videos on news websites or video conferencing, the viewer could switch the blurry view through a preferences dialog. A final option would be to show the same sharp view to both eyes. Since viewers with a weak eye are unlikely to have good stereopsis, the image sharpness could be preserved at the expense of the depth signal, to which they may not anyway be sensitive. In contrast, the alternating blur method successfully balances the blur across both eyes for a less-straining visual experience and will still appear pleasant even for viewers with one weak eye. In this case, the view corresponding to the viewer's weak eye would appear blurrier than the strong eye's view. Since at higher frame rates the alternating video would produce no perceptible flicker, the final fused percept would be spatially and temporally homogeneous in terms of sharpness, and would tend toward the sharper view as in standard binocular suppression.

The alternating blur method is superior to other methods that seek to balance the blur across both eyes. In [24], alternate horizontal slices of each view are blurred. If a viewer has one weak eye, a video prepared according to [24] would appear spatially alternately sharp and blurry. The proposal in [25] to switch the blurry view at each GOP, taken as 4 frames in their study, would induce very perceptible flicker. Since the frequency of alternation is reduced by a factor equal to the GOP size, the frame rate required to mask the flicker artifact would be driven up by the same factor. Thus, while this method is convenient in terms of compression, it is not practical to require playback at 240 Hz or greater across applications and devices. The alternating blur method is also not content-dependent, which is an advantage over the processing schemes in [26, 27].

Finally, there may be applications for which one method is preferred over another. The single-eye blur method is likely better for compression due to the large temporal correlation in each view and simpler codec design. It could also be suitably applied to scalable video coding, where devices would receive a high resolution stream for 2D viewing and 3D-enabled devices could also decode the auxiliary low resolution stream for 3D viewing. The alternating blur method, however, may be more amenable to decoder-side processing where having high resolution data in each view would be helpful. If one wanted to super-resolve the mixed resolution video, for instance, each low resolution frame would have 3 high resolution immediate neighbors (the preceding and subsequent frame, and the cotemporaneous frame from the other view) to borrow pixels from, but only one such neighbor in the single-eye blur case. Similarly for disparity estimation, the disparity could be estimated at the lower of the two resolutions, then upsampled using a method such as joint bilateral filtering [46] where the guidance image is a motion-compensated neighboring high resolution frame.

## 2.6 Conclusion

In this study, we compared two methods of mixed resolution coding, single-eye and alternating-eye blur, in terms of overall quality for short exposures and visual fatigue level for long exposures. For short videos, viewers prefer single-eye blur for moderate to large amounts of blur and for low frame rates. For small blur levels and frame rates 60 Hz or above, there is no preference. For long exposures, both methods were equally comfortable to view, but there was some preference for alternating-eye blur for animated scenes.

Prior work that focused on balancing the blur between the two eyes did so for two reasons: 1) to induce less eye strain, and 2) to deliver quality video to viewers with one weak eye. Our results show that the first reason is largely unfounded; subjects did not experience more visual fatigue when viewing single-eye blur than when viewing a balanced blur method such as alternating blur for most scenes. As for the second reason, the balanced delivery of blur to both eyes means that the alternating-eye method is better for viewers with one weak eye. This method is also superior to previously proposed approaches to balancing blur across both views [24–27], as discussed in Section 2.5.

Overall, both methods of mixed resolution coding appear viable for stereo compression, in terms of quality and visual comfort. It should be noted that in practice, smaller downsampling ratios will likely be used than simulated here, which would create fewer artifacts and less asymmetry. This change would ostensibly increase viewing comfort even more, so that the scores reported here are a sort of lower bound on comfort for practical use. The results dictate different use cases for each method, depending strongly on the viewing time, frame rate, and amount of downsampling. The intended application or post-processing pipeline also bears on the choice of encoding method. Our results also indicate some dependence of comfort ratings on the content being encoded.

One interesting avenue for future development would be to further investigate this dependence. As we noted, alternating-eye blur is more suited for animated content, but it may be that single-eye blur is better for other content where flicker artifacts are more salient. Another area of potential research would be to implement an efficient coding method for the alternating blur case. One could also design an integrated codec that, based on the scene content, viewing parameters, and desired bit rate, would decide upon the best encoding method. The method could be switched at scene cuts so that the change is masked [27].

## Acknowledgments

We thank Matt Scott for assistance in preparing the stimuli, and Donald MacLeod for helpful discussions and proofreading.

The text of Chapter 2 is, in part, a reprint of a paper that has been published as: A. K. Jain, A. E. Robinson, and T. Q. Nguyen, “Comparing perceived quality and fatigue for two methods of mixed resolution stereoscopic coding,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 3, pp. 418–429, Mar. 2014. The dissertation author was the primary investigator and author of this paper.

# Chapter 3

# Tally: A Web-Based Subjective Testing Tool

In image and video processing, subjective rating of stimuli is often used when a clear perceptual model of the stimuli does not exist. Researchers show images or videos to human subjects that are processed with some parameter set, ask them to rate features of the stimuli (such as quality or sharpness), and then correlate the responses with the input parameters. The derived relationship can be used to develop metrics, evaluate quality, or design algorithms as a function of the input parameters. Thus, subjective testing is a critical part of such research.

Subjective testing, from designing the experiment to acquiring, scheduling, and running subjects, can be a slow and difficult process. Inherent in this is the rapid growth of test conditions with the number and range of input parameters. Each parameter takes on multiple values, resulting in a number of test conditions equal to the product of the cardinality of each parameter set. Further, each test condition must be evaluated multiple times by each subject to obtain reliable statistics.

The difficulty introduced by the dimensionality of the test is exacerbated by the lack of proper data collection tools. For our modest-sized study in [12], over 1800 scores had to be collected by hand. For the fatigue experiment in Chapter 2 where scores had to be recorded twice per second, even resorting to laborious manual data collection was not an option.

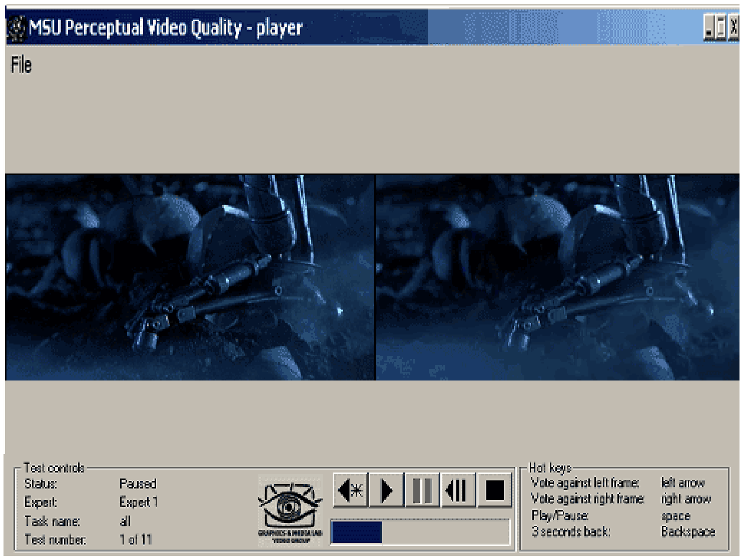
To enable the experiment in Chapter 2 and ease the tedium of studies like [12], we developed Tally: a new, web-based system for conducting subjective video quality experiments. We designed this tool to make subjective testing easier, faster, and more collaborative, and have made it publicly available. Our system can save researchers time in collecting and analyzing data while dramatically decreasing the potential for error. Tally is free, open source, and cross platform, and supports common testing methods.

In this chapter, we present the Tally system. First, we review current testing approaches and their shortcomings in Section 3.1. We then discuss the design principles and major features of our system in Sections 3.2–3.3, as well as the main components and implementation details of the tool in Section 3.4. A basic operating guide is provided in Section 3.5 along with links to download the tool in Section 3.6. Finally, we briefly summarize, discuss future extensions, and conclude in Section 3.7.

## 3.1 Other Testing Systems

Commonly, subjective tests are conducted by asking subjects to write their responses using pen and paper, which are then manually entered into computerized spreadsheets and analyzed. Not only is this process extremely slow, it is also prone to error. Some attempts have been made to automate this process, which requires researchers to write custom software. Many times this software is not applicable to different test scenarios, and may not be extensively tested if it is only used by a small group of people. Additionally, labs generally have not made their source code or even their compiled software readily available, which limits its effectiveness. Further, the lack of a unified tool for data collection compromises the research community's ability to repeat experiments and disseminate results.

Previous software tools for subjective testing displayed the test images or videos in the same window as the voting interface. Beyond the simplicity of development, this design has few benefits, and in fact has a number of severe limitations. First, this kind of system would require one to design display-specific user inter-



Test controls

| | |
|---|---|
| Status: | Paused |
| Expert: | Expert 1 |
| Task name: | all |
| Test number: | 1 of 11 |

Hot keys

| | |
|---|---|
| Vote against left frame: | left arrow |
| Vote against right frame: | right arrow |
| Play/Pause: | space |
| 3 seconds back: | Backspace |

**Figure 3.1**: MSU Perceptual Video Quality tool. The display and voting control are within the same interface.

faces for enabling both 2D and 3D. This can be a tedious task, especially since 3D formats are still evolving. Second, these systems do not allow for natural viewing of content at a distance. The subject must be at the computer terminal to respond to questions, or must control a keyboard/mouse from a far viewing distance. Third, previous software solutions only allow media to be displayed at limited resolutions. Since the voting interface requires some screen real estate, a full HD video, for example, could not be shown on an HD TV using such a system. Finally, since the voting control is coupled with the media display, these systems only allow the testing of one subject at a time.

An example of software for subjective tests developed by a research lab is given in [47]. A subjective testing tool developed by Lomonosov Moscow State University (MSU) [48] is shown in Fig. 3.1. Even this tool, a very good product designed for research use and public distribution, suffers from the limitations discussed above.

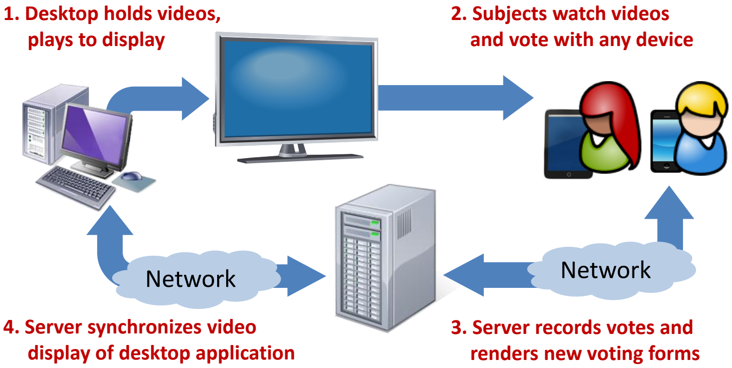

1. Desktop holds videos, plays to display
2. Subjects watch videos and vote with any device
Network
Network
4. Server synchronizes video display of desktop application
3. Server records votes and renders new voting forms

**Figure 3.2**: Workflow of the Tally system.

## 3.2 System Design and Overview

We designed Tally as a web-based system, a choice that addresses the limitations of hand-written tests and previous software solutions discussed in Section 3.1.

Our system consists of three major pieces: the desktop application, the server back end, and the web front end. The desktop application is installed on a local machine that is connected to the display device, and is responsible for displaying the videos through a media player to the screen. The web front end allows the researchers to create and run experiments, and allows subjects to cast their votes during subjective tests. The server back end controls all operations between the components and records data in the database. The server can be running on the local machine on which the desktop application is installed, or a different machine in any location.

The basic workflow of a subjective experiment is depicted in Fig. 3.2. Subjects log into the web front end (website) and select the appropriate test to begin. Once they are ready, the desktop application receives a command from the server, and it plays the first video to the display device. Note that only the file name is sent across the network; the actual videos are stored locally on the machine connected to the display. Subjects then vote on the video using any web-enabled device such as a smartphone, tablet, laptop, or desktop, and their scores are transmitted to the server and recorded. Once the video is done playing, the server tells the desktop which video to play next, and the process repeats until all test videos have been shown.

At its core, Tally is a voting application for videos. Beyond this basic structure, most any component can be swapped or customized: any web-enabled device can be used to vote, most any video player can be used to play the media, and any display device can be used to show the media. We natively support the Double-Stimulus Impairment Scale (DSIS), Double-Stimulus Continuous Quality Assessment (DSCQS) Type II, and Single-Stimulus Continuous Quality Evaluation (SSCQE) standard methods of the ITU outlined in [40], but also allow for custom test methods to be added.

## 3.3 Main Features

Tally's web-based design offers many benefits as a subjective testing tool. Collected data is available from anywhere, and sharing data is simple. Since each person has their own personal account on the website, many people can use the same system with their individual history and data securely saved and privately accessible. We discuss some of Tally's features here.

### 3.3.1 Decoupled Display and Voting Interface

Since subject scoring is done over a network, the voting control is decoupled from the media player. The media can be either 2D or 3D and at any resolution that the display supports. Since the voting control is separate from the display computer, there are no physical limitations on the orientation of the subjects with respect to the display. Another great advantage of our web-based approach is that any number of subjects can be tested simultaneously, and all of their scores will be recorded and synchronized with the test progression. This ability can drastically reduce testing time; even if only two subjects are tested simultaneously per instance, testing time will be reduced by 50%.

### 3.3.2 Open Source

While Tally is fully functional out of the box, we are distributing the tool free of charge and open source to allow researchers to customize the tool to suit their needs. We have designed the tool with developers in mind, making sure that our code is as modular and general as possible. Minor changes, such as the way instructions are worded, the appearance of the pages, or what demographic data is collected, are straightforward to make. We also anticipate people will want to make more complex extensions, such as adding new test methods, and have designed the code to allow for that.

Releasing our tool as open source will also allow a community of users and developers to grow around it. As developers make changes and additions, we hope that they will make their improvements available to the other researchers. This will improve the tool as its uses grow, ensuring that it remains extensively tested, documented, and supported.

### 3.3.3 Robustness and Security

Our testing tool is very robust and works well on most any device, web browser, and platform. In addition to being constructed from robust and well-tested tools (see Section 3.4), we have tested the overall system on a variety of platforms and devices. We have also made considerations for security of our site and data. Despite expecting relatively small-scale usage in safe environments, we have built our site with production-level security precautions for web attacks. Through our login and permissions system, the data is only accessible to those who collected it or with whom it was shared. The database can also be easily backed up on another machine or data drive to minimize risk of loss.

### 3.3.4 Repeatability and Collaboration

The system saves all data—subject information, test parameters, and subject responses. It has an easily navigable interface to view one's test history and data, and records when tests were created and run. This automated bookkeeping makes it simple for the user to repeat an experiment with the exact same parameters or add to the test data with more subjects at a later time.

Further, the fact that our tool is web-based enables remote collaboration. We have built in a sharing system that encourages this type of collaboration. When a user creates a test, he can share it with other researchers in the system. They can then run this test on their own, and the system will aggregate all of the data belonging to the same test. For instance, two collaborating labs could run identical or complementary experiments in different parts of the world and easily combine and share their data. Alternatively, a user can choose to share only the data from a test, rather than the entire test itself, so that others may download the data. This feature can be useful when making data available for published studies.

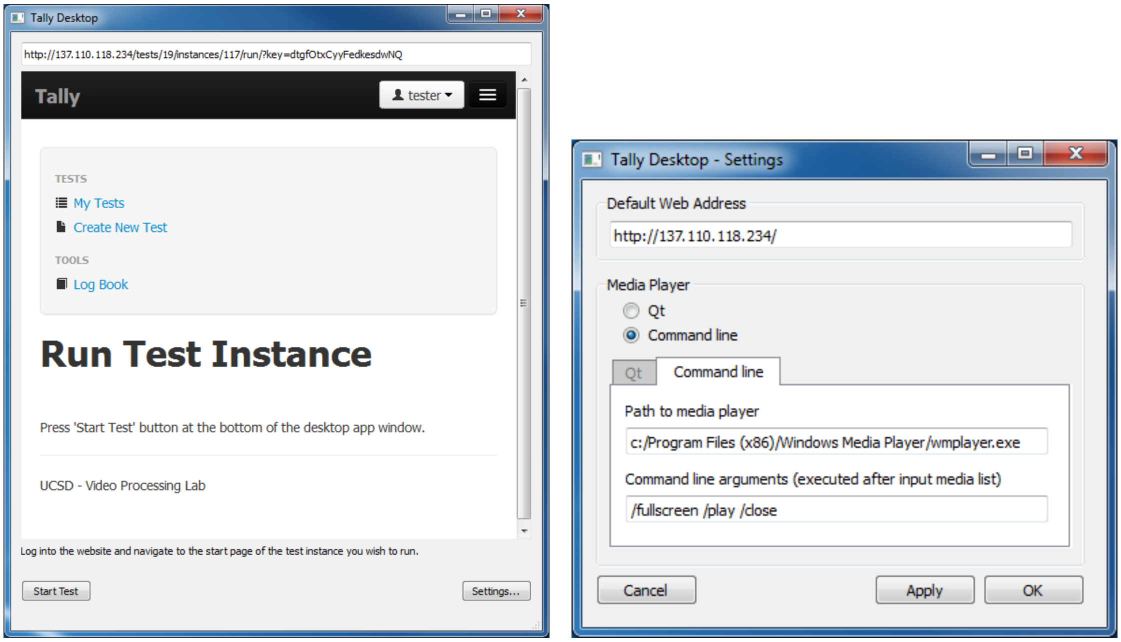
## 3.4 Components and Implementation

All components were built from commonly used, well documented, free, open source materials, so that distribution and modification of our tool will be as easy as possible.

### 3.4.1 Desktop Application

The desktop application resides on the local computer that holds the media files. Its main responsibilities are to receive commands from the server telling it which video to play, launch a video player to play the appropriate video to the screen, and notify the server when the video is finished. A screenshot of the desktop application and its settings menu is given in Fig. 3.3.

It is implemented in C++ using the Qt libraries [49]. It uses JsonCpp [50] to parse commands from the server and to encode signals to be sent to the server. The desktop application natively supports the Qt Phonon media player, as well as any command-line controllable media player, such as Windows Media Player or VLC media player [51].



http://137.110.118.234/tests/19/instances/117/run/?key=dtgfOtxCyyFedkesdwNQ

Run Test Instance

Press 'Start Test' button at the bottom of the desktop app window.

UCSD - Video Processing Lab

Log into the website and navigate to the start page of the test instance you wish to run.

(a)

Default Web Address: http://137.110.118.234/

Media Player: ○ Qt ● Command line

Path to media player: c:/Program Files (x86)/Windows Media Player/wmplayer.exe

Command line arguments (executed after input media list): /fullscreen /play /close

(b)

**Figure 3.3**: Software that runs on the desktop computer that contains the videos and is connected to the display. (a) Desktop application (b) Settings menu.

### 3.4.2 Server Back End

The server is the hub of our system and coordinates all aspects of the testing tool. The server collects responses from subjects and records them in the database, tells the desktop application which video to play and when, handles test case randomization and data unfolding, and renders the webpages for the users.

The back end was built with Python [52] using the Django framework [53]. We chose to use SQLite as our database system because it naturally integrates with Django and Python, and does not require a separate database server to be run. However, it is also possible to use any other database system compatible with the Django framework. The final component of the back end is a web server that runs the code and handles requests. The only requirement the server must satisfy is that it is able to run Python code. Gunicorn [54] for Linux or Apache + mod_wsgi for Windows are two such examples. The HTML pages for the website are rendered using Django's template system and styled using different packages as described in Section 3.4.3.

We chose the Django framework for three reasons. First, Django is full web framework with many useful libraries built in, such as tools for authentication, security, and static file handling. Most notably, Django automatically provides an administrator interface which allows the site manager to have complete control over the database should anything go wrong. These features minimized our coding effort as well as that of future developers of our system.

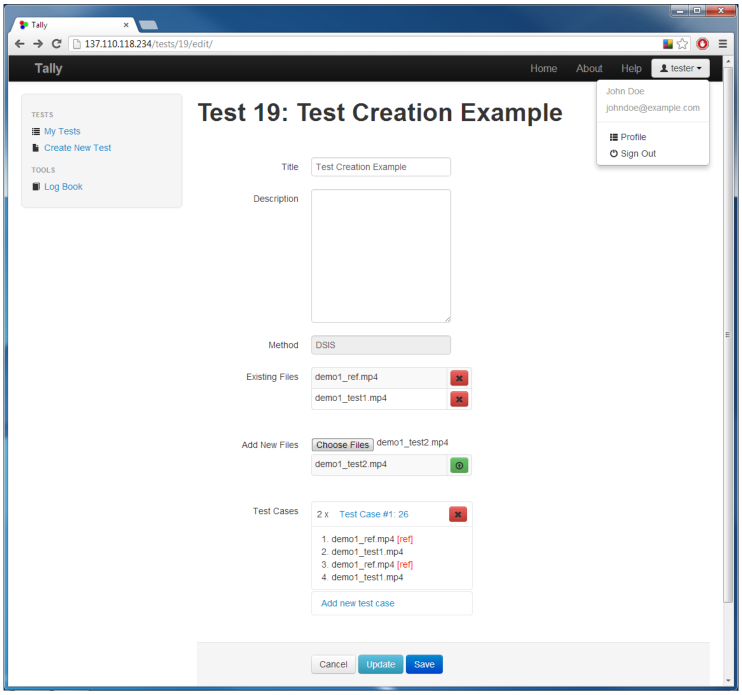
Second, Django is a stable and reliable framework that has been deployed on many large-scale sites. Reliability is critical for our application, and Django has been through several releases and has an active user community.

Third, Django is built from Python and is extremely well-documented. Since Python is a commonly used language, the barrier to entry for others wanting to develop our system is lower than for a different web framework, such as Ruby on Rails [55], which relies on the lesser-used Ruby language [56]. The excellent documentation of both Django and Python also eases the development for the user community that we hope will grow around our system.

### 3.4.3 Web Front End

The front end is the website that serves as the interface to the subjective testing tool. Users of the site are divided into two groups, Testers and Subjects. Testers are those who administer the subjective tests, and Subjects are those who participate in them. Anyone can register as a Subject, but Tester accounts can only be created with the permission of the site administrator (i.e. whoever downloads and sets up Tally). Each group has a different set of permissions and has separate parts of the website available to them upon login.

Testers use the website to create new tests, create instances of existing tests, share and export data, and run experiments. The pages of the Tester portion of the website were designed with Twitter Bootstrap [57], a front end framework that provides professional-looking CSS, JavaScript plug-ins, and HTML components without a lot of coding. We expect that most users will access the Tester part of the site from a desktop since it involves selecting video files or downloading data files. However, the site can still be accessed from mobile devices and appear quite



John Doe
johndoe@example.com

Test 19: Test Creation Example

Title: Test Creation Example

Description:

Method: DSIS

Existing Files: demo1_ref.mp4<br>demo1_test1.mp4

Add New Files: demo1_test2.mp4
demo1_test2.mp4

Test Cases: 2 x Test Case #1: 26

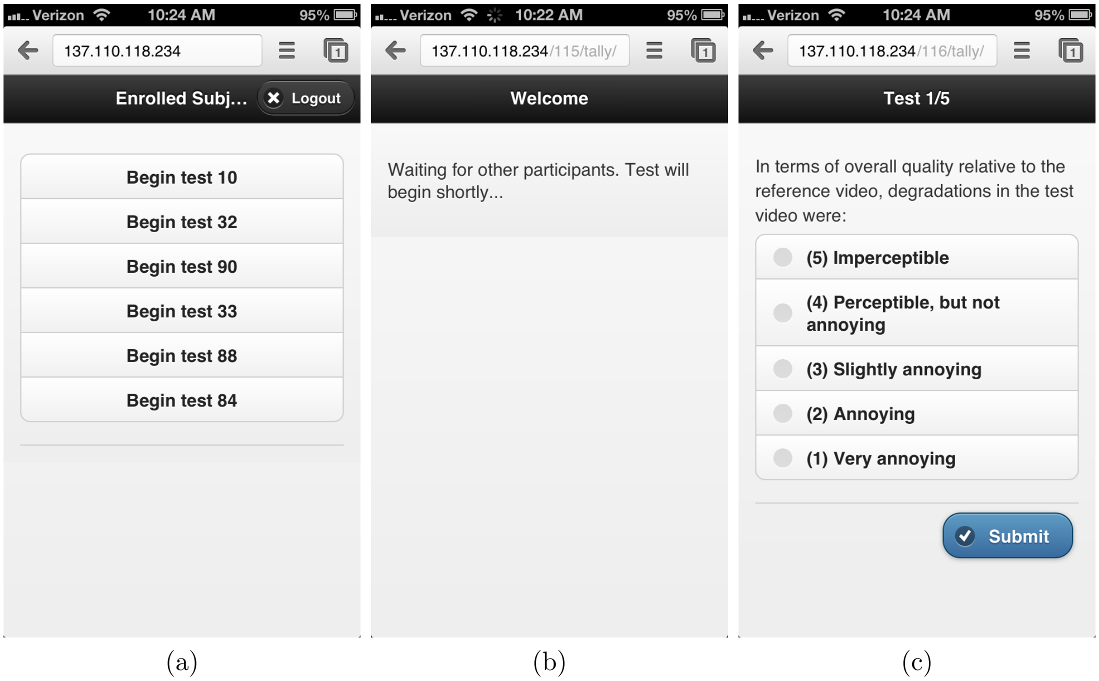
1. demo1_ref.mp4 [ref]
2. demo1_test1.mp4
3. demo1_ref.mp4 [ref]
4. demo1_test1.mp4

**Figure 3.4**: Screenshot of the webpage a Tester would use to create a test.

pleasing due to Bootstrap's flexible page structure. An example test creation page of the website is shown in Fig. 3.4.

Subjects use the site to select a test instance in which they are enrolled and provide responses when prompted as the test progresses. Since we envision subjects will mostly be using mobile devices (smartphones or tablets) to respond to the various test questions, we used jQuery Mobile [58] to style the pages for Subjects. jQuery Mobile is a touch-optimized framework built on top of the jQuery JavaScript library, providing plug-ins and CSS specifically designed for mobile devices. Of course, the Subject portion of the site can still be accessed via non-mobile devices.

When subjects login to the website, they are taken to a page with a list of tests in which they are enrolled. Once they click on a test, they are taken to the welcome page where they are instructed to wait until the rest of the subjects are ready to start as well. Once the test begins, Subjects are directed to the voting page. The webpages for this sequence of steps are shown in Fig. 3.5, where the voting interface is for the Double-Stimulus Impairment Scale (DSIS) test method

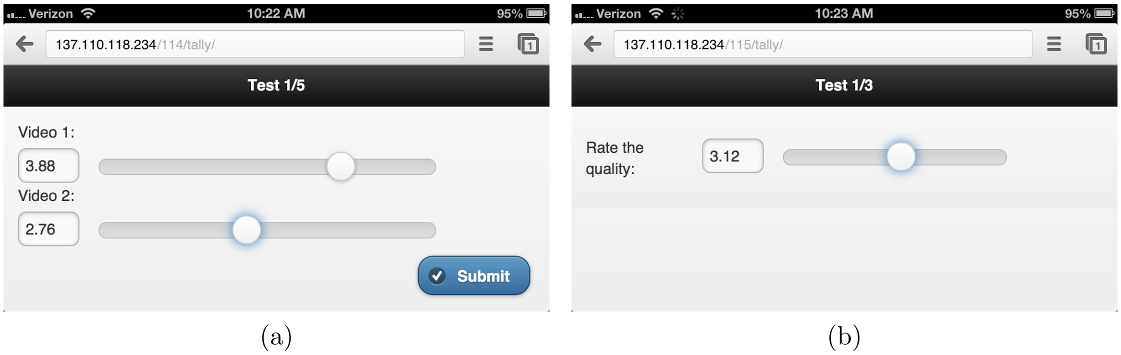


Waiting for other participants. Test will begin shortly...

In terms of overall quality relative to the reference video, degradations in the test video were:

(a) (b) (c)

**Figure 3.5**: Sample voting interface for subjects. (a) List of tests available to the subject. (b) Wait screen before test begins. (c) DSIS voting form.



Video 1: 3.88

Video 2: 2.76

Rate the quality: 3.12

(a) (b)

**Figure 3.6**: Samples of voting forms of the other supported test methods. (a) DSCQS (b) SSCQE.

from [40]. The voting forms are rendered differently according to the test method. Samples of the voting forms for other supported test methods are shown in Fig. 3.6.

## 3.5 Basic Operation

Once the software has been downloaded and installed, users can then run a test as follows:

1. Preparing the videos

   Before the test, the Tester makes the videos that he wants to show to the subjects. This step should be identical to what the tester would do for any other subjective test, except that the reference video (if there is one) and test videos should be labeled within the video as such if necessary, and contained in separate files.

2. Creating the test

   The Tester logs onto the site and creates a new test. After entering some metadata (title, description, etc.), he selects the test method to be used (e.g. DSCQS). Through the interface, he then selects the test and reference videos (if any) that comprise the test, and the system automatically generates test cases based on the test method and randomizes them. For example, if the test method is DSIS then each test case will have the reference video followed by one test video.

3. Running the test

   The Tester loads the videos onto the computer that will display them, and launches the desktop application. Through the application, he logs onto the site again and creates a new test instance of the test he wishes to run. The Tester adds Subjects to the test instance, and they log onto the site as well. The Subjects select the appropriate test instance to which they are enrolled and the Tester clicks ‘Start Test’ on the desktop application. The videos play and the Subjects provide responses until the test is finished, as described in Section 3.2.

4. Accessing the data

   After the test, the Tester can login to the site and download the test data in a variety of formats. The data is available in CSV (comma-separated value) format which can be opened with any text editor or Microsoft Excel, as a Matlab *.mat file, or as a Python *.py file. He can also share the test instance data with other Testers, or share the entire test so that someone else can run their own test instances of that particular test.

## 3.6 Download

Tally, along with full documentation and installation instructions, is available for download at the project website, http://canbal.github.com/Tally/. We maintain the source code in a git repository at github [59], so it is also possible to report bugs, ask for features, and contribute to the project at http://github.com/canbal/Tally. Additionally, we provide a support email address, tally.vpl@gmail.com, for any inquiries about Tally.

## 3.7 Conclusion

Many labs around the world conduct research that relies heavily on perception experiments with video. Standards for operating procedures, such as the ITU recommendations [40], exist for these kinds of experiments. However, the current tools for data collection for subjective testing are outdated or insufficient, and efforts to improve collection methods are too sparse and fractured to be widely effective.

To address this need, we have developed Tally, a web-based data collection tool for subjective video experiments. Our tool has numerous advantages, most notably the decoupled voting and viewing interface, flexibility and robustness, and the ability to share and collaborate on projects. Releasing the software as open source encourages growth and a well-supported user community.

There are several ways Tally can be extended. Providing support for other test methods such as SAMVIQ or two-alternative forced choice could be added. If network traffic is not a concern, the videos could be served over the web as well to allow for remote testing. Also, an online registration system for subjects would be a useful addition.

We believe our web-based design is not only superior to previous attempts at such a tool, but is also the proper way of addressing the data collection problem. It is our hope that Tally is widely adopted; the more people who use it, the better and more effective it will become. We designed Tally on the principles of openness, transparency, and collaboration, and we hope that our tool promotes these values among the research community through its use.

## Acknowledgments

The text of Chapter 3 is, in part, a reprint of a paper that has been published as: A. K. Jain, C. Bal, and T. Q. Nguyen, “Tally: a web-based subjective testing tool,” in *Fifth Int’l. Workshop on Qual. of Multim. Exp. (QoMEX)*, Jul. 2013, pp. 128–129. The dissertation author was the primary investigator and author of this paper.

# Chapter 4

# Restoration of Mixed Resolution Stereo Video

In mixed resolution stereo representation, much of the sharpness loss in one view is perceptually compensated by the other in the fused image. However, there are reasons to recover the high frequency information computationally at the decoder. Viewers with one weak eye which, by chance, corresponds to the full resolution view, would see a blurry image rather than a sharp one. The low resolution view may be used in view synthesis [60, 61], 2D viewing, or scaling across devices, where a higher resolution would be beneficial. If resolution can be recovered at the decoder, then even more aggressive downsampling at the encoder can be afforded, thereby saving more bandwidth. Restoring spectral symmetry to the stereo pair may be desired for certain types of content and viewing duration, where visual fatigue may be a concern [39] (also see Chapter 2). Finally, a full resolution stereo pair is sharper than a mixed one, so a method to convert from mixed to full resolution could be used for high quality viewing.

In this chapter, we introduce a method to super-resolve the low resolution half of a mixed resolution stereo pair. Our work is distinguished by its solution designed specifically for MRSC *video*, whereas previous work has dealt with either obliquely related compression scenarios or only images. We apply a 3D Markov Random Field (MRF) model to stereo super-resolution on compressed data, demonstrating high objective measures, sharp images for subjective comparison, and good temporal consistency in the super-resolved video.

We build upon our previous work [62] with several notable developments. Algorithmic modifications have improved quality while reducing computational complexity and memory demands, allowing the testing of longer and higher resolution videos. We have performed a much more extensive validation of our method, testing different downsampling ratios and parameter values to find the limits of the algorithm and applying a metric to validate the temporal consistency of the super-resolved output. Additionally, we use H.264-compressed stereo data, which increases the applicability and practical value of our method.

The rest of the chapter is organized as follows. In Section 4.1, we discuss our work in the context of related literature on super-resolution for mixed resolution content and MRF modeling. We introduce our method in Section 4.2, and validate its performance through several experiments in Section 4.3. We discuss the significance of the results for MRSC and provide insights for future work in Section 4.4. Finally, we conclude in Section 4.5 with a summary of our work and contributions.

## 4.1 Related Work

The problem of super-resolution for MRSC has been previously considered. In [63], the full resolution image is computed as the minimizing solution to a cost function using estimated disparity information. An autoregressive model is used in [64] to estimate the interpolated pixels of the low resolution view. However, both of these methods only address stereo images and not video, and are therefore susceptible to flicker artifacts when applied frame-by-frame to a sequence. Optical flow and post-processing is used to reduce flicker in [65], which uses mixed resolution to reduce rendering time of high quality 3D video. Our work differs in that we focus on compression, and use a spatio-temporal MRF to enforce temporal consistency directly without requiring motion estimation.

Restoration methods for related compression scenarios have been investigated as well, such as mixed resolution for monoscopic video [66] and mixed spatio-temporal resolution [67]. In the mixed resolution multiview plus depth scheme, several views of a scene are transmitted, alternating between full and low resolution, along with corresponding depth information for each view. In [28], high frequency information from neighboring views is projected onto the low resolution view. A similar approach is taken in [68], with an additional refinement search to handle possible geometric distortions in the provided depth maps. These methods do not consider the temporal dimension in their formulations ( [28] is performed exclusively on images), and are also targeted at different coding applications than MRSC.

Spatio-temporal MRF models have been applied in video disparity estimation. In [69], occlusion and motion information are incorporated into a 3D MRF which is optimized using loopy belief propagation. A similar paradigm is used in [70], where time-of-flight range data is fused with stereo imagery to obtain temporally consistent depth information. Other methods of video disparity estimation use steerable filters across spatio-temporal patches [71], aggregating temporal statistics [72], and frame-by-frame estimation followed by total variation minimization spatio-temporal filtering [73]. While many of these methods have good results, none of them are specifically designed for super-resolution.

A more suitable framework for our problem is found in [74, 75], where an image is super-resolved by matching patches from a database of high resolution images. The optimal matching patch is found by minimizing the energy in an MRF where spatial consistency constraints are imposed on the low and high frequency candidate patches. The work in [76] extends this method to video by augmenting the training set with the previous super-resolved frame. Instead, we model the problem as a spatio-temporal MRF which optimizes patch selection along all dimensions. This technique is also more suitable for stereo, where temporal information can help disambiguate multiple matches in the spatial domain [77] (also see Chapter 5). Thus, we extend the work in [75] to the case of stereo video, as described in the next section.

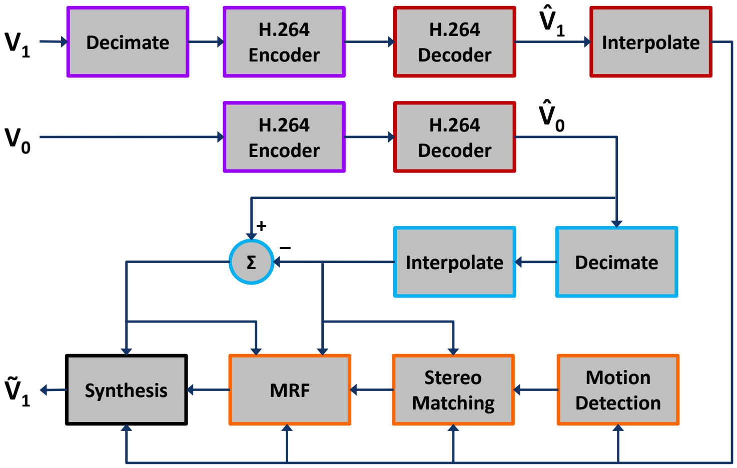


**Figure 4.1**: Overview of the MRSC compression pipeline and proposed approach for super-resolution.

## 4.2 Proposed Method

Based on the review in Section 4.1, there is cause for development of a technique that leverages video data and provides analysis specific to the MRSC problem. This arena is ideal, and perhaps uniquely suited, for extending example-based super-resolution approaches as in [75] to video. A frame-by-frame reconstruction of a video can result in flicker artifacts. Conversely, a database of spatio-temporal training examples would have to be enormous in order to represent the variety of possible object motion. In MRSC, the "database" is the full resolution view and has high correlation with the low resolution "test" video. Thus, a patch from the test set either has a near-exact match or many similar matches available from which to choose. This patch-matching amounts to disparity estimation; however, the final goal is not a dense, pixel-accurate, disparity map, but rather a visually plausible reconstruction of high frequency information.

Fig. 4.1 gives an overview of the compression scenario and proposed super-resolution approach. Of the two source views $V_0$ and $V_1$, the latter is downsampled and then both are independently coded with H.264, transmitted, and decoded. We refer to the full and low resolution views after decoding as the reference and test, and denote them $\hat{V}_0$ and $\hat{V}_1$, respectively. After decoding, the test view $\hat{V}_1$ is interpolated with bicubic interpolation to full resolution. These are the outputs normally viewed as MRSC content. Instead, they are passed to the subsequent stages that comprise the super-resolution algorithm, described as follows.

First, the reference view $\hat{V}_0$ is separated into low-pass and high-pass bands using the same processing as for the test view $\hat{V}_1$. The full resolution view is anti-aliased, downsampled, then upsampled to form the low pass band, $\hat{V}_0^L$, and this is subtracted from $\hat{V}_0$ to form the high pass band, $\hat{V}_0^H$. We let $\hat{V}_1^L$ denote the bicubic interpolated low resolution view of the mixed resolution stereo pair.

Next, stereo matching is performed between the two low frequency bands. We divide $\hat{V}_1^L$ into overlapping spatio-temporal color (RGB) patches. Using spatio-temporal patches allows for more accurate matches to be found since motion can disambiguate spurious matches [77] (also see Chapter 5). For each patch in $\hat{V}_1^L$, we form a list of candidate patches in $\hat{V}_0^L$ that best match the test patch in terms of least $L_2$ norm error. The patches are constrained to lie along the same scan line as the target patch for rectified stereo, as in the experiments in Section 4.3, but can also be chosen from within a small band of the patch scan line for converged stereo. Matching patches are only sought from corresponding frames between the two views.

Stereo matching is the most costly operation of the super-resolution procedure. In order to make this step more efficient, matching is performed only where there is motion. Otherwise, the candidate indices from the previous patch are propagated forward, and the match error is recomputed for the candidates. Motion detection is performed on the low-resolution view by thresholding the absolute frame difference. This is a simple, efficient technique that reduces the number of patches that have to be matched by 87% on average (see Section 4.3.4).

Once we have a candidate list for each patch, we select the best candidate through global optimization based on constraints for consistency and match error. To do this, we treat the spatio-temporal patches as nodes in a 3D Markov network depicted in Fig. 4.2. The high resolution patches, $x$, are considered hidden nodes and are what we seek to estimate. The low resolution patches, $y$, are observation nodes indicated by the orange circles in the figure. Each node is attached to 4 spatial neighbors, 2 temporal neighbors, and an observation node, where lines

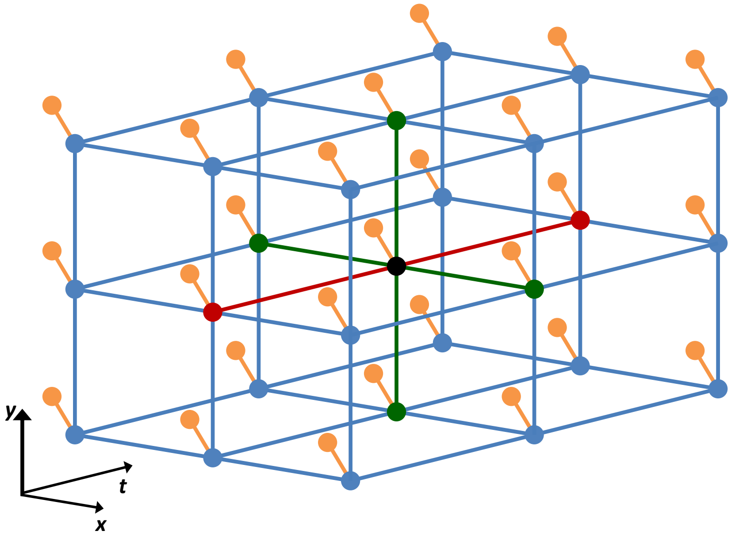


**Figure 4.2**: Spatio-temporal Markov network. The node under consideration (black) is connected to 4 spatial neighbors (green) and 2 temporal neighbors (red). Nodes representing the observed low resolution patches are marked in orange; all others are hidden nodes representing high resolution patches.

in the figure indicate statistical dependency. Since optimization of the Markov network will ensure consistency between all patches, the addition of 2 temporal neighbors is the key difference between our method and [75], and is crucial to mitigating flicker.

With the Markov network in place, the math is very similar to the 2D Markov case, and we follow [75]. Given a low resolution observation $y$, the probability of a candidate high resolution patch at that node is

$$P(x|y) \sim \prod_{(i,j)} \psi_{ij}(x_i, x_j) \prod_{i} \phi_i(x_i, y_i), \tag{4.1}$$

where $(i, j)$ refers to neighboring node pairs, and $\psi$ and $\phi$ are compatibility functions between pairs of high resolution patch choices (from $\hat{V}_0^H$) and pairs of low resolution patches from the test and reference views (from $\hat{V}_1^L$ and $\hat{V}_0^L$), respectively. Therefore, the first product in (4.1) reflects a smoothness cost for candidate high resolution patches, and measures how well the high resolution patches agree with their neighbors. The second product is a data cost associated with matching

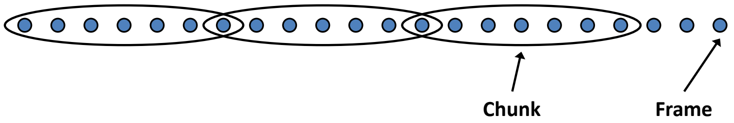


**Figure 4.3**: Chunk processing of video. Several frames compose each chunk, which are overlapped by one frame. Overlapped frames are averaged in the final output.

between the two low resolution views. Modeling the compatibility between low resolution patches as well as between possible high frequency patches distinguishes this approach from general-purpose disparity estimation schemes. These functions ensure spectral consistency across spatial frequency bands—an important consideration in super-resolution.

The compatibility functions are learned from the data by means of sampling patches such that they overlap in each dimension. To enforce spatial smoothness between patch choices, the compatibility function is chosen such that it is small when the overlapped pixels are very different between patches and near one when they are similar:

$$\psi_{ij}(x_i, x_j) = \exp\left(-d_{ij}(x_i, x_j)\right). \tag{4.2}$$

Here, $d_{ij}(x_i, x_j)$ is the sum of squared differences in the overlap region of the two input patches. The function $\phi$ has a similar form.

The high resolution patches are determined among the candidates as the set that maximizes the network probability. A fast, approximate maximum a posteriori (MAP) solution is found using belief propagation. Messages, or terms in the probability computation, are passed between connected nodes and their state probability is updated at each iteration until a fixed stopping point. The state maximizing the network probability at the end of the iteration loop is then chosen as the MAP estimate. Again, the key difference in our approach is the addition of temporal neighbors, which pass messages to the central node to update its probability.

Once the appropriate high frequency patches have been selected from $\hat{V}_0^H$, they are added to the corresponding observed low resolution patches in $\hat{V}_1^L$. Where they overlap, the patches are averaged to yield the final super-resolved video.

Running the above procedure on a video as a whole is limiting and ineffective. First, holding full input videos and all supporting data, candidate lists, and outputs in memory prohibits the processing of high resolution or lengthy videos. Second, patches from very separate temporal segments of the video should not affect one another, and their messages will not be propagated that far with a reasonable number of iterations of belief propagation. Finally, having to receive and decode the entire video before processing would introduce too much delay for streaming applications. Thus, we process the video in chunks, as shown in Fig. 4.3. Several frames are processed simultaneously in a chunk, and the MRF ensures consistency of all frames within it. State information with temporal dependence, such as motion detection and stereo matching, is passed from the previous chunk to the one being processed. Chunks are overlapped by one frame, and overlapped frames are averaged to promote consistency between chunks. Note that the full video is encoded and decoded in its entirety, and only the super-resolution processing is partitioned into chunks.

## 4.3 Results

We tested our algorithm on five datasets: *Balloons* from [78], *Book_Arrival* and *Outdoor* from [79], and *Poznan_Hall2* and *Poznan_Street* from [80, 81]. The latter two source videos were decimated by a factor of 2 in each dimension to $960 \times 544$ to reduce computation time, and the first 100 frames of all five sequences were used in testing. The videos have a variety of objects, lighting, depth, color, and motion. All of the datasets are multiview sequences from which two views were used as a stereo pair. The left-right view indices and other parameters are summarized in Table 4.1.

We ran a set of four experiments on these five datasets to validate the performance of the algorithm under various conditions. First, we test different combinations of patch grid sampling parameters for optimal quality and run time. The next experiment characterizes the rendering quality by the downsampling ratio of the low resolution view with and without compression. In the third experiment, we measure algorithmic performance as a function of the bit rate of the input videos by varying the compression quality. Fourth, we demonstrate the advantages of spatio-temporal processing by comparing the temporal consistency of the super-resolved output to frame-by-frame processing.

Certain parameters and test conditions remained fixed throughout our experiments. The left view was always kept at full resolution and the right view was chosen as the low resolution view. Videos were encoded with H.264 using the x264 software [82] with a fixed quantization parameter (QP) that is the same for each view. To estimate the states of the MRF, we used 10 candidate patches for each node and 5 iterations of belief propagation. For the motion detection block, a pixel was flagged as being in motion when the absolute difference between it and its value on the previous frame exceeded 5% of the dynamic range. A patch was determined to be in motion when over 30% of its pixels were in motion. Since we are not considering the effects of noise in this work, the thresholds are fixed and purposefully set somewhat low. We choose to err on the side of false positives, which means having to perform matching for more patches, in order to ensure high rendering quality. However, we are consistently able to filter out static backgrounds and our computational savings are significant, as shown in Section 4.3.4.

The default values for the other parameters are as follows. We used a downsampling ratio of 2 in each spatial dimension and a QP of 30 for each view. The default patch size was $7 \times 7 \times 3$ sampled every 4 pixels in $x$ and $y$ and computed at each frame in time. The videos were processed in chunks of 7 frames each, overlapping by one frame. Justifications for some of these choices are given in the experiment results.

Prior to encoding, the right view was prefiltered with a square box filter and downsampled. The size of the filter was chosen to spatially bandlimit the video to guarantee no aliasing artifacts. That is, for a downsampling ratio of $D$, the spatial bandlimit should be $1/D$. Taking the first zero in the spectrum of the prefilter as its bandwidth, the aliasing requirement dictates filter sizes of $4 \times 4$, $8 \times 8$, and $16 \times 16$ for the downsampling ratios of 2, 4, and 8, respectively, used in our experiments. The frequency responses of these filters are shown in Fig. 4.4.

**Table 4.1**: Characteristics of datasets used in experiments.

| Dataset | Resolution | Frame | Camera Index | |
|---|---|---|---|---|
| | | Rate (Hz) | Left | Right |
| *Balloons* | 1024 × 768 | 30 | 1 | 3 |
| *Book_Arrival* | 1024 × 768 | 16.7 | 7 | 6 |
| *Outdoor* | 1024 × 768 | 16.7 | 5 | 4 |
| *Poznan_Hall2* | 960 × 544 | 25 | 10 | 8 |
| *Poznan_Street* | 960 × 544 | 25 | 5 | 4 |

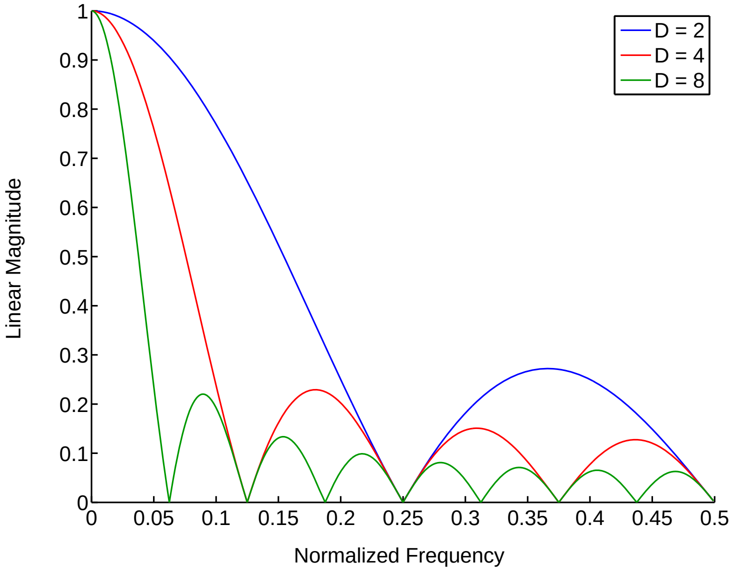


**Figure 4.4**: Positive frequency response of anti-aliasing filters used in experiments for different downsampling ratios.

While less aggressive filtering has been previously applied [28,63] or none at all [64], we emphasize that considering the application of compression and visual quality, aliasing must be avoided. The jagged, high-frequency artifacts that are typical of aliasing will be visible in the fused 3D percept. Aliasing could also potentially remove fine details from one of the two views, resulting in binocular rivalry when viewed stereoscopically. Since the encoder cannot assume that the videos will be super-resolved at the decoder, these artifacts may remain in the decoded MRSC videos. Therefore, though even in our own informal experiments we have achieved higher PSNR results will gentler filtering, we choose prefilters that appropriately bandlimit the data.

The following results use the peak signal-to-noise ratio (PSNR) between the super-resolved output and the uncompressed source video as a quality measure, reported as the average over all frames in the sequence. We also computed the structural similarity index (SSIM) [83] for all results, which showed similar trends to those of PSNR. To avoid redundancy, we have omitted these results but comment on them where relevant.

### 4.3.1 Patch Sampling

After the compression stage, the parameters that most affect the quality and computational cost of the super-resolution algorithm are those that define the grid of patches: the patch size and patch interval. Fig. 4.5 illustrates a spatial patch grid and indicates these two parameters.

To determine an appropriate combination of patch size and interval, we vary these parameters and compute the output quality, keeping the other parameters fixed at their default values. We test for spatial parameters only, and fix the temporal patch size and interval to 3 frames and 1 frame, respectively. The patch size must be small enough to capture fine details, but large enough to span enough structure to enable informative matching. The patch interval must be less than the patch size since we require patches to be overlapped in order to compute compatibility directly from the data. Thus, we test patch sizes of 5, 7, and 9 pixels in $x$ and $y$, and patch intervals of 4, 6, and 8 pixels. Table 4.2 gives the results in PSNR for this experiment. The maximum PSNR for each dataset is bolded, and the blank entries represent invalid size and interval combinations that do not satisfy the overlap requirement.

The variation in PSNR within each dataset is small, below 0.3 dB in all cases. However, there are clear and consistent trends among the five sequences. Larger patches with smaller intervals give better results, with the combination of a $9 \times 9$ patch size sampled every 4 pixels producing the highest quality. The least optimal patch grid parameters are 9 and 8 for the patch size and interval, respectively.

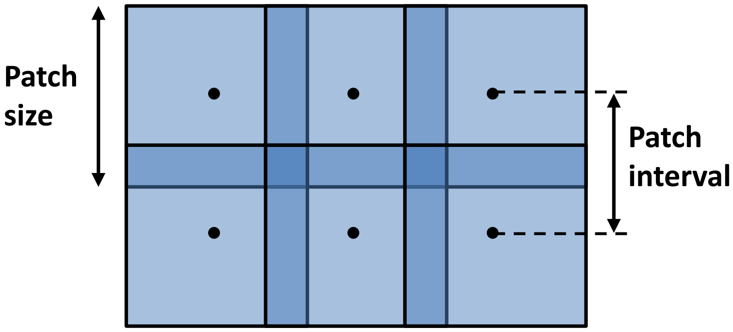


**Figure 4.5**: Patch sampling grid (only spatial dimensions shown). Patch size and interval are equal in $x$ and $y$.

**Table 4.2**: PSNR (dB) for different spatial patch grid parameters.

| Dataset | Patch Size | Patch Interval 4 | 6 | 8 |
|---|---|---|---|---|
| *Balloons* | 5 | 33.78 | | |
| | 7 | 33.96 | 33.74 | |
| | 9 | **33.97** | 33.87 | 33.69 |
| *Book_Arrival* | 5 | 33.66 | | |
| | 7 | 33.82 | 33.63 | |
| | 9 | **33.83** | 33.74 | 33.57 |
| *Outdoor* | 5 | 33.60 | | |
| | 7 | 33.75 | 33.57 | |
| | 9 | **33.76** | 33.68 | 33.51 |
| *Poznan_Hall2* | 5 | 33.51 | | |
| | 7 | 33.66 | 33.48 | |
| | 9 | **33.67** | 33.59 | 33.43 |
| *Poznan_Street* | 5 | 33.45 | | |
| | 7 | 33.61 | 33.42 | |
| | 9 | **33.62** | 33.53 | 33.37 |

A patch size of 7 and interval of 4 yields almost identical performance to the optimal patch parameter combination, only 0.01 dB below optimum across the datasets. The computational cost and memory requirement is directly related to the square of the patch size and inversely related to the square of the interval.

For example, a patch size of 9 and interval of 3 compared to a grid with a patch size of 7 and interval or 6 requires roughly $\left(\frac{9}{7}\right)^2\left(\frac{6}{3}\right)^2 = 6.6$ times as much computation/memory. Therefore, we choose a patch size of 7 and interval of 4 as the default parameter set since it provides virtually optimal quality for less computational cost.

### 4.3.2 Downsampling Ratios

As a baseline measure of rendering quality, we process the full resolution reference view and the low resolution test view directly without compression. The test view is decimated by a factor of $D = 2$, $D = 4$, or $D = 8$ in each spatial dimension. We also repeat the experiment for compressed data using the default QP of 30 in order to understand the performance degradation due to coding.

Figs. 4.6–4.7 show a zoomed-in portion of the results for the *Balloons* and *Poznan_Street* datasets. Visual quality is maintained even with severe downsampling. Super-resolution from coded data introduces more distortion than from source data, but still performs well.

Table 4.3 gives the PSNR of the output and its improvement over applying bicubic interpolation to the test view. The algorithm performs well, offering as much as a 6.1 dB gain, and a 3.7 dB gain on average across downsampling ratios and datasets, over bicubic interpolation for uncompressed data. For coded data, the maximum gain is about 3.7 dB with an overall average of 2.3 dB.

These data are plotted in Fig. 4.8 as a function of the total bit rate of the encoded reference and test views. For comparison, the PSNR of the reference view is also shown. Here the relative performance of the algorithm with compressed and uncompressed data is apparent. Super-resolution from source data produces a nearly uniform gain over downsampling ratios versus the case when the inputs are compressed. Across all datasets and ratios, coded data results in a 2.6 dB drop in performance compared to uncompressed videos.

For both types of input, compressed or not, there is an average loss of 3 dB with each octave step down. Since a 3 dB loss represents a doubling of error, and each octave step represents a quarter of the information present (subsampling is

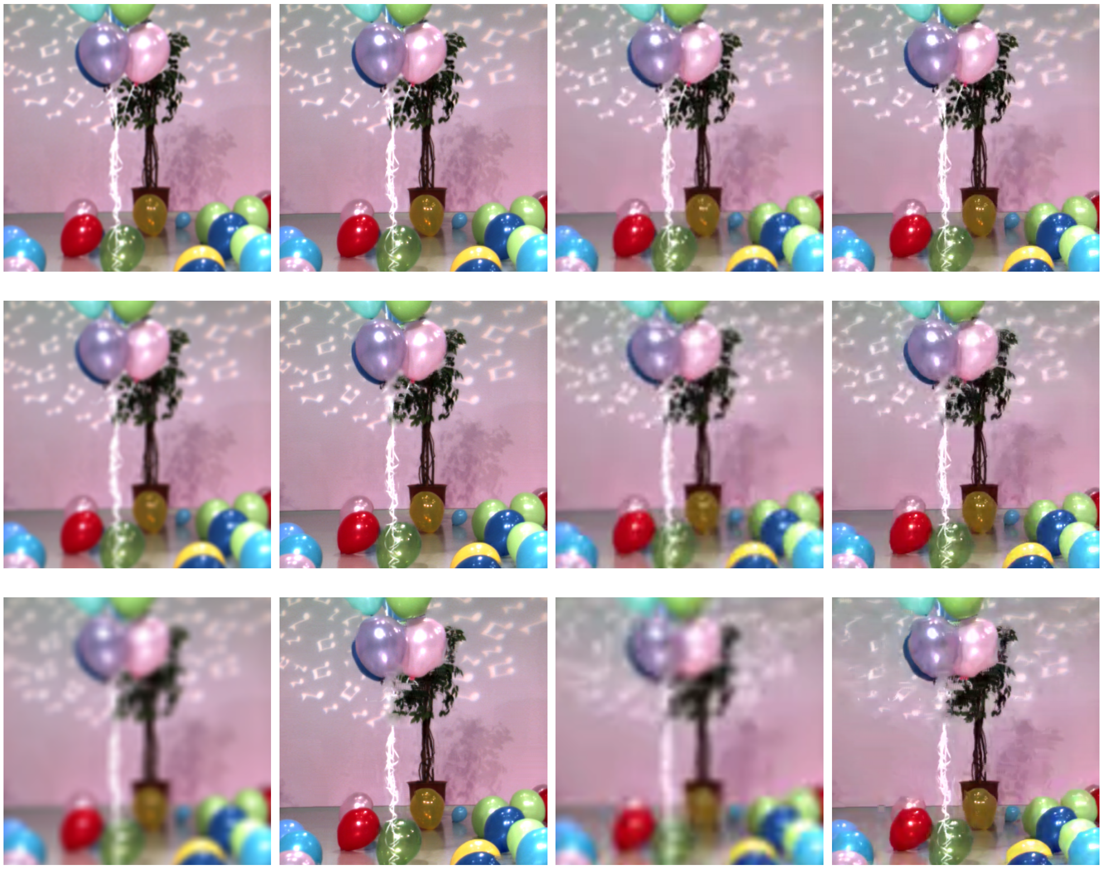


**Figure 4.6**: Sample image results from *Balloons* sequence. Images are 480 × 480 cropped from frame 5 of sequence, top left pixel coordinate (row, column) = (220,200). Rows (top to bottom): downsampling factor of 2, 4, and 8. Columns (left to right): bicubic interpolation of uncompressed test view, super-resolution output of uncompressed test view, bicubic interpolation of coded test view, super-resolution output of coded test view.

done in two spatial dimensions), the algorithm copes well with the loss of data. However, the performance drop from $D = 2$ to $D = 4$ is less severe than from $D = 4$ to $D = 8$. Further, there is an average of 14% drop in bit rate from $D = 2$ to $D = 4$, but only an additional 6% drop from $D = 4$ to $D = 8$. Thus, decreasing the resolution of the test view beyond a factor of 4, both in terms of bit rate savings and potential super-resolution quality, is a losing trade-off.

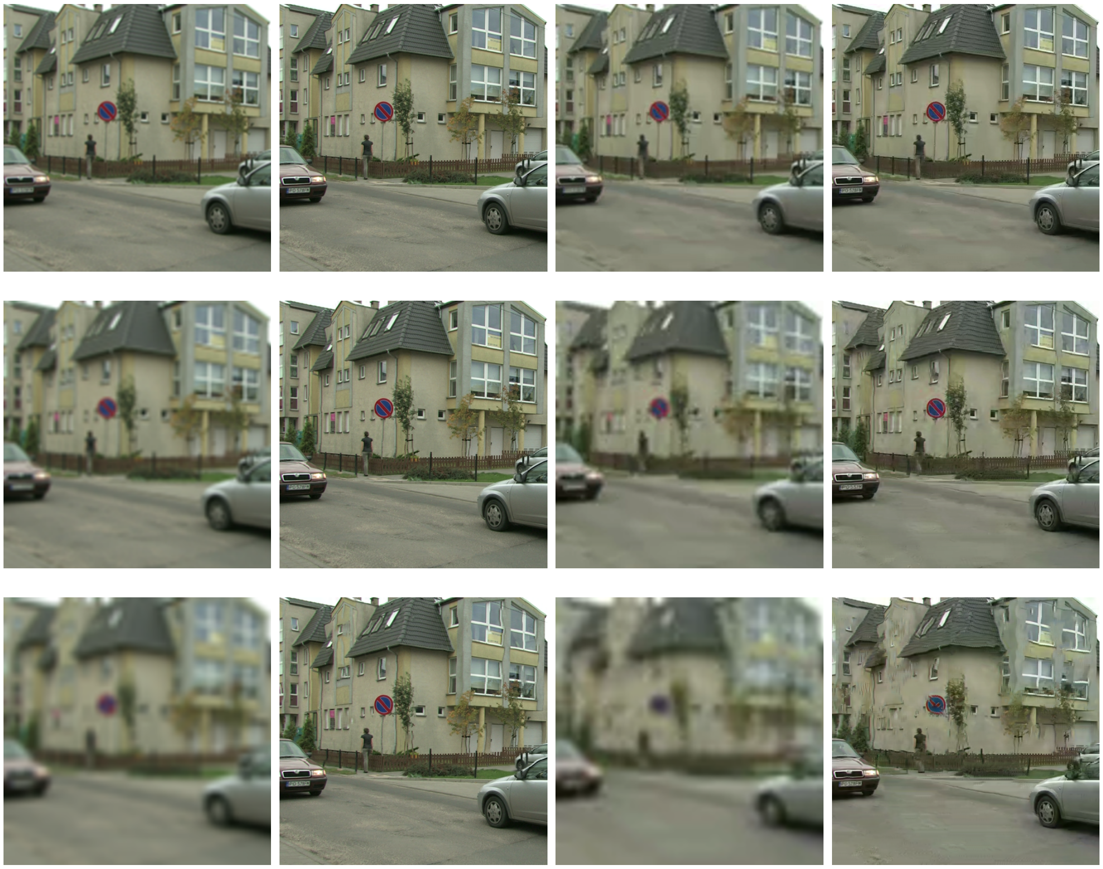


**Figure 4.7**: Sample image results from *Poznan_Street* sequence. Images are 480 × 480 cropped from frame 5 of sequence, top left pixel coordinate (row, column) = (1,100). Rows (top to bottom): downsampling factor of 2, 4, and 8. Columns (left to right): bicubic interpolation of uncompressed test view, super-resolution output of uncompressed test view, bicubic interpolation of coded test view, super-resolution output of coded test view.

### 4.3.3 Compression Quality

In this experiment, we compress the mixed resolution stereo pair at different bit rates, super-resolve the low resolution view at the decoder, and measure the PSNR of the output versus the bit rate of the input. The two views were coded independently with identical parameters, other than resolution. We varied the bit rate by changing the QP from 20 to 50 in steps of 5 while keeping the downsampling ratio of the low resolution view at the default factor of 2. These results for the five datasets are plotted in Fig. 4.9. The data plotted in this figure is the same

**Table 4.3**: PSNR (dB) of super-resolved output, with and without compression, and improvement over bicubic interpolation.

| Dataset | $D$ | Uncompressed | | Compressed | |
|---|---|---|---|---|---|
| | | Proposed | ΔBicubic | Proposed | ΔBicubic |
| *Balloons* | 2 | 36.97 | +3.06 | 33.15 | +1.81 |
| | 4 | 33.18 | +3.45 | 30.05 | +2.33 |
| | 8 | 28.89 | +3.35 | 26.30 | +2.16 |
| *Book_Arrival* | 2 | 31.11 | +0.56 | 30.61 | +1.35 |
| | 4 | 29.82 | +1.85 | 28.77 | +1.74 |
| | 8 | 27.49 | +1.80 | 25.91 | +0.95 |
| *Outdoor* | 2 | 32.69 | +4.83 | 30.45 | +3.50 |
| | 4 | 30.59 | +5.87 | 27.80 | +3.69 |
| | 8 | 27.76 | +5.57 | 24.38 | +2.63 |
| *Poznan_Hall2* | 2 | 37.14 | +3.21 | 34.23 | +2.04 |
| | 4 | 34.07 | +3.11 | 31.42 | +1.82 |
| | 8 | 30.93 | +2.62 | 28.61 | +1.44 |
| *Poznan_Street* | 2 | 33.82 | +5.85 | 30.60 | +3.72 |
| | 4 | 31.11 | +6.11 | 27.62 | +3.34 |
| | 8 | 27.08 | +4.38 | 23.78 | +1.59 |

as in Fig. 4.8, representing the PSNR of the reference view, the super-resolved and bicubic-interpolated outputs of the coded test view at different QPs, and the super-resolved output from uncompressed data.

Varying the QP results in the same trend across datasets. Algorithmic performance is bounded by the envelope defined by the reference view quality and the rendering quality from uncompressed data. At low bit rates, the performance of the proposed method and bicubic interpolation are almost equivalent. Coding artifacts dominate the video quality and there is not a lot of potential for improvement. The full resolution reference view PSNR also converges at these lower bit rates, showing that the rendering is limited by input quality. This trend is illuminated in Fig. 4.10, where the sequence is coded with QP = 50. The coding artifacts are very apparent in the the reference and test views, and the interpolated

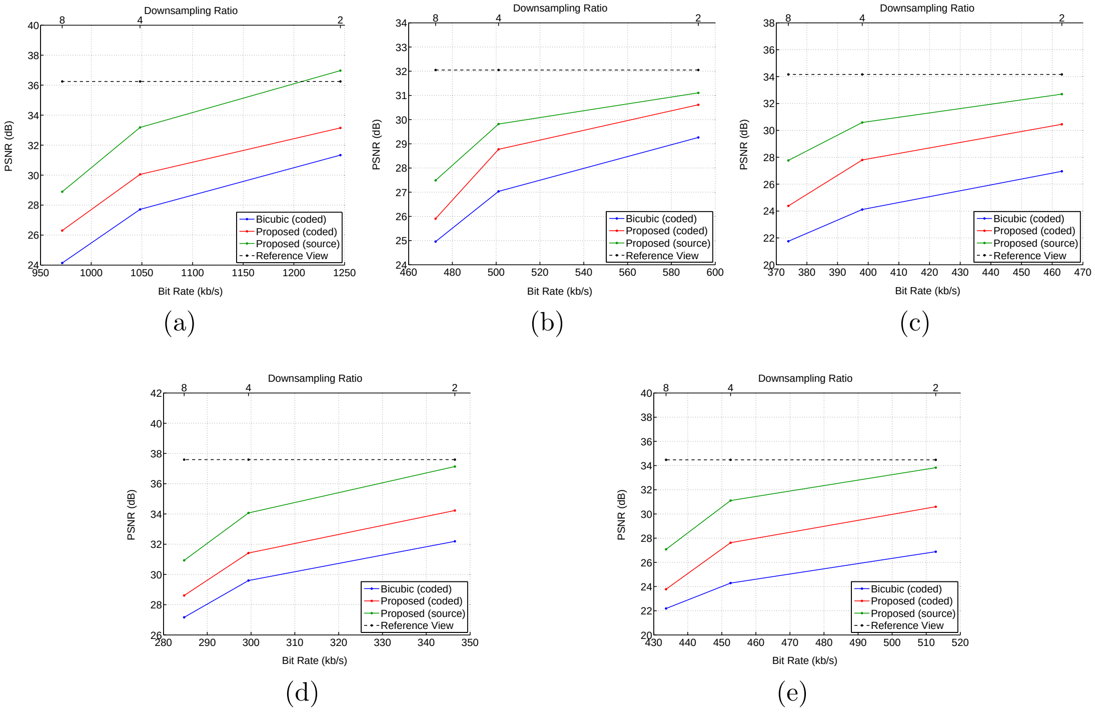
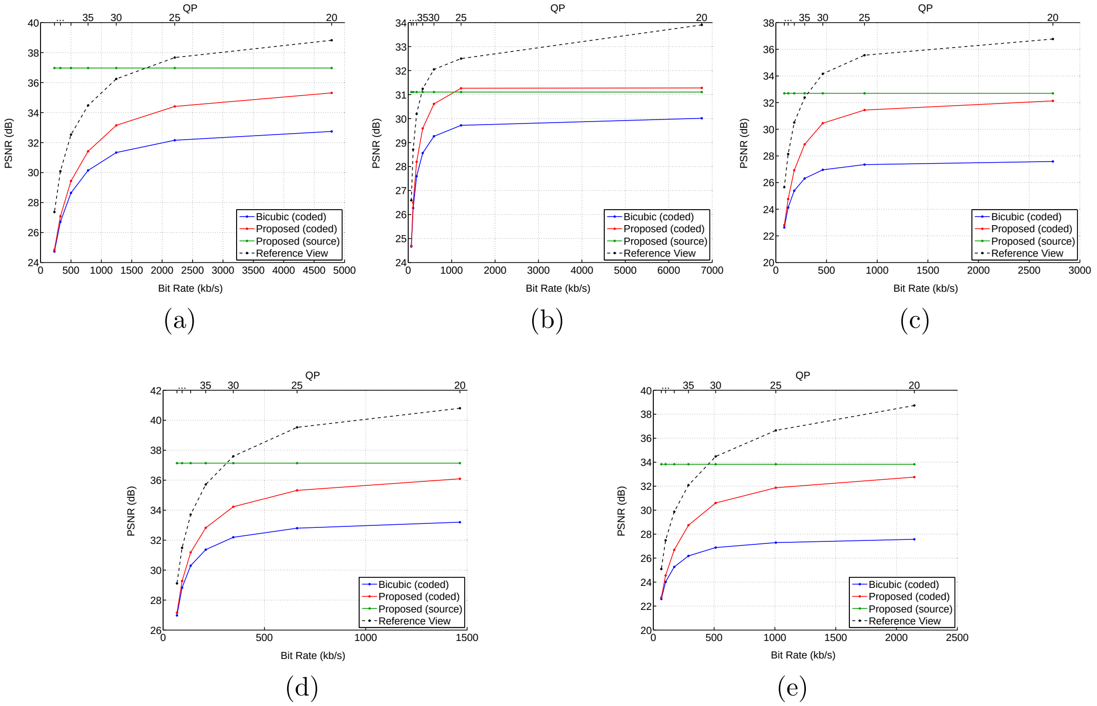


**Figure 4.8**: PSNR vs. bit rate for all datasets as the downsampling factor is varied (top axes show corresponding downsampling factor). Curves show PSNR of full resolution view (same for all downsampling factors), bicubic interpolation of coded test view, and super-resolved output for coded and uncompressed test view. (a) *Balloons* (b) *Book_Arrival* (c) *Outdoor* (d) *Poznan_Hall2* (e) *Poznan_Street*.

and super-resolved outputs are virtually indistinguishable.

At higher bit rates, the curves in Fig. 4.9 diverge and tend toward constant values. In this region, rendering quality is limited by differences between the two cameras, matching accuracy, and information lost during the downsampling process. There is a small anomaly with the *Book_Arrival* dataset as can be seen in Fig. 4.9b, for which the PSNR of the output from uncompressed data is slightly lower than the output PSNR from compressed data for a QP of 20 and 25. This result is likely due to noise in the source sequence, which becomes smoothed out with even light compression. Smoothing improves match accuracy and hence the rendering quality. The difference is minimal, and the anomaly does not exist under the SSIM index.

Improvement over bicubic interpolation is as great as 5.2 dB, and is 2.5–



(a)

(b)

(c)

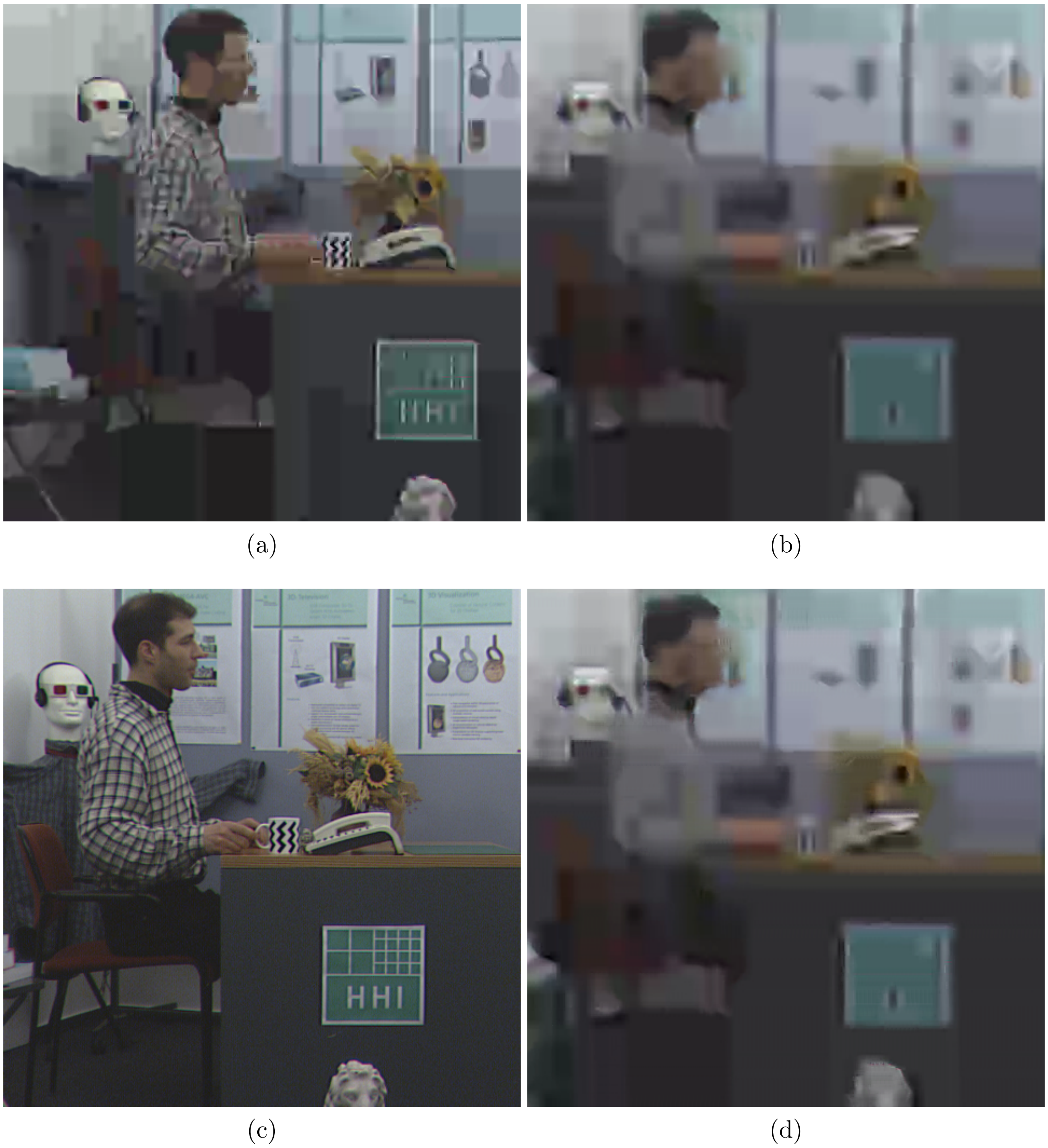
(d)

(e)

**Figure 4.9**: PSNR vs. bit rate for all datasets as the QP is varied (top axes show corresponding QP values). Curves show PSNR of full resolution view, bicubic interpolation of coded test view, and super-resolved output for coded and uncompressed test view. (a) *Balloons* (b) *Book_Arrival* (c) *Outdoor* (d) *Poznan_Hall2* (e) *Poznan_Street*.

3.3 dB for the lowest three QP values on average across the datasets. After QP = 30, the average drops to 1.8–0.1 dB. This is also approximately the point of intersection of the reference view quality and the quality of super-resolution from uncompressed data for all the datasets. This point signifies that, for a given down-sampling ratio, any increase in bit rate will not greatly aid the super-resolution process. Therefore, a QP of 30 serves as a good compromise between bit rate and rendering quality, which is why we set it as the default value for the other experiments.

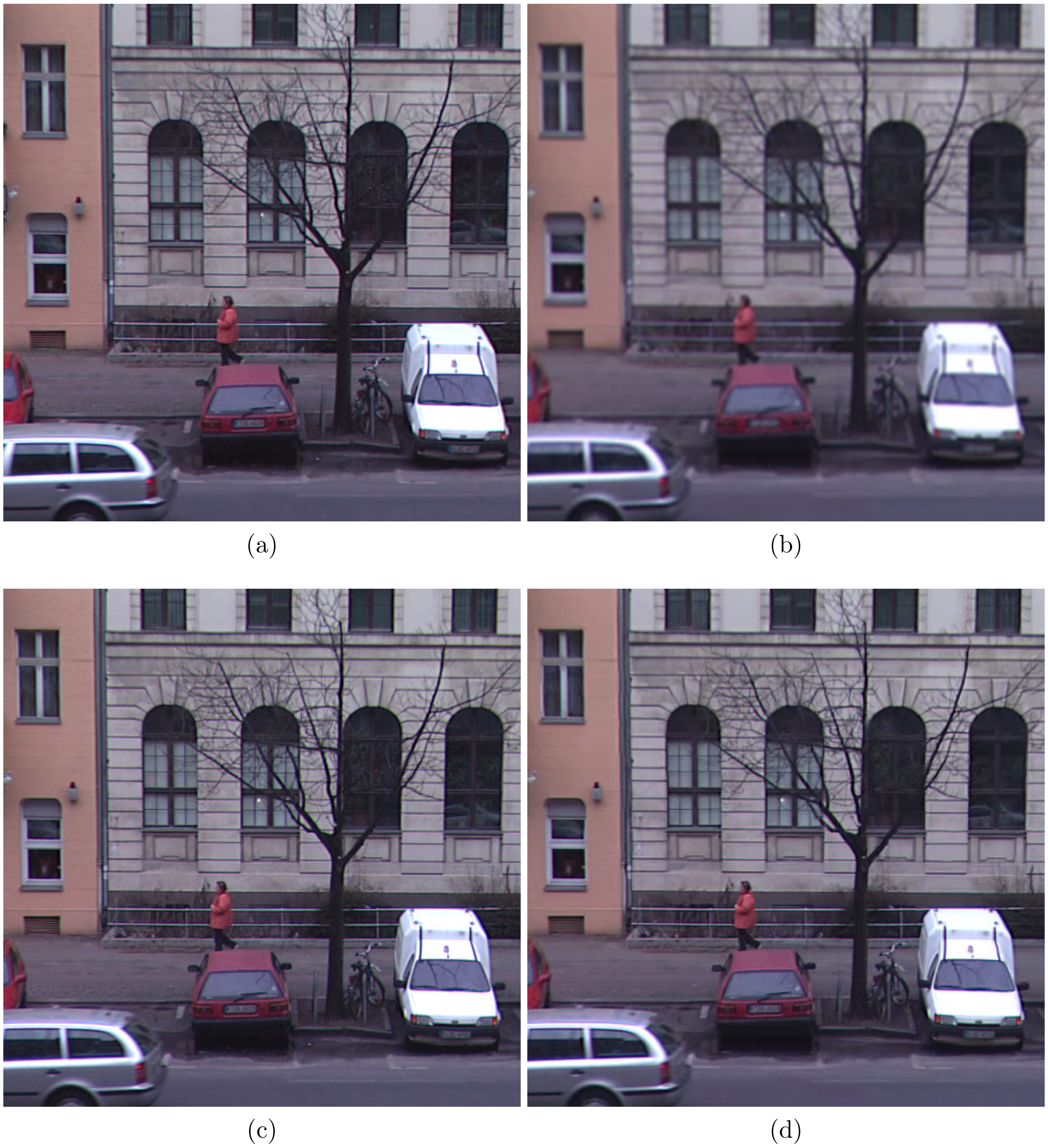
Though the performance over bicubic interpolation is significant, the PSNR of the super-resolved output is still lower than that of the reference video by as much as 6 dB. The difference in PSNR is minus 2.3–4.3 dB across the range of QPs averaged over the datasets. However, this gap in quality between the two views



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

**Figure 4.10**: Sample image results from *Book_Arrival* sequence for QP = 50. Images are 480 × 480 cropped from frame 5 of sequence, top left pixel coordinate (row, column) = (125,150). (a) Reference (left) view (b) interpolated test (right) view (c) ground truth test view (d) super-resolution output of test view.

may not be as wide as the PSNR suggests. Using SSIM, the quality of the rendered output is closer to the reference view quality for most QP values. For the three highest bit rates, the output quality curve moves 6% further away for the *Balloons* dataset, but moves 24% closer to the reference quality curves for the other four



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

**Figure 4.11**: Sample image results from *Outdoor* sequence for QP = 20. Images are 480 × 480 cropped from frame 5 of sequence, top left pixel coordinate (row, column) = (110,200). (a) Reference (left) view (b) interpolated test (right) view (c) ground truth test view (d) super-resolution output of test view.

datasets. The subjective quality is also of primary importance. In Fig. 4.11, the sequence is coded with QP = 20. The super-resolution result is 4.6 dB lower than the reference view PSNR, but visually the two views have similar quality.

### 4.3.4 Benefits of Spatio-Temporal Processing

In order to demonstrate the benefit of temporal processing, we generate an output using the default parameters but turn off all temporal dependence except for motion detection. In this experiment, except for stereo match propagation, each frame is processed independently: patches are matched using only spatial information, the MRF optimizes over one frame at a time, and there is no temporal averaging of patches since they do not overlap in time.

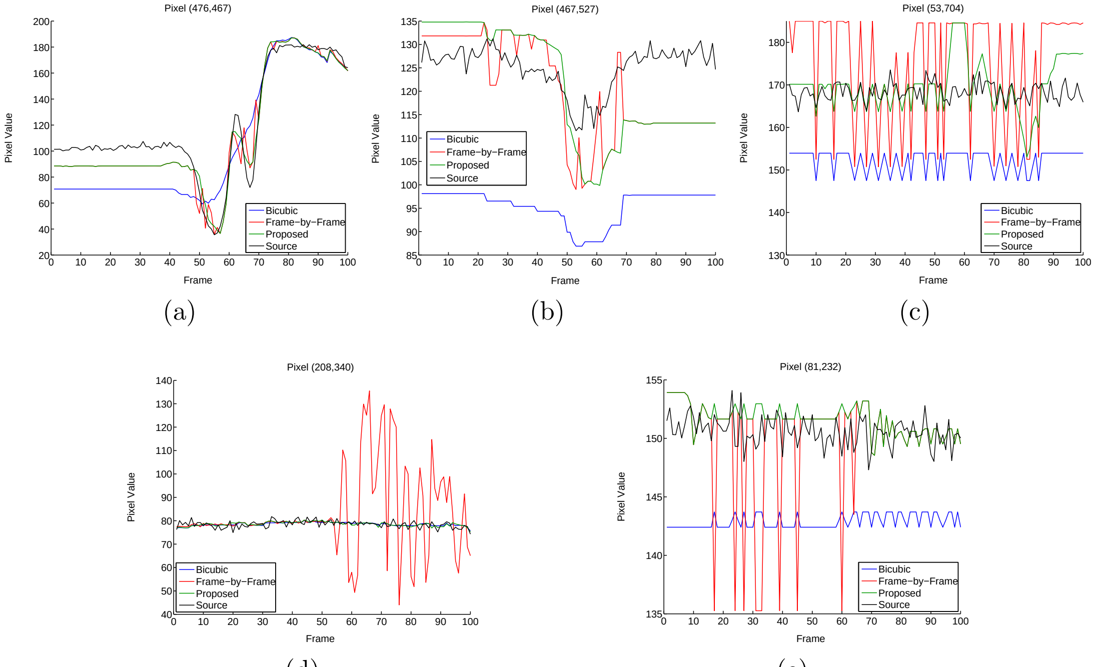
The resulting video has noticeably more flicker. While the proposed method can introduce temporal artifacts due to errors in patch matching, these artifacts are smoothed by averaging patches in the overlapped regions. Further, many temporal artifacts are avoided altogether by the increased accuracy of spatio-temporal patch matching and optimization with the spatio-temporal MRF. In the frame-by-frame processed video, flicker artifacts are introduced in primarily static regions of the video where matching is more susceptible to noise.

To provide some measure of the flicker, we use a variation of the metric proposed in [76], another work addressing example-based video super-resolution. First, we form the difference between the source video and the super-resolution output and convert the result to grayscale. Next, we high-pass filter this signal along the temporal dimension. The high-pass filter is designed with a cutoff of 1/4 of the frame rate, and is realized by first low-pass filtering the signal with a box filter of length 8, and subtracting this from the signal itself. The high-pass result is squared to compute the energy in the passband, which we call the flicker energy. This procedure is conducted for the frame-by-frame and proposed approaches.

Table 4.4 reports several statistics computed from the flicker energy for the frame-by-frame (FbF) and proposed approaches. The mean flicker energy, averaged across pixels and frames, and the temporal variance of the mean flicker energy averaged across pixels, are given as ratios. Summing the flicker energy across frames yields a spatial representation of the flicker. The percentage of pixels where the flicker is greater for the frame-by-frame approach than for the proposed approach is also listed in the table. In all cases, the proposed approach greatly outperforms frame-by-frame processing. Using temporal information yields an average of

**Table 4.4**: Flicker energy statistics for frame-by-frame versus proposed processing.

| Dataset | Mean energy<br>FbF : Prop. | Temporal variance<br>FbF : Prop. | % pixels<br>FbF > Prop. |
|---|---|---|---|
| *Balloons* | 12.80 | 138.00 | 99.78 |
| *Book_Arrival* | 5.67 | 10.73 | 98.78 |
| *Outdoor* | 6.52 | 35.27 | 99.35 |
| *Poznan_Hall2* | 5.30 | 2.67 | 98.71 |
| *Poznan_Street* | 4.66 | 28.59 | 98.20 |



(a) (b) (c)

(d) (e)

**Figure 4.12**: Temporal pixel response for frame-by-frame processing, bicubic interpolation, proposed method, and uncompressed source video. Pixel coordinates are given as (row, column) indices from top left corner of the image. (a) *Balloons* (b) *Book_Arrival* (c) *Outdoor* (d) *Poznan_Hall2* (e) *Poznan_Street*.

7 times less flicker energy. Additionally, an average of 99% of the pixels have less flicker energy across the sequence than for frame-by-frame processing. We note that while the *Balloons* dataset yields high objective measures for flicker, it is not as perceptually salient due to camera motion throughout the sequence.

It is also informative to look at the raw temporal response of a flickering

**Table 4.5**: Percentage of patches detected as having motion across all experiments.

| Dataset | Percentage of patches | | |
|---|---|---|---|
| | Min | Max | Mean |
| *Balloons* | 13.9 | 22.0 | 20.4 |
| *Book_Arrival* | 10.3 | 13.2 | 11.6 |
| *Outdoor* | 10.1 | 14.2 | 13.1 |
| *Poznan_Hall2* | 8.4 | 12.9 | 11.6 |
| *Poznan_Street* | 6.9 | 8.9 | 7.8 |

pixel for the two processing methods, as plotted in Fig. 4.12. Curves showing the response from the source video as well as the bicubic interpolated test view are also shown for comparison. The chosen pixel is from a static object in each dataset, which accounts for the relatively flat response of the source pixel for Figs. 4.12c and 4.12e. The source pixel response in Fig. 4.12d is also flat despite camera motion throughout the sequence because the pixel trajectory is across a large object uniform in color. In Figs. 4.12a–4.12b, camera motion and an illumination change, respectively, cause the pixels to have a non-flat temporal response.

In all cases, the proposed method pixel response is generally close to that of the source. There is often a large bias between the source response and the response from the interpolated test view, which reflects the inaccuracy inherent in simple interpolation. Significant flicker can be observed for all datasets for frame-by-frame processing, with pixel transitions of up to 30% of the dynamic range. The frame-by-frame processing produces pixels that often switch between two values. This toggling is due to the inability of the patch matching to distinguish between two best candidates, whereas the ambiguity is resolved in the proposed approach by using neighboring frames in the video.

Another important advantage of temporal processing is the ability to exploit inter-frame correlation for more efficient patch matching. As discussed in Section 4.2, we reap this benefit by only searching for patch matches where motion is detected; otherwise, only the match error for the 10 best candidates from the previous node has to be recomputed. In Table 4.5, we list the percentage of patches that were detected as having motion across all of the experiments discussed in this chapter. The minimum, maximum, and average percentage of detected patches is given for each dataset. For the *Poznan_Street* sequence, there is not a lot of motion, and so less than 8% of the patches needed to be fully matched. Even for the *Balloons* sequence with much camera and object motion, only about 1 out of every 5 patches had to be fully matched. Since patch matching is an expensive operation and many videos have large static regions, motion detection results in significant savings. Overall, only 12.9% of patch matches had to be computed for all datasets and experiments.

## 4.4 Discussion

We designed the experiments in Section 4.3 to validate the performance of our algorithm, and to provide insight into how super-resolution affects mixed resolution compression. In this section, we compare our results to other work, interpret the implications of our findings for mixed resolution compression, and cite areas of future development.

While the difference in focus among prior art and the proposed method precludes a direct and conclusive comparison, we make some notes about relative performance where similar data or procedures have been used. For the *Balloons* sequence, we achieve a 2.7 dB gain over [64], even though more favorable testing conditions were used in the latter (lower resolution data, no anti-aliasing filter). In [68], a maximum gain of 2.6 dB over bicubic interpolation is reported. We report a maximum gain of 6.1 dB and an average gain of 3.8 dB over bicubic interpolation for the same downsampling factors. Depth information is also assumed to be given in [68]; however, the experiments were run on different data. The same camera views and resolution of the *Poznan_Street* sequence was used in [28], which yields a 3.7 dB and 5.9 dB gain over bicubic interpolation for downsampling the test view by a factor of 2 and 4, respectively. Our method produces an additional gain of 1.2 dB and 0.2 dB for the same downsampling ratios over [28], which also uses less prefiltering and assumes given depth information. We additionally achieve similar or higher gains for compressed data. Compared to our earlier work in [62], the proposed method garners a mean increase of 4.2 dB PSNR and 0.07 SSIM index when run on the same data under the same conditions. Further, less than 20% of the patches had to be matched versus our previous method. The objective metric gains are mostly owed to increases in efficiency, such as match propagation and chunk processing, that allow for finer searches for matching patches.

In MRSC, there are two axes along which the bit rate can be independently controlled. The downsampling factor of the low resolution view can be adjusted, and the QP of both views can be adjusted. A given bit rate can be met by a multitude of combinations of these two parameters, all of which strike some balance between sharpness and coding artifacts. The results of our experiments can help one to navigate this space of trade-offs by showing how decoder-side super-resolution affects the rate distortion curves. For instance, when the QP is very high, super-resolution is limited by the excess of coding artifacts as discussed in Section 4.3.3. We also know from Section 4.3.2 that higher downsampling ratios give diminishing returns on the bit rate. These results imply that, if the target bit rate can be met, higher quality compression in conjunction with higher downsampling ratios is a better combination for super-resolution. In this region of the parameter space, super-resolution performance has the most potential for improvement, and downsampling has the most potential for bit rate reduction. As seen in Figs. 4.6–4.7, the proposed method can handle large downsampling ratios, albeit with some distortion, but is completely bound by severe compression, as in Fig. 4.10. Further, especially for stereo, blur is a preferable degradation than coding artifacts. Of course, whether the target bit rate can be met by light compression and heavy downsampling will depend on the rate distortion for the particular source video. In that case, curves such as the ones in Figs. 4.8–4.9 can help gauge performance.

Our algorithm for super-resolving mixed resolution stereo video can be further developed in several ways. For very low resolution sequences, temporal super-resolution can be applied to the low resolution video alone, then the other stereo view can be used to further increase the resolution. This method would require that the encoder not anti-alias the downsampled sequence so that the high frequency components remain in the low resolution sequence. Our method could be made more robust by implementing a scene change detection algorithm. If the scene changes are known, then the algorithm could properly reset and also divide the temporal chunks used in MRF optimization accordingly. The search for correspondence between the views can also be restricted by propagating disparity information from previous frames. Patch motion would have to be known but could be extracted from the encoded video. Restricting the search range also enforces greater spatial correlation, which may help with dealing with flat regions in the video. Currently, our algorithm can generate false textures when few features are available for matching.

Another important area of development would be to employ a degradation model for compression as a function of input resolution and QP. As described in Section 4.2 and depicted in Fig. 4.1, the full resolution view undergoes the same decimation procedure after compression that the low resolution view does before compression. The two views are then interpolated and matches are sought between them. However, this processing chain implicitly approximates the compression stage as a linear and shift-invariant system, since the matching stage assumes that the two views contain very similar frequency content. This approximation may hold at higher bit rates, but will contribute to rendering error as the bit rate falls. A model of how the codec degrades quality of the input could be applied so that the two views are more similar in quality before performing stereo matching. This corrective step could improve super-resolution at lower bit rates. Depending on the degradation relationship between resolution and QP, it may even be possible that better rendering quality is attained for compressing the low resolution view with a lower QP than the high resolution view in order to equalize the quality at the decoder.

## 4.5 Conclusion

While mixed resolution coding offers an attractive economy on bandwidth for a small sacrifice in sharpness, the discarded high frequency information from the impoverished view has utility in certain applications. Existing literature has dealt with variations of the MRSC problem, but has not considered either the temporal dimension of video, analysis of compressed data, or treatment of the stereo-specific problem. The aim of the present work was to fill that void.

We proposed a method to recover the lost high frequencies in the low resolution video by efficiently using information from the full resolution view. Our method uses temporal information that can aid in stereo matching, minimize repeated computation, and enforce temporal consistency of the reconstructed data. Experiments validating our parameter choices, algorithmic performance, and improvements due to temporal processing were presented.

Our experimentation with, and analysis of, compressed video data add new considerations for MRSC. Demonstrating agreeable super-resolution performance at the decoder without the use of accompanying depth or side information can reduce bandwidth and improve video quality. These factors contribute to the viability of mixed resolution as a compression technique for stereo video.

## Acknowledgments

The text of Chapter 4 is, in part, a reprint of a paper that is being prepared for publication as: A. K. Jain and T. Q. Nguyen, “Video super-resolution for mixed resolution stereoscopic Coding,” *IEEE Trans. Circuits Syst. Video Technol.*, in preparation. The dissertation author was the primary investigator and author of this paper.

# Chapter 5

# Efficacy of Motion in Disambiguating Stereo Matches

Stereo matching, the computational process of finding correspondences between two different perspectives of a scene, is a critical part of virtually any stereo processing application. It is employed in traditional signal processing tasks for stereo, such as compression, denoising, and super-resolution [62] (also see Chapter 4). Stereo matching is also the process by which relative depth or disparity information is derived, which is used explicitly in many computer vision and rendering tasks such as view synthesis [60, 61], retargeting, object matting, 3D model reconstruction, and user interaction.

Cues such as image structure (edges), color, and luminance are used to match regions of a stereo image [84]. Since this registration occurs between two viewpoints in space rather than two different points in time, as in 2D image registration, it is possible to use motion as an additional matching cue. Intuitively, matching spatio-temporal blocks should provide better discrimination of the disparity signal than performing exclusively spatial (single frame) block matching alone. The benefit arises from the fact that matching blocks have to agree along three dimensions, and so motion can help distinguish between spatially similar blocks. Enforcing compatibility along this third dimension decreases the probability of false matches.

This concept is illustrated in Fig. 5.1, where a reference patch from the

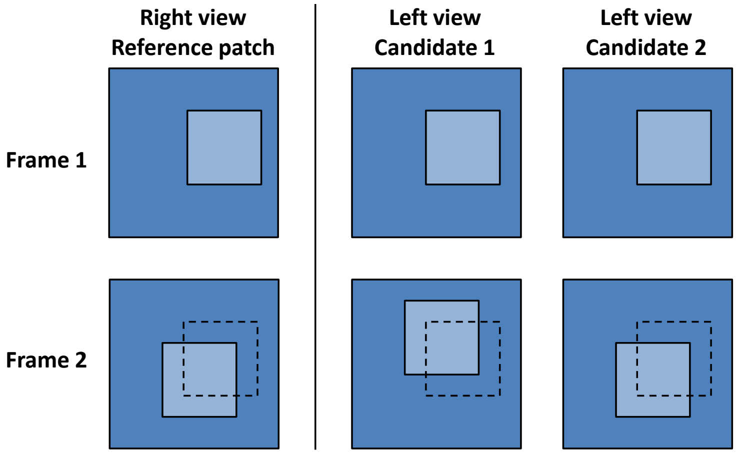


**Figure 5.1**: A reference patch in one stereo view may match several candidates in the other. By using multiple frames, candidate patches are much easier to distinguish due to motion.

right stereo view has two candidate matches in the left view. The large, dark area indicates the spatial extent of the patch and the lighter square indicates a feature within it. On Frame 1, the three patches are spatially identical, and a decision between the two candidate patches would be based on noise alone. On the next frame, however, the first candidate patch has a different motion than the reference patch while the second candidate has a matching motion. Thus, Candidate 2 would be selected based on both frames of data.

In this chapter, we study the nature and extent of the benefit garnered by spatio-temporal stereo registration. In the first part of our analysis, we examine the underlying assumptions of spatio-temporal stereo matching and use stereo geometry to derive the conditions under which these assumptions are valid. In the main part of the chapter, we calculate the probability of matching error as a function of image noise, number of frames in the spatio-temporal block, probability of motion, and image features. We treat motion as a random process, and give specific results for two motion models: motion vectors that are random at every frame, and motion vectors that remain constant from frame to frame. Our theoretical analysis is verified through simulation and experiments on stereo videos. Together, these components of our study give insight into the spatio-temporal stereo matching problem, can be used to select parameters for algorithms, and are also applicable beyond the stereo domain. For ease of analysis and applicability to the most common scenario, we restrict ourselves to the rectified stereo case.

The remainder of the chapter is organized as follows. We discuss literature relevant to our work in the areas of stereo matching, spatio-temporal applications, and image registration in Section 5.1. Next, in Section 5.2, we present our analysis of the assumptions inherent in spatio-temporal stereo matching. In Section 5.3, we develop the data model and derive results for the probability of match error. We validate our theoretical results and observations through simulation and experimentation on stereo video in Section 5.4. We interpret these results in Section 5.5, where we discuss the context of spatio-temporal matching, related issues, and applications that can benefit from this work. Finally, we conclude in Section 5.6 with a summary of our key observations and avenues for future research.

## 5.1 Related Work

Our work is related to the areas of spatio-temporal disparity estimation and its applications. In an early piece on spatio-temporal stereo [85], the 3D structure and motion in a multiview camera setup is simultaneously recovered using spatio-temporal matching of object patches across cameras. A straightforward extension from 2D to 3D block matching was tested for structured light and passive stereo for scenes with different motion characteristics in [86]. A new matching template and corresponding cost function was recently proposed in [71], in which voxels are filtered with differently oriented spatio-temporal filters to account for motion, and matched across binocular views under geometric constraints.

Enforcing temporal smoothness for disparity has been employed directly within optimization frameworks. The authors in [87] minimize a cost function consisting of data and temporal consistency terms using dynamic programming iteratively applied to the spatial and temporal directions. In [69], disparity is computed by incorporating data and occlusion costs into a 3D Markov Random Field (MRF) model and optimizing the network probabilities. A similar framework is used in [70], which fuses time-of-flight depth maps with stereo imagery.

Techniques for video disparity estimation have been proposed that compute frames sequentially based on prior estimates. Disparity flow, the changes in optical flow and disparity under geometric and constant brightness constraints, is used in [88] to cross-validate the disparity estimates for the current frame and to predict disparity for the next frame. Temporal consistency is enforced by penalizing disparity hypotheses on the current frame that are different from the predicted value. Similarly, the work in [89] attaches a penalty on changes from the previous frame's motion-compensated disparity map.

While the aforementioned works produce video disparity estimates directly, other methods have found success in a two-pass approach, where disparity is first estimated frame-by-frame using existing image disparity methods and then refined using temporal information. In [90], bundle optimization is applied to the frame-by-frame estimates with geometric constraints between frames for videos of static scenes. The result is further optimized with spatial, temporal, and sparse correspondence constraints on each frame. A signal processing approach is taken in [73], where spatio-temporal total variation minimization is applied to the volume of frame-by-frame computed disparity to remove the noisy, spurious estimates. Similarly, the authors in [91] use an edge-preserving spatio-temporal filter in matching cost space formed by block matching in order to promote more consistent estimates.

The popular image disparity method in [92], which forms an adaptive support window by aggregating costs based on Gestalt similarity of the neighboring pixels, has given rise to recent extensions to the video domain. The approach is reformulated in a joint bilateral filtering framework in [93] and extended to video by aggregating costs over a five-frame window using Gaussian weights. The method in [94] forms a weighted combination of costs in current and previous frames depending on image noise and photometric similarity. Rather, in [95], motion is included as a Gestalt parameter by temporally aggregating costs based on similarity of motion.

The methods in [69–71, 73, 85–91, 93–95] discussed thus far all naturally involve motion, either implicitly or explicitly, in their formulation. While the present study does not address how to estimate disparity from a video sequence, it deals with the question of how valuable motion is to the estimation quality. Accordingly, the results can be applied to most any spatio-temporal disparity estimation problem in the algorithm design or parameter selection.

Although spatio-temporal matching between video streams is most common in the realm of stereo or multiview imaging, a similar process can be useful within monocular video. For instance, single-video super-resolution [96], inpainting [97], or denoising [98] all involve searching for spatio-temporal matches within a single video. Though our assumptions and results are specific to the stereo video case, they can be easily generalized to other scenarios.

Since disparity estimation is a special case of image registration, our work is also informed by, and has bearing on, this classic problem as well. In particular, prior work has been conducted in establishing bounds on registration performance. In [99], the limits of estimation accuracy of motion vectors using a global translation model are found. Based on this work, limits for registration accuracy for the stereo image case, or disparity estimation, were found in [100], and for slanted surfaces imaged by multiple cameras in [101]. Our focus is distinct from these works in that we analyze the role of motion in registration in stereo video, whereas prior art dealt exclusively with images. Further, we are concerned with the ability of motion as a discriminating agent, not explicitly with registration accuracy.

Our analysis is most closely connected to the experiments in [72, 102, 103]. The work in [103] uses time-varying information in an underwater scenario to disambiguate stereo matching, and shows how multiframe optical flow for stereo matching can resolve the aperture problem. The authors in [72] examined the performance of spatio-temporal disparity estimation, in terms of percent correct disparity pixel assignments, using block matching under different matching criteria and noise conditions. In [102], similar experiments were conducted, varying the spatial and temporal dimensions of the matching window. The studies in [91, 93, 94] also contain experiments for disparity estimation performance versus additive image noise, and all record improved performance when temporal information is used.

By contrast, our work examines disparity estimation performance, specifically discriminability, from a theoretical perspective beginning from a model of the data and the error signal. Additionally, the present study is much more thorough in its analysis and extensive in terms of the parameters considered. Our results can explain many of the observations noted in [72, 102, 103].

## 5.2 Assumptions Inherent in Spatio-Temporal Stereo Matching

Spatio-temporal stereo block matching presupposes two facts about the scene and (rectified) imaging system: motion is identical in the left and right cameras, and the disparity signal is constant over the time interval of the block. If either of these assumptions were false, then matching across frames would influence the cost function in a manner not representative of the parameter of interest, namely horizontal disparity. Before spatio-temporal matching is employed in disparity estimation, these two assumptions must be verified in order for the matching results to be meaningful. This section serves as a guide for determining when spatio-temporal matching is valid.

Beginning with general 3D motion as captured by a rectified stereo camera, we show that identical motion vectors and constant disparity are in fact the same requirement. Next, we show how this requirement can be met given the sampling parameters of the imaging system, and analyze the range of validity for three different real cameras. While the relationship between motion vectors and disparity has been studied before, most similarly in [71, 88], we additionally consider the sampling properties of the imaging system and the object velocities of the scene in the context of spatio-temporal block matching. Some of this analysis is an extension of our work in [62].

### 5.2.1 Derivation of Motion-Disparity Relationship

Consider a rectified stereo camera with focal length $f$ and baseline $b$ in meters and a coordinate system with the origin at the optical center of the left

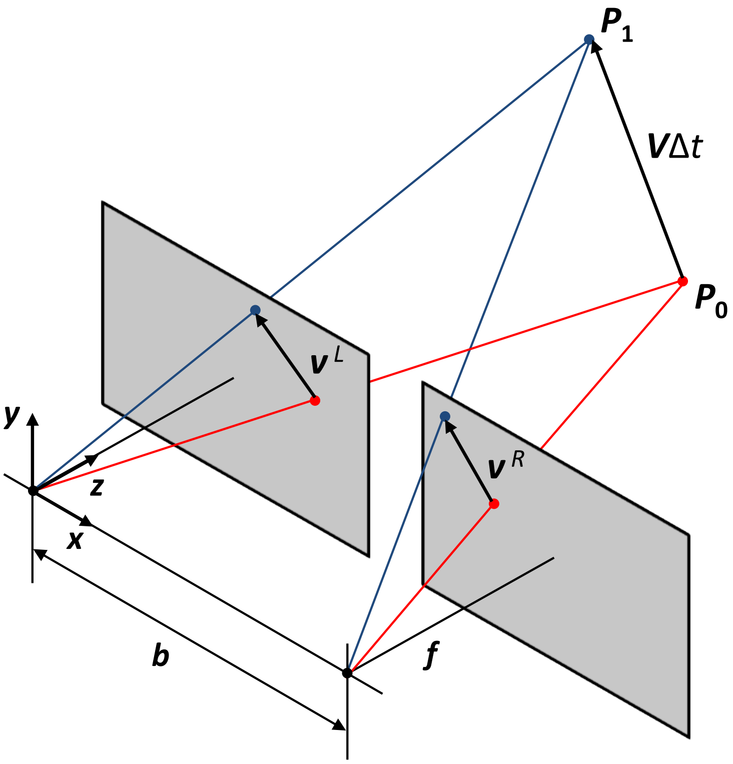


**Figure 5.2**: Pinhole camera model of rectified stereo rig imaging a point moving over a time interval $\Delta t$. The coordinate system and induced motion vectors in each image plane are also shown.

camera, as in Fig. 5.2. A point $\boldsymbol{P}$ travels from $\boldsymbol{P}_0$ at time $t_0$ to $\boldsymbol{P}_1$ at time $t_1 = t_0 + \Delta t$. The endpoints of this path are related by

$$\boldsymbol{P}_1 = \boldsymbol{P}_0 + \boldsymbol{V}\Delta t, \tag{5.1}$$

where $\boldsymbol{V} = \begin{bmatrix} V_x & V_y & V_z \end{bmatrix}^\top$ is a unit vector in the direction of travel with units meters/second.

Let the point $\boldsymbol{P}$ have world coordinates given by $\boldsymbol{P}_i = \begin{bmatrix} X_i & Y_i & Z_i \end{bmatrix}^\top$ at time $t_i$. The image of this point in the left and right image planes has coordinates

$$\boldsymbol{p}_i^L = \frac{f}{Z_i}\begin{bmatrix} X_i \\ Y_i \end{bmatrix} \tag{5.2}$$

$$\boldsymbol{p}_i^R = \frac{f}{Z_i}\begin{bmatrix} X_i + b \\ Y_i \end{bmatrix}. \tag{5.3}$$

The right image point position is shifted by $\frac{fb}{Z_i}$, the disparity of $\boldsymbol{P}_i$, since the relative camera pose consists of an identity rotation and a purely horizontal translation.

By definition, the $x$-$y$ motion vector is the displacement of the point $\boldsymbol{P}$ in image coordinates over a given time interval. As indicated in Fig. 5.2, the motion vector in each camera with respect to the time interval $\Delta t$ is given by

$$\boldsymbol{v}^C = \boldsymbol{p}_1^C - \boldsymbol{p}_0^C, \qquad C = L, R. \tag{5.4}$$

To derive the relationship between the motion vectors and disparity, we examine the motion vector difference:

$$\boldsymbol{v}^L - \boldsymbol{v}^R = \frac{f}{Z_1}\begin{bmatrix} -b \\ 0 \end{bmatrix} - \frac{f}{Z_0}\begin{bmatrix} -b \\ 0 \end{bmatrix} \tag{5.5}$$

$$= \begin{bmatrix} fb\left(\frac{1}{Z_0} - \frac{1}{Z_1}\right) \\ 0 \end{bmatrix}. \tag{5.6}$$

The $y$-component of the motion vector difference is identically zero; there is no stereo resolution in the vertical direction. The $x$-component is equal to the difference in disparity over the interval $\Delta t$. The motion vectors in each camera for corresponding points will be identical if and only if $Z_0 = Z_1$. Thus, the assumption of identical motion vectors in each camera is the same as the assumption of constant disparity over the patch interval.

Our analysis shows that spatio-temporal stereo block matching inherently assumes that objects undergo fronto-parallel motion over the duration of the block. When this assumption is satisfied, identical motion vectors and constant disparity is guaranteed. Note that any block matching algorithm is predicated upon translational, fronto-parallel motion with respect to the image plane. Therefore, spatio-temporal stereo matching imposes no further assumptions on the scene motion beyond the translational motion model of block matching.

### 5.2.2 Validity Conditions of the Constant Disparity Requirement

As shown in Section 5.2.1, object disparity must remain constant in order for spatio-temporal block matching to work. However, the finite sampling rates of the image sensor and disparity estimation scheme allow this condition to be met while admitting some radial motion: disparity need only be *undetectable* rather than identically zero. Here we examine the conditions under which the constant disparity assumption in Section 5.2.1 is valid.

Let $\mu$ be the pixel pitch of the sensor in meters/pixel and $r$ be the resolution of the disparity estimation algorithm in pixels. For example, half-pixel disparity estimation corresponds to $r = 0.5$ and for integer disparity, $r = 1$. If we require the observed object disparity to be undetectable, then the disparity difference over $\Delta t$ should be less than $r$. That is, based on the $x$-component in (5.6):

$$\frac{fb}{\mu}\left(\frac{1}{Z_0} - \frac{1}{Z_1}\right) < r. \tag{5.7}$$

Using the relation in (5.1) and solving the inequality in (5.7), we bound $V_z$ as

$$V_z < \frac{1}{t}\left(\frac{Z_0^2}{Z_{max} - Z_0}\right) \tag{5.8}$$

$$Z_{max} = \frac{fb}{\mu r}. \tag{5.9}$$

The quantity $Z_{max}$ is the stereoacuity of the stereo camera, the maximum distance at which an object can still induce a detectable disparity (of $r$ pixels) between its images in the two cameras of the stereo pair. When the condition in (5.8) is satisfied, an object beginning at a distance $Z_0$ from the camera and receding at

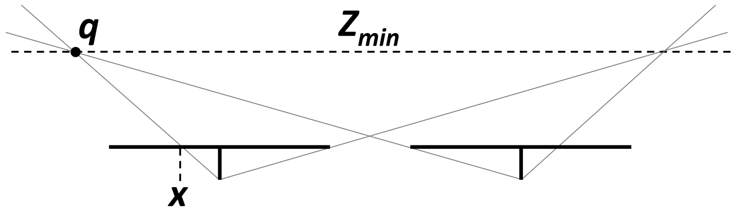


**Figure 5.3**: Rectified stereo rig imaging a point $q$ at the minimum imaged depth plane $Z_{min}$, which appears at point $x$ in the left camera.

speed $V_z$ will induce less than $r$ pixels of disparity difference over the duration $t$.

Since the bound in (5.8) decreases for smaller $Z_0$, the constant disparity assumption is more likely to be violated by objects close to the camera. However, there is a limit as to how close objects can be imaged since good stereo requires significant overlap between the fields of view of the two cameras. To determine this near limit, consider the point $q$ in Fig. 5.3. This point appears at position $x$ pixels in the left camera and position 0 (the very left edge) in the right camera. Letting $Z_{min}$ be the minimum distance from the stereo rig at which objects appear, the disparity induced by an object at $q$ is

$$d_{max} = \frac{fb}{\mu Z_{min}} = x. \tag{5.10}$$

We can express $x$ as a fraction $w$ of the horizontal resolution $N$ of the camera. Then $w$ describes the maximum width of the stereo border, the region that lacks correspondence with the other camera, as a fraction of the sensor resolution. We have $Z_{min}$ as a function of $w$:

$$Z_{min} = \frac{fb}{\mu N w}. \tag{5.11}$$

The expressions in (5.8), (5.9), and (5.11), along with their constituent parameters, characterize a stereo system for spatio-temporal block matching. Table 5.1 lists the parameters and corresponding $Z_{min}$ and $Z_{max}$ values for hypothetical stereo rigs constructed from three different types of cameras: a point-and-shoot (Panasonic LUMIX ZS30 [104]), a DSLR (Canon EOS Rebel T4i [105]), and a professional camera (Red Epic Dragon [106]). The table assumes a baseline of

**Table 5.1**: Different cameras, their parameters, and associated speed limits for spatio-temporal block matching.

| Camera Model | $N$ | $\mu$ | $Z_{min}$ | $Z_{max}$ |
|---|---|---|---|---|
| | (pixels) | ($10^{-6}$ m) | (m) | (m) |
| Panasonic Lumix ZS30 | 1920 | 1.26 | 4.17 | 2000 |
| Canon EOS Rebel T4i | 1920 | 4.30 | 1.22 | 586 |
| Red Epic Dragon | 6144 | 5.00 | 0.33 | 504 |

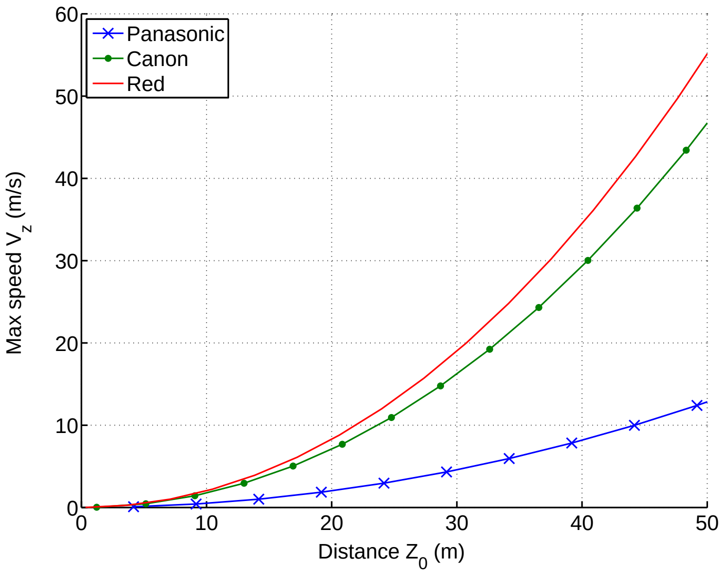


**Figure 5.4**: Maximum radial velocity as a function of object distance for three different cameras. The curves represent the maximum velocity at which an object can recede from the camera in order for spatio-temporal stereo cost aggregation to be a valid measure of horizontal disparity.

$b = 63$ mm (equal to human interocular distance), $w = 0.25$, $r = 1$ (integer disparity estimation), $t = 0.1$ seconds, and $f = 40$ mm. The horizontal resolutions $N$ listed are what each camera supports in video mode.

A plot of the bound on radial velocity versus object distance in (5.8) using the parameters in Table 5.1 is given in Fig. 5.4. The small pixels of the Panasonic theoretically make disparity very distinguishable, resulting in low velocity bounds.

The other two cameras have larger pixels and therefore smaller stereoacuity, and so the velocity bounds become harder to violate. Clearly, the constant disparity assumption must be verified prior to using spatio-temporal block matching for disparity computation.

Our analysis here can be used to determine the validity of the constant disparity assumption, and to select the maximum number of frames that can be used in a spatio-temporal block for a given stereo system and expected object speeds. It is important to remember that the velocity bound applies only to the radial component of motion, which is less dominant for the many applications where cameras are oriented to capture interesting events across the field of view.

## 5.3 Discriminability of Spatio-Temporal Stereo Matching

Once the assumptions of spatio-temporal stereo matching can be verified for a given scene and imaging system, a suitable disparity estimation algorithm can be applied. Here, we quantitatively analyze the situation depicted in Fig. 5.1. Using the rectified stereo image assumption and treating motion as a stochastic process, we set up a model for the data. Next, we define the error signal and analyze its magnitude in terms of image features and motion. Finally, we answer the question of how informative motion is in discriminating patches by deriving the probability of false matches. Our results are applicable for any motion model, but we analyze the random motion and constant motion cases in particular.

### 5.3.1 Data Model

We denote the observed pixel values at discrete spatial coordinate $\boldsymbol{p}$ of the $i$th frame of the left and right views of the video as $z_i^L(\boldsymbol{p})$ and $z_i^R(\boldsymbol{p})$, respectively. The observed video frames $z_i^L(\boldsymbol{p})$ and $z_i^R(\boldsymbol{p})$ are the sum of the underlying image signal $f(\boldsymbol{p})$, shifted by the appropriate motion and disparity vectors, and white Gaussian noise. We assume that all noise samples $\epsilon_i^L(\boldsymbol{p})$, $\epsilon_i^R(\boldsymbol{p}) \sim \mathcal{N}(0, \sigma_n^2)$ are independent and identically distributed (i.i.d.).

The left and right motion vectors for each frame, $\boldsymbol{v}_i^L(\boldsymbol{p})$ and $\boldsymbol{v}_i^R(\boldsymbol{p})$, represent the displacement from the previous frame, with the convention that $\boldsymbol{v}_0^L = \boldsymbol{v}_0^R = \mathbf{0}$. The total displacement due to motion of the scene with respect to the first frame is denoted by $\boldsymbol{m}_i^L(\boldsymbol{p})$ and $\boldsymbol{m}_i^R(\boldsymbol{p})$.

The motion vectors and underlying image signal in each view are related by the (horizontal) disparity vector, $\boldsymbol{d}(\boldsymbol{p})$. The image signal is referenced to the first frame of the left camera, and the disparity vector represents the right-to-left displacement of corresponding pixels. The spatio-temporal block has dimensions $J \times K \times T$, the pixel index $\boldsymbol{p}$ ranges over the spatial region defined by $J \times K$, and the frame index $i$ ranges from 0 to $T-1$. Thus, we arrive at the data model:

$$z_i^L(\boldsymbol{p}) = f\left(\boldsymbol{p} - \boldsymbol{m}_i^L(\boldsymbol{p})\right) + \epsilon_i^L(\boldsymbol{p}) \tag{5.12}$$

$$z_i^R(\boldsymbol{p}) = f\left(\boldsymbol{p} + \boldsymbol{d}(\boldsymbol{p}) - \boldsymbol{m}_i^R(\boldsymbol{p})\right) + \epsilon_i^R(\boldsymbol{p}) \tag{5.13}$$

$$\boldsymbol{m}_i^C(\boldsymbol{p}) = \sum_{k=0}^{i} \boldsymbol{v}_k^C(\boldsymbol{p}), \qquad C = L, R \tag{5.14}$$

$$\boldsymbol{v}_i^R(\boldsymbol{p}) = \boldsymbol{v}_i^L(\boldsymbol{p} + \boldsymbol{d}(\boldsymbol{p})). \tag{5.15}$$

We treat the motion vectors as random variables identically distributed across all pixels with some probability distribution, and assume that the motion vectors at non-corresponding pixels in each view are uncorrelated. For instance, at point $\boldsymbol{p}$, $E\left[\boldsymbol{v}_i^L(\boldsymbol{p}+\boldsymbol{d}')\boldsymbol{v}_i^R(\boldsymbol{p})\right] = E\left[\boldsymbol{v}_i^L(\boldsymbol{p}+\boldsymbol{d}')\right] E\left[\boldsymbol{v}_i^R(\boldsymbol{p})\right]$ for any shift $\boldsymbol{d}' \neq \boldsymbol{d}(\boldsymbol{p})$. The aggregate motion, $\boldsymbol{m}_i^L(\boldsymbol{p})$ and $\boldsymbol{m}_i^R(\boldsymbol{p})$, is then the sum of $i$ random variables and is a random walk model of motion.

### 5.3.2 Analysis of Error Signal

We consider the popular sum of squared differences (SSD) error measure in this analysis:

$$\delta\left(\boldsymbol{d}'\right) = \sum_{\boldsymbol{p},i} \left(z_i^L(\boldsymbol{p}+\boldsymbol{d}') - z_i^R(\boldsymbol{p})\right)^2. \tag{5.16}$$

Though optimization, occlusion handling, or other more sophisticated techniques for stereo matching are typically applied, block matching using the SSD error metric is a common first step in forming a data cost for stereo matching. The goal of spatio-temporal block matching is to find the $\boldsymbol{d}'$ that minimizes (5.16) for each point $\boldsymbol{p}$ in the image. On average, $\boldsymbol{d}' = \boldsymbol{d}(\boldsymbol{p})$ is the minimizing disparity, although errors can occur due to image noise.

In the case when two patches are spatially dissimilar, the error signal is high relative to the noise and incorrect disparities are thus easily discarded. Here, we examine the error signal for the more interesting case when two blocks appear spatially similar on one frame, and thus noise affects the outcome. That is, we examine the case where

$$f(\boldsymbol{p} + \boldsymbol{d}') = f(\boldsymbol{p} + \boldsymbol{d}(\boldsymbol{p})), \qquad \boldsymbol{d}' \neq \boldsymbol{d(p)} \tag{5.17}$$

where $\boldsymbol{p}$ ranges over the spatial extent of the block. The expression in (5.17) states that, except for noise, a block from the left view is identical to a block in the right view on its first frame, but may of course have differing motion subsequently. However, note that since the distribution of motion vectors is assumed to be identical everywhere, the moments of the image signal with respect to the motion distribution are equal.

To determine the strength of the error signal, we average over the distribution of motion vectors and denote this operator as $E_{\boldsymbol{m}}[\cdot]$. In the following, we use the shorthand $\Delta f_i(\boldsymbol{p}, \boldsymbol{d}') = f\left(\boldsymbol{p} + \boldsymbol{d}' - \boldsymbol{m}_i^L(\boldsymbol{p})\right) - f\left(\boldsymbol{p} + \boldsymbol{d}(\boldsymbol{p}) - \boldsymbol{m}_i^R(\boldsymbol{p})\right)$ and $\Delta\epsilon_i = \epsilon_i^L(\boldsymbol{p}_1) - \epsilon_i^R(\boldsymbol{p}_2)$ for any points $\boldsymbol{p}_1$, $\boldsymbol{p}_2$. For any incorrect disparity $\boldsymbol{d}'$, the motion-averaged error signal is

$$\begin{aligned} E_{\boldsymbol{m}}\left[\delta\left(\boldsymbol{d}'\right)\right] &= \sum_{\boldsymbol{p},i} E_{\boldsymbol{m}}\left[\left(z_i^L\left(\boldsymbol{p} + \boldsymbol{d}'\right) - z_i^R(\boldsymbol{p})\right)^2\right] &(5.18)\\ &= \sum_{\boldsymbol{p},i} E_{\boldsymbol{m}}\left[\left(\Delta f_i(\boldsymbol{p}, \boldsymbol{d}') + \Delta\epsilon_i\right)^2\right] &(5.19)\\ &= \sum_{\boldsymbol{p},i} 2\sigma_{\boldsymbol{m},i}^2(f) + \Delta\epsilon_i^2, &(5.20)\end{aligned}$$

where $\sigma^2_{\boldsymbol{m},i}(f)$ is the variance with respect to the motion distribution of a pixel on the $i$th frame. We used the fact that image noise is independent of motion, the distribution of motion vectors is identical everywhere, and the motion vectors for non-corresponding locations (incorrect disparities) are uncorrelated.

From (5.20), we see that the error signal consists of two components: that due to motion and that due to image noise. We can evaluate the first error term by applying a second-order Taylor series expansion about the mean of random motion variables. We have

$$\sigma^2_{\boldsymbol{m},i}(f) \approx \boldsymbol{\nabla f}^\top \boldsymbol{\Sigma}_{\boldsymbol{m},i} \boldsymbol{\nabla f} \tag{5.21}$$

where $\boldsymbol{\Sigma}_{\boldsymbol{m},i}$ is the covariance matrix of the $x$ and $y$ motion components of the aggregate motion signal $\boldsymbol{m}_i^L$ and $\boldsymbol{\nabla f}$ is the gradient of the image evaluated at the mean of the distribution. The variance formulation in (5.21) compactly expresses a lot of information about how the motion and image structure affect spatio-temporal block matching. The signal strength (i.e. magnitude of error signal in (5.20)) depends on both the image structure and the motion distribution, which is a manifestation of the aperture problem. When the image patch contains gradients in one direction, the motion distribution must be spread along the perpendicular direction in order for spatio-temporal block matching to be beneficial for this feature type. Two-dimensional features, such as a corner, are most efficient in that any motion will be registered in the error signal.

Scenes with no motion clearly are not helped by spatio-temporal block matching. However, the presence of motion does not guarantee additional discriminability. Still scenes of faraway or isodistant objects with uniform motion, such as a panning shot of a landscape, have a motion distribution with no variance; thus, they are also bereft of the advantages of spatio-temporal block matching. Similarly, featureless patches do not benefit from spatio-temporal matching regardless of the motion velocity.

So far, we have made no assumption on the motion distribution, and all of our results depend only on its second-order characteristics. The random walk motion model in (5.14) allows the use of any probability distribution of motion and any correlation between steps of the walk. We analyze numerically two particular extreme cases: the motion vectors are independent and identically distributed at every frame, and the constant motion case where the motion vectors are perfectly correlated from frame to frame.

In the first case, the aggregate motion vector in (5.14) on each frame is a sum of $i$ i.i.d. random variables, and thus

$$\boldsymbol{\Sigma}_{\boldsymbol{m},i} = i\boldsymbol{\Sigma}_{\boldsymbol{v}} \tag{5.22}$$

where $\boldsymbol{\Sigma}_{\boldsymbol{v}}$ is the covariance matrix of one of the constituent motion vectors $\boldsymbol{v}_k$. Thus, we have

$$\sigma^2_{\boldsymbol{m},i}(f) \approx i\sigma^2_{\boldsymbol{v}}(f) \tag{5.23}$$

by inserting (5.22) into (5.21), where $\sigma^2_{\boldsymbol{v}}(f)$ is the variance due to the motion vector over the image. Combining (5.23) with (5.20), the error signal for this case becomes

$$E_{\boldsymbol{m}}\left[\delta\left(\boldsymbol{d}'\right)\right] = T(T-1)\sum_{\boldsymbol{p}} \sigma^2_{\boldsymbol{v}}(f) + \sum_{\boldsymbol{p},i} \Delta\epsilon_i^2. \tag{5.24}$$

In the second case, the constant motion model dictates that the motion vectors $\boldsymbol{v}_k$ are the same at every frame and (5.14) becomes

$$\boldsymbol{m}_i^C = i\boldsymbol{v}^C, \qquad C = L, R \tag{5.25}$$

with covariance matrix

$$\boldsymbol{\Sigma}_{\boldsymbol{m},i} = i^2\boldsymbol{\Sigma}_{\boldsymbol{v}}. \tag{5.26}$$

Following the same procedure as in the i.i.d. motion case, the error signal for the constant motion case becomes

$$E_{\boldsymbol{m}}\left[\delta\left(\boldsymbol{d}'\right)\right] = \frac{T(T-1)(2T-1)}{3}\sum_{\boldsymbol{p}} \sigma^2_{\boldsymbol{v}}(f) + \sum_{\boldsymbol{p},i} \Delta\epsilon_i^2. \tag{5.27}$$

Thus, we see that constant motion results in a much stronger error signal for a given temporal patch length $T$ versus i.i.d. motion. The constant motion signal grows as $T^3$ whereas the i.i.d. motion signal grows quadratically in $T$.

### 5.3.3 Discriminability for Two Candidate Patches

We have analyzed how motion affects the error signal when multiple patches are spatially similar to the reference patch. But how informative is motion in terms of ensuring the correct patch is selected in the presence of noise?

The error signal for spatially similar patches at an incorrect disparity is given by

$$\delta(\boldsymbol{d}') = \sum_{\boldsymbol{p},i} \left(\Delta f_i(\boldsymbol{p}, \boldsymbol{d}') + \Delta\epsilon_i\right)^2 . \tag{5.28}$$

For the correct disparity, the error signal will only be due to noise:

$$\delta(\boldsymbol{d}) = \sum_{\boldsymbol{p},i} \Delta\epsilon_i^2. \tag{5.29}$$

Note that $\Delta\epsilon_i \sim \mathcal{N}(0, 2\sigma_n^2)$, and so (5.28) is a noncentral chi-squared distribution with noncentrality parameter $\lambda$ given by

$$\lambda = \sum_{\boldsymbol{p},i} \frac{\Delta f_i^2(\boldsymbol{p}, \boldsymbol{d}')}{2\sigma_n^2}, \tag{5.30}$$

and (5.29) is a chi-squared distribution. The probability of choosing the wrong disparity is equal to the probability that the error for the correct disparity is greater than the error for incorrect disparity. Thus, defining the (independent) random variables $\mathcal{X} = \delta(\boldsymbol{d})$ and $\mathcal{Y} = \delta(\boldsymbol{d}')$, the probability of error is

$$\begin{align} P_{\mathcal{X}>\mathcal{Y}} &= \int_{-\infty}^{\infty}\int_{-\infty}^{x} f_{\mathcal{X},\mathcal{Y}}(x, y)\, \mathrm{d}y\, \mathrm{d}x \tag{5.31} \\ &= \int_{-\infty}^{\infty} F_{\mathcal{Y}}(x) f_{\mathcal{X}}(x)\, \mathrm{d}x \tag{5.32} \\ &= \int_{-\infty}^{\infty} \sum_{j=0}^{\infty} e^{-\lambda/2} \left(\tfrac{\lambda}{2}\right)^j \frac{1}{j!} F_{\chi^2_{N+2j}}\!\left(\tfrac{x}{2\sigma_n^2}\right) \frac{1}{2\sigma_n^2} f_{\chi^2_N}\!\left(\tfrac{x}{2\sigma_n^2}\right)\, \mathrm{d}x \tag{5.33} \\ &= \sum_{j=0}^{\infty} e^{-\lambda/2} \left(\tfrac{\lambda}{2}\right)^j I_j \tag{5.34} \end{align}$$

where $f_{\chi^2_N}$ and $F_{\chi^2_N}$ are the probability density function and cumulative distribution function for a chi-squared distribution (of standard normal variables) of degree $N$,

**Table 5.2**: Parameter values at extreme SNRs.

| SNR | $\sigma_n^2$ | $\lambda$ | $P_\lambda$ | $e^{-\lambda/2}\left(\frac{\lambda}{2}\right)^j$ |
|---|---|---|---|---|
| 0 | $\infty$ | 0 | 1 | $\delta_j$ |
| $\infty$ | 0 | 0 | $P_\emptyset$ | $\delta_j$ |
| | | $\infty$ | $1 - P_\emptyset$ | 0 |

and $N = J \cdot K \cdot T$ is the total number of samples within the spatio-temporal block. Note that $\mathcal{X}$ consists solely of noise samples and is therefore independent of $\mathcal{Y}$ regardless of motion. In the last step, we have changed the order of summation and integration, and also made the substitution $u = \frac{x}{2\sigma_n^2}$ in order to define

$$I_j \triangleq \frac{1}{j!} \int_{-\infty}^{\infty} F_{\chi^2_{N+2j}}(u) f_{\chi^2_N}(u)\, \mathrm{d}u. \tag{5.35}$$

Since $\lambda$ depends on the stochastic motion, the probability of error in (5.34) is itself random. Taking the expectation with respect to motion, we have

$$E_{\boldsymbol{m}}\left[P_{\mathcal{X}>\mathcal{Y}}\right] = \sum_{j=0}^{\infty} E_{\boldsymbol{m}}\left[e^{-\lambda/2}\left(\tfrac{\lambda}{2}\right)^j\right] I_j. \tag{5.36}$$

In a similar way, we can also compute the second moment of the probability of error in (5.34) and combine this expression with the square of (5.36) to get the variance of the probability of error:

$$\begin{aligned} \sigma^2_{\boldsymbol{m}}\left[P_{\mathcal{X}>\mathcal{Y}}\right] = \sum_{j=0}^{\infty}\sum_{l=0}^{\infty} \Big( & E_{\boldsymbol{m}}\left[e^{-\lambda}\left(\tfrac{\lambda}{2}\right)^{j+l}\right] - \\ & E_{\boldsymbol{m}}\left[e^{-\lambda/2}\left(\tfrac{\lambda}{2}\right)^{j}\right] E_{\boldsymbol{m}}\left[e^{-\lambda/2}\left(\tfrac{\lambda}{2}\right)^{l}\right]\Big) I_j I_l. \end{aligned} \tag{5.37}$$

Note that (5.36) and (5.37) are functions of $\lambda$, which is a measure of the SNR of the scene. By examining the endpoints of the range of SNR and the associated values of $\lambda$, we can understand the trends of the mean and variance of the probability of error.

When $\sigma_n^2 = \infty$, then $\lambda = 0$ with probability 1. Thus, $e^{-\lambda/2}\left(\frac{\lambda}{2}\right)^j = \delta_j$ where $\delta_j$ is the Kronecker delta function. Conversely, consider when $\sigma_n^2 = 0$ and the SNR

**Table 5.3**: Mean and variance of probability of error at extreme SNRs.

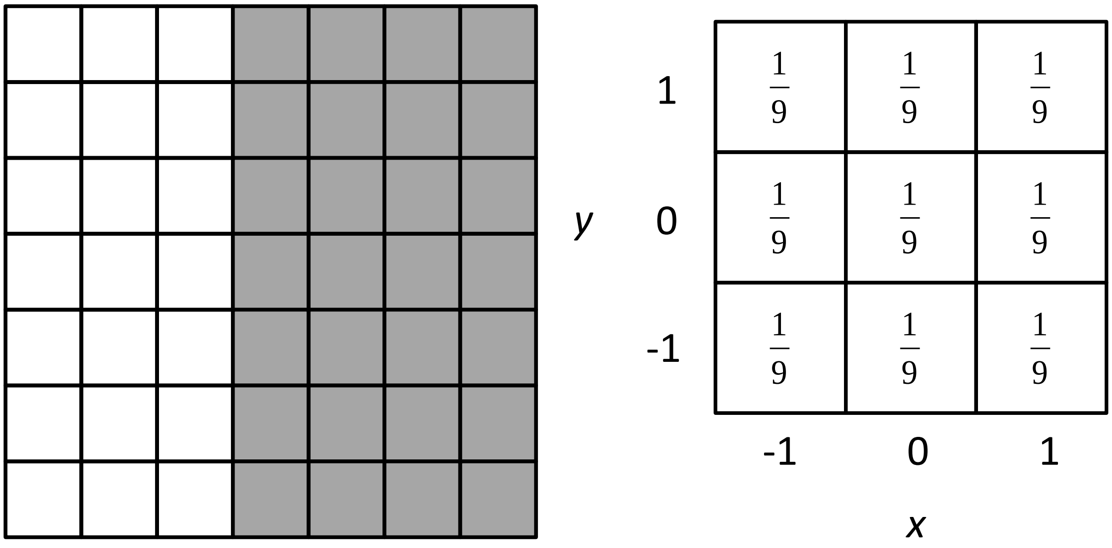
| SNR | $E_{\boldsymbol{m}}\left[P_{\mathcal{X}>\mathcal{Y}}\right]$ | $\sigma^2_{\boldsymbol{m}}\left[P_{\mathcal{X}>\mathcal{Y}}\right]$ |
|---|---|---|
| 0 | $\frac{1}{2}$ | 0 |
| $\infty$ | $\frac{1}{2}P_{\emptyset}$ | $\frac{1}{4}P_{\emptyset}\left(1-P_{\emptyset}\right)$ |

is infinite. If the difference signal over the frames is nonzero, then $\lambda = \infty$ and thus $e^{-\lambda/2}\left(\frac{\lambda}{2}\right)^j = 0$. If the difference signal is also zero, then $\lambda \to 0$ in the limit by l'Hôpital's Rule, and $e^{-\lambda/2}\left(\frac{\lambda}{2}\right)^j = \delta_j$. We summarize these parameter and function values at the extreme SNRs in Table 5.2, where $P_{\emptyset}$ denotes the probability of the event $\sum_{\boldsymbol{p},i} \Delta f_i^2(\boldsymbol{p}, \boldsymbol{d}') = 0$.

Using the values in Table 5.2, the expressions (5.36) and (5.37) can be easily evaluated for SNR $= 0$ and SNR $= \infty$. These limits are given in Table 5.3. At SNR $= 0$, the image is so swamped in noise that regardless of the motion or image features, the matching patch decision is purely based on chance and is thus equivalent to spatial matching for spatially identical patches. As the SNR increases, the mean probability of error asymptotically reaches the lower bound of $\frac{1}{2}P_{\emptyset}$. Thus, even in the absence of noise, there is potential for matching error due to the chance of an identical error signal for the false and correct match. The variance of the probability of error is 0 at SNR $= 0$ and approaches a finite value.

The expressions in Tables 5.2–5.3 are not specific to any motion model or parameter set, and the asymptotic expressions for mean and variance depend only on $P_{\emptyset}$. If the motion in the reference and candidate patch is the same, then the error signal $\Delta f_i(\boldsymbol{p}, \boldsymbol{d}')$ will be zero. However, even if the motion is not identical, the error signal could still be zero due to features leaving the patch window or motion in the direction of the image gradients. This is, in part, another consequence of the aperture problem. Thus, $P_{\emptyset}$ is lower-bounded by the probability of identical motion in each patch. This probability of zero error is finite, but decreases with number of frames in the spatio-temporal patch. Intuitively, a longer time sequence allows for a more complex motion path, which is less likely to occur in two patches by chance.

We plot the mean and variance of the probability of matching error in



(a)

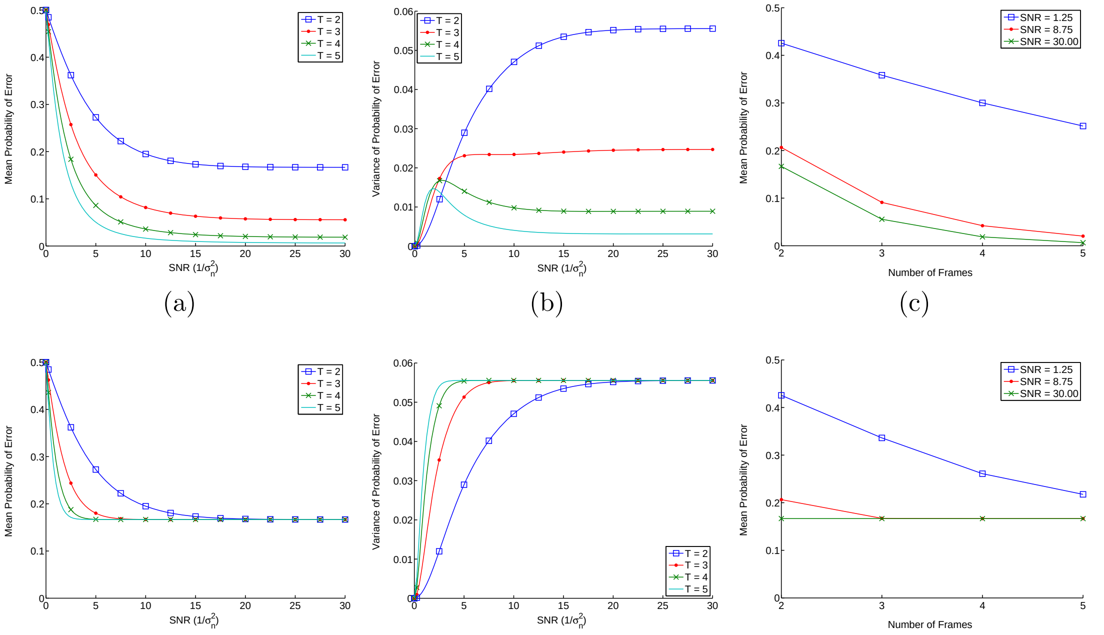
(b)

**Figure 5.5**: Experimental setup for theoretical and simulated data. (a) Spatial feature within 7 × 7 image patch. Feature is at full contrast (white represents 0 and gray represents 255). (b) Motion vector probability mass function.

Figs. 5.6–5.7 as a function of SNR, taken to be $\frac{1}{\sigma_n^2}$. However, the expressions (5.36) and (5.37) depend on $\lambda$, and hence image features, and the motion distribution as well. For example, a situation equivalent to SNR = 0 can occur when there is no motion, no motion perpendicular to the image gradients, when the image is featureless, or when the image is swamped in noise. Therefore, we must plot results for a particular image feature and motion distribution.

Figs. 5.6–5.7 are shown for a 7 × 7 spatial block size ($J = K = 7$) and for various values of $T$, the number of frames in the spatio-temporal block. The spatial pattern for the patch is a vertical edge covering 4 of the 7 columns, and is assumed to extend beyond the patch such that no new features are introduced into the window. The motion model is a discrete uniform distribution over [-1, 1] in both $x$ and $y$ directions independently. This means that the motion vectors have 9 equally likely values: 1 pixel in any of the 8 immediate directions or zero. The image patch and probability function are shown in Fig. 5.5.

The trends discussed above and summarized in Table 5.3 are illuminated in plots of the mean probability of error for the i.i.d. and constant motion models



(a) (b) (c)

(d) (e) (f)

**Figure 5.6**: Probability of matching error for two candidate patches for i.i.d. (top row) and constant (bottom row) motion models. (a), (d) Mean probability of matching error as a function of SNR for different number of frames. (b), (e) Variance of probability of matching error as a function of SNR for different number of frames. (c), (f) Mean probability of matching error as a function of number of frames for different SNRs.

in Figs. 5.6a and 5.6d. In Fig. 5.6a, we see that increasing the number of frames decreases the error probability. Since the feature does not leave the window, $P_\emptyset$ is equal to the probability of equal motion in each camera. At each frame, the probability of identical motion is $\frac{1}{3}$ since only horizontal motion affects the vertical patch feature. Since the motion is i.i.d. on every frame, the asymptotic limits are given by $\frac{1}{2}\left(\frac{1}{3}\right)^{T-1}$ as per Table 5.3.

In Fig. 5.6d, all curves approach the same limit of $\frac{1}{6}$. Since the motion is constant from frame to frame, there is no additional randomness of motion beyond the first frame. Thus, there is an equal chance of identical motion regardless of the number of frames in the spatio-temporal patch. The only benefit of more frames in the patch is that the asymptotic limit is reached faster with respect to SNR. Note that the curve for $T = 2$ is identical in these figures because for two frames in the spatio-temporal block, the two motion models are equivalent.

The variance functions for the two motion models and frame depths are plotted in Figs. 5.6b and 5.6e. The asymptotic variance decreases with frame depth for i.i.d. motion, but is the same for all frame depths for constant motion. This trend is due to the fact that $P_\emptyset$ decreases with number of frames for i.i.d. motion but is constant for constant motion.

It is also informative to view the probability of error as a function of number of frames in the spatio-temporal patch. These curves, for three SNR levels, are plotted in Figs. 5.6c and 5.6f. For the i.i.d. motion model, the probability of error continues to drop with number of frames, even at high SNRs. For the constant motion model, however, the error probability decreases with frame number only for low SNRs. For moderate and high SNRs, there is virtually no advantage in having more than 3 frames in the spatio-temporal patch.

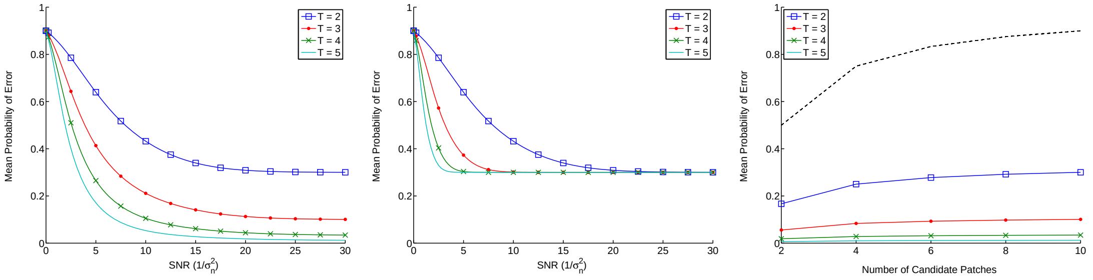
### 5.3.4 Discriminability for Many Candidate Patches

We can extend our analysis of probability of error of matching a reference patch to two candidates, one correct and one false, to $M$ candidates, 1 correct and $M-1$ false. Assuming all patches are independent, then the correct patch is chosen when the random variable $\mathcal{X}$, representing the error of the correct patch, is less than the random variable $\mathcal{Y}_i$, representing the error of the $i$th incorrect patch, for all $i$, $1 \le i \le M-1$. Thus, the probability of error is given by

$$\begin{aligned} P_e &= 1 - P_{\mathcal{X}<\mathcal{Y}_1,\ldots,\mathcal{X}<\mathcal{Y}_{M-1}} && (5.38)\\ &= 1 - \int_{-\infty}^{\infty}\int_{x}^{\infty}\cdots\int_{x}^{\infty} f_{\mathcal{X},\mathcal{Y}_1,\ldots,\mathcal{Y}_{M-1}}(x, y_1, \ldots, y_{M-1})\,\mathrm{d}y_{M-1}\cdots\,\mathrm{d}y_1\,\mathrm{d}x && (5.39)\\ &= 1 - \int_{-\infty}^{\infty} f_{\mathcal{X}}(x)\,[1 - F_{\mathcal{Y}}(x)]^{M-1}\,\mathrm{d}x && (5.40) \end{aligned}$$

where $f$ and $F$ are the probability density and cumulative distribution functions of their subscript variables, respectively.

Figs. 5.7a–5.7b shows the mean probability of error for $M = 10$ for the two motion models. The same trends are seen as discussed in Section 5.3.3 and



(a) (b) (c)

**Figure 5.7**: Mean probability of match error for multiple candidate patches. (a) As a function of SNR for i.i.d. motion, 10 candidate patches. (b) As a function of SNR for constant motion, 10 candidate patches. (c) As a function of number of spatially similar candidate patches. Dashed curve shows maximum error rate (same for all frames and motion models) and other curves show minimum error rate for the different number of frames.

shown in Figs. 5.6a and 5.6d. However, the error rate for the extreme SNR cases of 0 and $\infty$ are different. At an SNR of 0, the probability of error is 0.9, or more generally, $1 - \frac{1}{M}$, since the matching patch is effectively chosen at random.

Fig. 5.7c plots the extreme SNR error rates as a function of number of candidate patches. The dashed line shows the SNR $= 0$ case, which is common for all frames and motion models. As noted above, this line has equation $1 - \frac{1}{M}$. The other lines in the figure correspond to the SNR $= \infty$ case for the different number of frames in the patch for the i.i.d. motion model. The line representing $T = 2$ is also the curve for the constant motion model and is identical across $T$ for this model. As the number of spatially similar candidates grows, so does the range of error probabilities and thus the potential benefit of spatio-temporal matching. For scenes with periodic structures, large regions of similar texture, or high resolution video relative to the patch size, conditions that will all induce many spatially similar matches to a given patch, spatio-temporal matching can greatly reduce the ambiguity.

In comparing i.i.d. motion to constant motion in Sections 5.3.3–5.3.4, we find that i.i.d. motion is much more discriminable, except at low SNRs. This is because features become more displaced over the course of the sequence whereas for i.i.d. motion, the features could remain relatively static. The high noise level masks the small variations in motion. At higher SNRs, though, the discriminability of patches under the constant motion model is limited by the higher probability of two patches having identical motion. This is because the motion signature is fully determined by the motion on the first frame of the sequence. For i.i.d. motion, however, the number of possible trajectories a feature could follow grows exponentially with the number of frames, and hence its motion signature rapidly becomes harder to match. Thus, even though the error signal is much stronger for constant motion as derived in Section 5.3.2, the more varied motion of the i.i.d. model increases its discriminability.

### 5.3.5 Motion Vector Dependence

In Sections 5.3.1–5.3.2, we assumed that motion vectors are uncorrelated for non-corresponding points in the two stereo views, and Sections 5.3.3–5.3.4 further assumed that the motion vectors for the candidate patches were independent. This assumption may hold for many cases, but may not be true for patches from the same object or from objects with related motion. Here we discuss some of the implications of relaxing this assumption.

For correlated motion among $M$ spatially similar candidate patches, one of which is correct, the joint distribution function over $M-1$ variables describes the probability of motion of all the incorrect patch candidates. The error signal for the correct patch consists only of noise samples, and is thus always independent of the error signal for the incorrect patch. The distribution function will factor into $K$ potentially different functions over fewer variables, where $K$ is the number of independent motions among the $M-1$ patches. This factoring is a generalization of that given in (5.40), where the patch motions are independent and identically distributed everywhere.

In the simpler case of two candidate patches, one correct and one incorrect, the formulation of the error probability is the same for correlated motion as it is for uncorrelated motion as derived earlier, with a joint probability distribution replacing a single-variable probability distribution. What changes is the behavior at high SNRs governed by $P_{\emptyset}$, the probability of a zero error signal between the

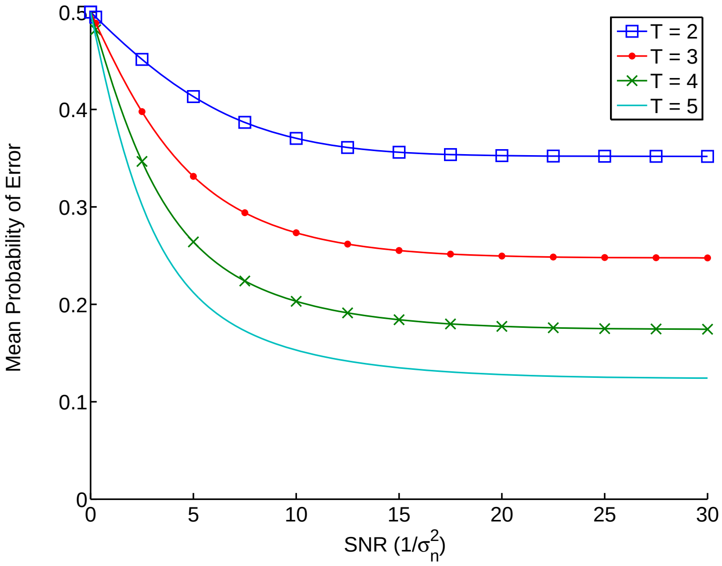


**Figure 5.8**: Mean probability of error for i.i.d. motion as a function of SNR for different number of frames for 2 candidate patches. The incorrect candidate patch has motion correlated with the reference patch.

reference and incorrect candidate patches. This probability depends on the joint distribution of motion vectors between the reference and candidate patches.

For instance, consider the case of linear dependence where $\boldsymbol{v}_i^R(\boldsymbol{p}) = c\boldsymbol{v}_i^L(\boldsymbol{p}+\boldsymbol{d}')$ for an incorrect disparity $\boldsymbol{d}'$, candidate patch motion vector $\boldsymbol{v}_i^L(\boldsymbol{p}+\boldsymbol{d}')$, reference patch motion vector $\boldsymbol{v}_i^R(\boldsymbol{p})$, and any real constant $c \neq 1$. Since the two motion vectors are equal only when they are the zero vector, the probability $P_\emptyset$ will decrease relative to the case when reference and candidate patch motion is independent, thus decreasing the probability of match error.

We demonstrate the other case, where $P_\emptyset$ increases relative to the independent motion scenario, using the same experimental setup as in Fig 5.5. We impose a correlation among the motion vectors such that the candidate patch generally moves in the same direction as the reference patch. Specifically, the incorrect candidate patch motion vector $\boldsymbol{v}_i^L(\boldsymbol{p}+\boldsymbol{d}')$ is equal to $\boldsymbol{v}_i^R(\boldsymbol{p})$ or one of its two immediate directional neighbors, with equal probability. For example, when $\boldsymbol{v}_i^R(\boldsymbol{p}) = \begin{bmatrix}1 & 1\end{bmatrix}^\top$, then $\boldsymbol{v}_i^L(\boldsymbol{p}+\boldsymbol{d}')$ is $\begin{bmatrix}0 & 1\end{bmatrix}^\top$, $\begin{bmatrix}1 & 1\end{bmatrix}^\top$, or $\begin{bmatrix}1 & 0\end{bmatrix}^\top$, each with probability $\frac{1}{3}$. In the case of no movement, $\boldsymbol{v}_i^L(\boldsymbol{p}+\boldsymbol{d}') = \boldsymbol{v}_i^R(\boldsymbol{p}) = \boldsymbol{0}$.

The mean probability of error for this joint distribution and the i.i.d. motion model is shown in Fig. 5.8. Compared to the independent motion case as shown in Fig. 5.6a, the error probability limits reached at high SNRs have greatly increased, thereby reducing the capacity for discriminability. The probability $P_\emptyset$ has increased because the chance of having equal motion vectors on a given frame is $\frac{1}{3}$ rather than $\frac{1}{9}$ as in the case of independent motion for this setup.

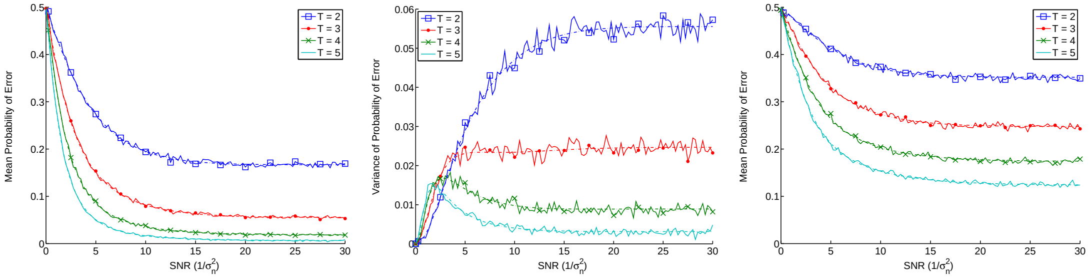
## 5.4 Experiments

We validate our derivations and observations in Section 5.3 in two ways. First, we simulate noisy patches, compute the error signals, and measure the probability of matching error directly. This experiment establishes direct statistical agreement between the theoretical results and simulation.

Second, we perform spatio-temporal stereo matching under different noise, motion, and frame depth conditions on six stereo video sequences with ground truth disparity. We measure the error and explain the trends in the context of the preceding analysis. Note that replicating the theoretical curves for error probability from real stereo video data is extremely difficult. For a given image feature, thousands of spatially similar patches would have to be found in the other stereo view whose motion is drawn from the same distribution as the reference patch. Further, without control or knowledge of actual object motion and features, this probability estimation would be confounded by motion estimation error and approximations of the motion distribution. Rather, this experiment highlights the intuition derived from the theory presented earlier.

### 5.4.1 Simulations

To verify the theoretical curves in Section 5.3, we simulate spatio-temporal matching using the same experimental setup as shown in Fig. 5.5. On each trial of the simulation, we construct 3 patches: a reference patch and a (false) candidate patch, each with independently generated random motion vectors according the the i.i.d. or constant motion model, and a second (correct) candidate patch identical to



(a) (b) (c)

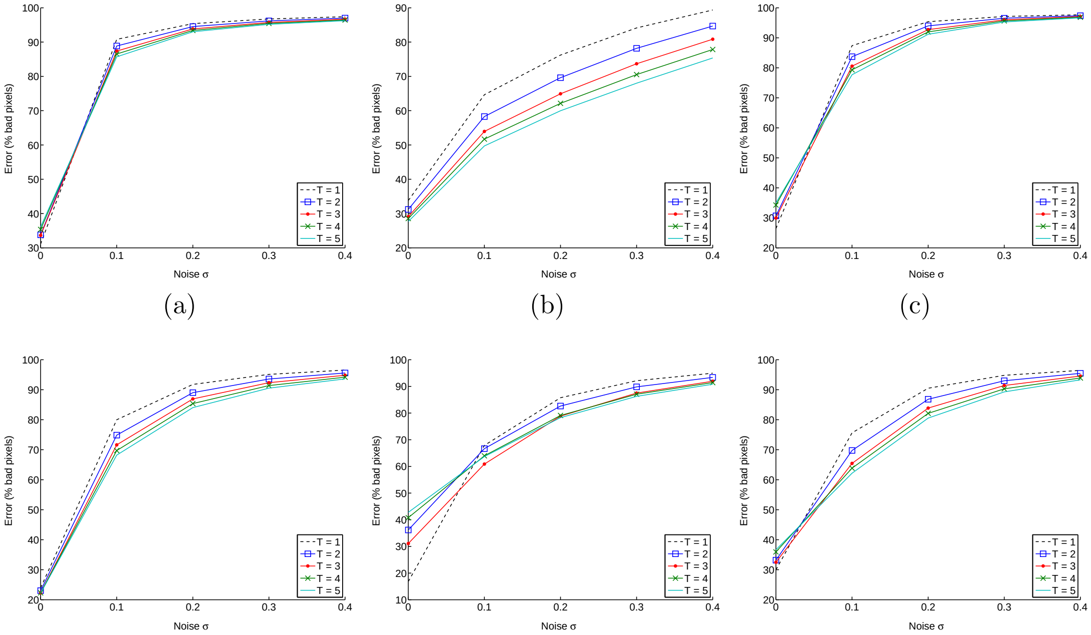
**Figure 5.9**: Simulations for probability of matching error for two candidate patches as a function of SNR for different number of frames for i.i.d. motion model. Simulated results shown as solid lines, theoretical curves overlayed as dashed-dotted lines for comparison. (a) Mean probability of error. (b) Variance of probability of error. (c) Mean probability of error for correlated motion between reference and incorrect candidate patches.

the reference patch. White Gaussian noise is independently added to each patch, and the reference-candidate SSD is computed for the two candidates. A match error is recorded if the correct candidate SSD score is greater than or equal to the SSD score of the incorrect candidate. We perform the same set of simulations for correlated motion, using the joint distribution given in Section 5.3.5. We computed the probability of error statistics over 10000 trials.

The results for the i.i.d. case are shown in Fig. 5.9. For both the independent and dependent motion scenarios, good agreement can be seen between the theoretical and simulated results. This simulation corroborates the theoretical results under the correct set of assumptions.

### 5.4.2 Stereo Video

We performed spatio-temporal block matching for 6 stereo video sequences with ground truth disparity: *Books*, *Street*, *Tanks*, *Temple*, and *Tunnel* from [93], and a sequence we refer to as *UNL* (from the University of Nebraska, Lincoln) from [94]. All of these sequences are noiseless and have either local motion or global camera motion. The first 5 videos have resolution 400 × 300 pixels and length 100 frames, except for *Books* which is 41 frames long. *UNL* is 640 × 480



(a) (b) (c)

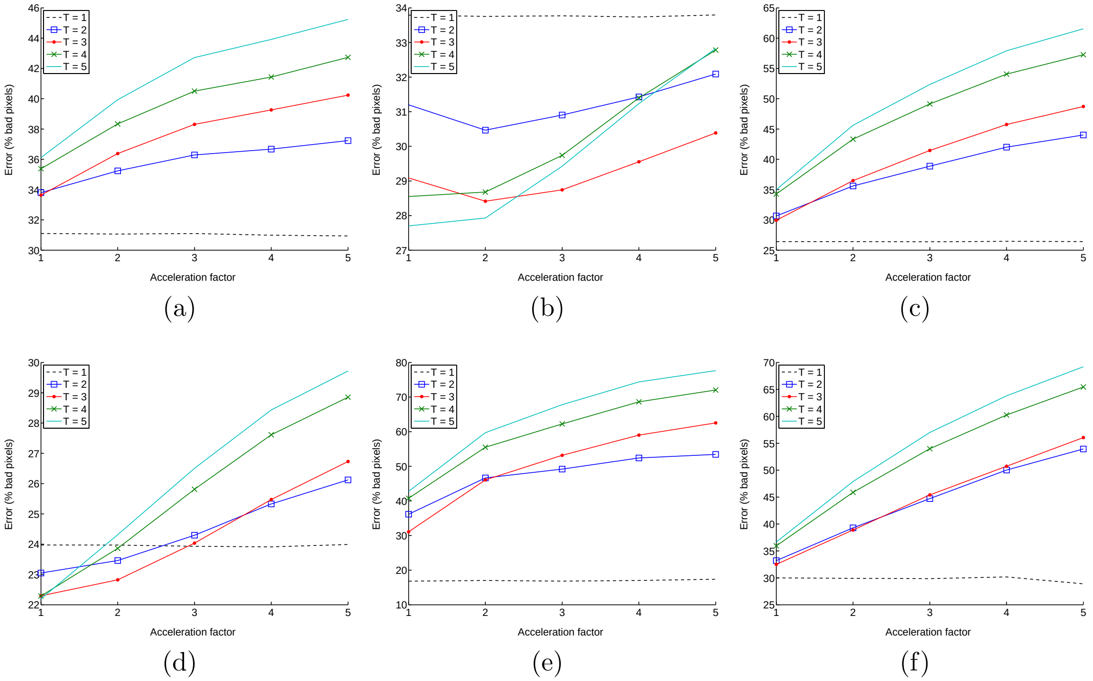
(d) (e) (f)

**Figure 5.10**: Disparity error versus noise level. (a) *Book* (b) *Street* (c) *Tanks* (d) *Temple* (e) *Tunnel* (f) *UNL*.

pixels and 120 frames long.

Consistent with our formulation in Section 5.3, we estimated the right view (integer) disparity map by seeking the best matching patch in the left view to a reference patch in the right view using the SSD error metric. A spatial block size of 7 × 7 was always used, and matches were only sought to the right of the reference patch (i.e. only nonnegative disparities were permitted) due to the geometric constraints of rectified stereo.

We conducted two experiments in which we pre-processed the left and right color videos to affect patch discriminability. For each of the experiments, stereo matching was performed after the processing for a block length of 2–5 frames, and is also compared to frame-by-frame (i.e. block length of 1 frame) disparity estimates. Results are based on the percentage of incorrect disparity estimates using the ground truth disparity as a reference, and excluded the monocular strip of the right view and border pixels/frames where cost aggregation did not have full support.



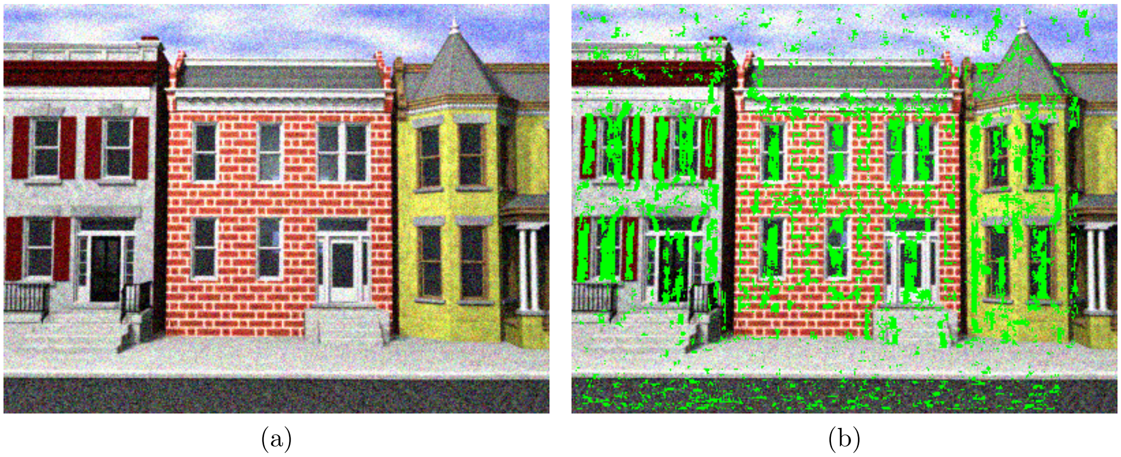
(a) (b) (c)

(d) (e) (f)

**Figure 5.11**: Disparity error versus speed level. (a) *Book* (b) *Street* (c) *Tanks* (d) *Temple* (e) *Tunnel* (f) *UNL*.

In the first experiment, we added different levels of zero-mean white Gaussian noise to the videos. With a normalized image dynamic range of 0 to 1, the additive noise had standard deviation $\sigma = \{0.1, 0.2, 0.3, 0.4\}$, and the noiseless case of $\sigma = 0$ was also tested for reference.

Results for the noise experiment are plotted in Fig. 5.10. For all of the sequences, more frames in the spatio-temporal block generally lowers the percentage of bad pixels. The exception is the noiseless case where patch discrimination is already excellent, so adding frames to the block increases the chance of aggregating costs across disparity levels rather than increasing discriminability. This is most notable in the *Tunnel* sequence, which has a lot of texture and radial motion. The benefit of adding frames to the spatio-temporal patch diminishes at the higher noise levels, nullifying the potential for gains from additional discriminability. Spatio-temporal block matching is also most effective for the *Street* video. Even for high noise levels, disparity estimates for this sequence continue to improve with increased frame depth.



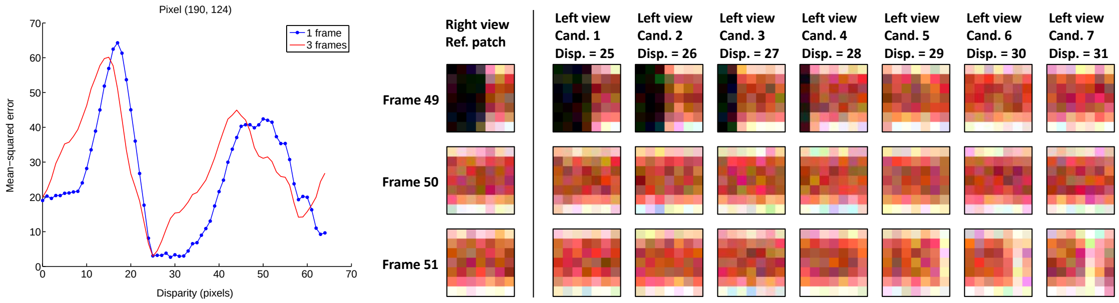
(a) (b)

**Figure 5.12**: (a) Frame 50 of *Street* sequence with Gaussian noise of standard deviation 0.1. (b) Same frame with green pixels indicating locations where spatio-temporal stereo matching with 3 frames resulted in correct disparity estimates versus frame-by-frame matching.

In the second experiment, the noiseless videos were temporally subsampled at different rates, termed the acceleration factor, in order to simulate faster motion distributions. Acceleration factors of 1 (no acceleration) to 5 were tested.

The results for the speed experiment are given in Fig. 5.11. For *Book*, *Tanks*, *Tunnel*, and *UNL*, error increases with acceleration factor and with number of frames in the patch. Further, the error is uniformly lower for frame-by-frame matching. The *Temple* sequence follows a similar trend, but only for the higher acceleration factors does frame-by-frame matching outperform spatio-temporal matching. For the lower acceleration factors, a frame depth of 3 is optimal. For *Street*, spatio-temporal block matching is uniformly better than frame-by-frame matching. At normal speed, more frames in the block is better, but at higher speeds, 3 frames in the block is optimal. These behavioral differences with speed underscore the importance of considering the motion distribution in selecting parameters for spatio-temporal disparity estimation.

Spatio-temporal stereo matching is particularly beneficial to the *Street* sequence. This performance is due to the largely planar objects in the video. While the camera and objects are not always relatively parallel, the motion still allows for sufficient discrimination among the patches.



(a) (b)

**Figure 5.13**: Match ambiguity for (row, column) pixel (190,124) of frame 50 resolved by motion of neighboring frames. (a) Cost functions for using one frame (frame 50) and three frames (frames 49–51) for matching. (b) Candidates in left view that are all spatially similar to the reference patch of frame 50 in the right view.

Fig. 5.12 shows a frame from the *Street* sequence tagged with locations where 3-frame spatio-temporal stereo matching resulted in correct disparity estimates versus frame-by-frame matching. For this frame, 15.7% of the pixel disparity estimates were corrected by using spatio-temporal matching. Most of the improvement is within regions of flat texture near strong edges. By using neighboring frames, the correspondence problem for textureless regions is resolved since motion brings the nearby edge into the spatio-temporal window.

This observation is demonstrated in Fig. 5.13. The mean SSD cost for one particular patch from the frame in Fig. 5.12 using 1-frame and 3-frame stereo matching is plotted in Fig. 5.13a as a function of possible disparity values. The estimated disparity will be the one that minimizes its cost in Fig. 5.13a. Note that there are several disparity values for 1-frame matching that have approximately equal minimal costs, resulting in a decision based on noise. The disparity value of 29 happens to be the minimizing disparity, and is incorrect. For 3-frame matching, however, there is a single clear minimum at a disparity of 25, the correct disparity.

The reference patch from the right view and several candidate patches from the left view are shown in Fig. 5.13b. The seven candidate patches are all spatially similar to the reference patch on frame 50, so block match disparity estimation based on this frame alone is clearly ill-posed. By including the neighboring frames in the block, the strong edge helps distinguish the candidates: only Candidate 1 matches the reference patch on all three frames, and the correct disparity is selected.

## 5.5 Discussion

Motion is a strong cue for distinguishing objects, and thus can be very useful for solving the stereo correspondence problem. The human visual system can use temporal information to resolve stereo ambiguities [107], and similarly, the motion cue is useful in estimating disparity for stereo video. Our results show how effective spatio-temporal block matching is in disambiguating stereo matches as a function of number of frames, image noise, patch features, and motion distribution. We have sought to answer the question of how beneficial the motion cue is to stereo matching in a quantifiable manner.

Though the answer is interesting in its own right, our results can be used in several ways. Firstly, the results in Section 5.2 provides guidelines as to when the basic assumptions of constant disparity and identical camera motion are valid. Based on the camera and scene parameters, one can determine whether the simple translational model of disparity and motion is sufficient or if something more sophisticated needs to be employed. Only for very fast or very close up radial motion should the latter be necessary.

Secondly, the plots and equations describing probability of error offer much insight into the types of scenes and motion that lend themselves to spatio-temporal matching. The analysis of match error probability as a function of number of spatially similar candidates also shows that high resolution scenes or those with periodic or homogeneous areas, where there is more chance of having many false matches, will benefit more from spatio-temporal matching. Scenes with high motion relative to the noise will benefit from the motion cue as long as the motion is varied. More frames can be used in the spatio-temporal block in either the slow motion or high noise case. The type of motion matters as well. For scenes with random motion, such as fluttering leaves or turbulent water, much better discrimination can be achieved. Motion that is approximately constant over a small group of frames, such as a rolling ball or a person walking, offers less potential for patch discriminability. Applications where the motion model is explicitly known or can be estimated, such as for active learning robots [108], image capture during zooming [109], or the stereo case itself [110], can insert the motion model into our expressions to compute the discriminability.

Finally, our results can be used to tune spatio-temporal matching algorithms or select parameters. Previously, the number of frames used in block matching was chosen heuristically or empirically [72, 91, 111]. Using our results and some knowledge of the video parameters such as noise and motion, the number of frames can be chosen to meet a certain discriminability threshold. The video parameters could be estimated from the data or may be available from some other processing schemes: motion vectors might be available as a side product of compression or noise variance might be estimated as part of a denoising routine. Further, the number of frames could be adjusted based on characteristics of the patch. If the patch has very weak features, more frames could be stacked up for matching. Similarly, the level of motion could be detected by simple frame differencing followed by a threshold. If there is no detected motion, the previous frame's disparity for that block could be propagated forward. If there is motion, then spatial or spatio-temporal block matching can be applied. At mid-range to high SNRs for constant motion, having 3 frames in the spatio-temporal window is the most efficient. In this way, the required number of computations can be reduced compared to spatio-temporal matching across the entire video.

While in this chapter we have focused on the ability of motion to disambiguate potential matches, there are other issues with spatio-temporal matching to consider. Assuming matching blocks are sought from corresponding points in time, then spatio-temporal matching will require roughly $T$ times as many computations versus spatial matching, where $T$ is the number of frames in the spatio-temporal block. However, this computational burden may be somewhat offset if the SSD minimization procedure is to be followed by a global optimization method. This is often the case, where the error of the best block match is used as the data cost in an MRF model [112]. Due to the reduced number of false matches, fewer candidate patches need to be considered in global disparity refinement methods, which can lower the memory requirement and reduce the number of computations necessary to yield a satisfactory result. Iterative algorithms such as Belief Propagation [112] would need fewer iterations to reach a stopping criterion.

The overall mean-squared error of a disparity estimator will also change for spatio-temporal versus spatial matching. The results in [99] can be extended to the spatio-temporal stereo case, by which it can be shown that the Cramér-Rao lower bound on the variance of a horizontal disparity estimator is lower for spatio-temporal block matching than for spatial matching. This is because there are more samples in the estimation window. However, the bias of this estimation problem may also increase due to adhesion noise. This type of noise occurs near depth discontinuities in occlusion areas, where the feature may be assigned a depth of a nearby object. This results in an elongation of objects in disparity space, which is well-explored in spatial matching [113], but less so in spatio-temporal matching.

## 5.6 Conclusion

We have studied the role of motion in stereo matching, and quantified its efficacy in disambiguating potential matches in terms of probability of match error. We began by using arguments from stereo geometry to determine whether the intrinsic assumptions of spatio-temporal matching are valid for a given scene and imaging system. Previous work has either used the constant disparity assumption [91, 102] or something more complex [71, 86], without justification. Our analysis provides a way to properly judge the assumption, after which a spatio-temporal stereo matching algorithm appropriate for a given imaging system can be designed.

Our main contribution was to analyze the spatio-temporal block matching error signal, and derive the probability of a false match as a function of number of frames, motion variance, image features, and noise. While previous works have noted the benefit of spatio-temporal matching in match discrimination [71,72,86,110,111], ours is the first to study the extent and nature of using motion to improve match accuracy. Our expressions can account for much of the intuition and heuristic knowledge behind spatio-temporal stereo matching. We demonstrated this intuition by explaining experimental results of spatio-temporal stereo matching applied to six stereo videos. Further, as discussed in Section 5.5, this work can be used to select parameters or adjust matching algorithms based on scene characteristics.

This line of research can be continued in several ways. Our analysis could be extended to other error measures for spatio-temporal stereo matching. Though this was somewhat attempted in [72], the study was not very extensive or theoretical. Similarly, an analysis of discriminability for global disparity estimation methods could be conducted. Together, these analyses of local and global methods would be useful in understanding the most efficient and effective ways of processing stereo video. Convergent stereo systems could also be studied, which would have bearing on how to best capture scenes for depth determination.

In the interest of reproducibility and to support future research, all of our code is available at http://videoprocessing.ucsd.edu/~ankitkj/research/.

## Acknowledgments

We thank anonymous reviewers for *IEEE Transactions on Image Processing* whose comments have improved the organization and completeness of this chapter.

The text of Chapter 5 is, in part, a reprint of a paper that has been published as: A. K. Jain and T. Q. Nguyen, “Discriminability limits in spatio-temporal stereo block matching,” *IEEE Trans. Image Process.*, vol. 23, no. 5, pp. 2328-2342, May 2014. The dissertation author was the primary investigator and author of this paper.

# Chapter 6

# Anisotropic Spatial Integration in the Sensing of Horizontal Disparity

Thus far, we have considered mixed resolution schemes employing either a radially symmetric filter (as in Chapter 2) or one that is symmetric in the cardinal directions (as in Chapter 4). However, due to the lateral separation of the eyes, there is a natural preference for vertical versus horizontal contours in stereopsis. This anisotropy suggests that a stereo pair downsampled with a vertically elongated kernel could deliver a greater impression of depth relative to a stereo pair that is isotropically downsampled. Such a design would allow greater bandwidth reduction for a given depth response. Thus, since the aim of mixed resolution coding is to find an efficient representation of stereo content, we must also consider the method by which downsampling is performed.

In this chapter, we investigate this anisotropy in stereopsis. Our study is useful for efficient compression of stereo content, design of metrics for depth saliency, and determining processing mechanisms within the visual system. The stereo anisotropy in the visual system suggests that, in order to be optimally efficient, specialized mechanisms for processing horizontal disparity should exist.

Indeed, such biological evidence has been found in mammals. In [114], it was shown that certain neurons in the cat's visual cortex are specialized for disparity sensitivity, and later showed that cells tuned to near vertical orientations modulate a much wider range of binocular phases [115, 116]. This evidence suggested a disparity encoding model based on phase rather than position, and also pointed to a specialized processing scheme for horizontal disparity and the vertical contours through which it is most efficiently transmitted. The authors in [117] measured the neuronal response of monkeys to a wide range of directional disparities, and found that responses were strongest to horizontal disparities, even for receptive fields tuned to other monocular orientations.

Psychophysical studies have also been conducted to investigate anisotropies in disparity detection and processing. Using random dot stereograms, it has been found that the visual system is more sensitive to horizontal depth modulations than vertical ones [118–121]. The authors in [120] show that this increased sensitivity is due to the visual system integrating in depth over a larger horizontal than vertical excursion, and posit that this anisotropy exists to compensate for the inefficient transmission of horizontal disparity by predominantly horizontal luminance contours. In [119], a similar mechanism is suggested for the finding that at supra-threshold disparities, cyclopean edges at the oblique orientation appear to have more depth than those at cardinal orientations. We discuss these results as they relate to the luminance anisotropy investigated in this chapter.

In the luminance domain, log stereoacuity thresholds measured from thin bars are degraded roughly as the sine of the orientation angle measured from horizontal [122]. In [123], subjects identified which of two intervals contained a binocular correlation among random line stereograms. Correlation detection rates were twice as great for vertically versus horizontally oriented lines. Using filtered random dot stimuli, the study in [124] showed that stereopsis does not exclusively use vertical contours, however, and that oblique contours carry significant disparity information. In those experiments, stimuli with oriented spatial frequencies were generated by simulating astigmatic blur in the oblique and cardinal directions. Disparity thresholds were measured for two configurations: each eye receives the same orientation of stimuli, or each eye receives an orientation perpendicular to the other eye's stimulus. In this manner, spatial frequency components could be selectively mismatched in order to isolate their contribution to stereo matching.

Here, we choose the more direct approach of obtaining contrast sensitivity functions for detecting depth as a function of spatial frequency and orientation using bandpass filtered random noise patterns. We compare three texture orientations as well as two isotropic spatial frequency distributions. Further, we develop a model based on the power spectrum of the stimuli and the phase model for disparity encoding [114]. Our model reveals how luminance contours with different spatial frequencies and orientations are used in stereopsis, and adequately explains our data. Though an anisotropy favoring vertical contours was found in [125] using a similar procedure and parameters, our experiments are more extensive, consider stereopsis models underlying the data, and show that the relationship between sensitivity for depth detection and orientation is more complex than suggested by their results.

## 6.1 Methods

### 6.1.1 Stimuli

In total, 5 different bandpass filters were tested: 3 anisotropic filters with circular passbands at different orientations and 2 isotropic filters with annular passbands. The extent of the circular passbands was chosen such that they lie tangent to one another in frequency space. Thus, they have constant log bandwidth and radius

$$r = f_c \sin 22.5^\circ, \tag{6.1}$$

where $f_c$ is the center frequency. The three texture orientations tested were 0°, 45°, and 90° (horizontal, oblique, and vertical, respectively). Throughout, we refer to the filters by these spatial domain orientations.

Two isotropic filtering conditions were tested. In one condition, the width of the annular passband was set equal to that of the circular passbands in the oriented filter conditions. In the second condition, the bandwidth was chosen so that the number of Fourier components was equal to the oriented filter conditions.

That is, the outer and inner radii defining the annulus are equal to the center frequency plus or minus, respectively, the quantity

$$r_a = \frac{r^2}{2f_c} = \tfrac{1}{2} f_c \sin^2 22.5^\circ. \tag{6.2}$$

Thus, the anisotropic and the larger isotropic filters have passband widths of 1.16 octaves, and the equal-area isotropic filter has a passband width of 0.21 octaves. We refer to the larger bandwidth isotropic filter as the iso+ orientation. Similarly, the filter with an equal number of Fourier components as the anisoptropic filters is denoted as the iso= orientation.

The spectra of the five filters had a value of 1 within their passbands and 0 elsewhere. A uniformly distributed binary random signal was filtered digitally in the Fourier domain with the bandpass filters to produce the stimulus textures. The textures were all normalized to have the same RMS contrast and mean gray value. A schematic of the bandpass filter spectra are shown in Fig. 6.1. Examples of the spatial textures produce for a center frequency of 7 cpd can be seen in Fig. 6.2.

Spatial center frequencies of 2, 7, and 12 cpd were tested at a disparity of 3.72 arcmin. In pilot work, we found the 12 cpd, 3.72 arcmin disparity parameter combination to be very difficult at the oblique, vertical, and equal spectral area isotropic orientations. Thus, we left out these three conditions, and added a 1.24 arcmin disparity condition for all filters at the 12 cpd center frequency. These parameters comprised a set of 17 unique test conditions.

### 6.1.2 Apparatus and Procedure

Stimuli were presented on a 22” LaCie electron22blueIV Diamondtron CRT driven by an NVIDIA GeForce GT 545 video card running at 1152 × 864 pixel resolution with a refresh rate of 75 Hz. The experiment was controlled using Matlab running the Psychophysics Toolbox, version 3 [35–37] on a Windows XP computer. The images were presented dichoptically using a mirror stereoscope. A chinrest was used to maintain a constant viewing distance of 1.96 m, at which the perceived resolution was 96 pixels/degree visual angle.

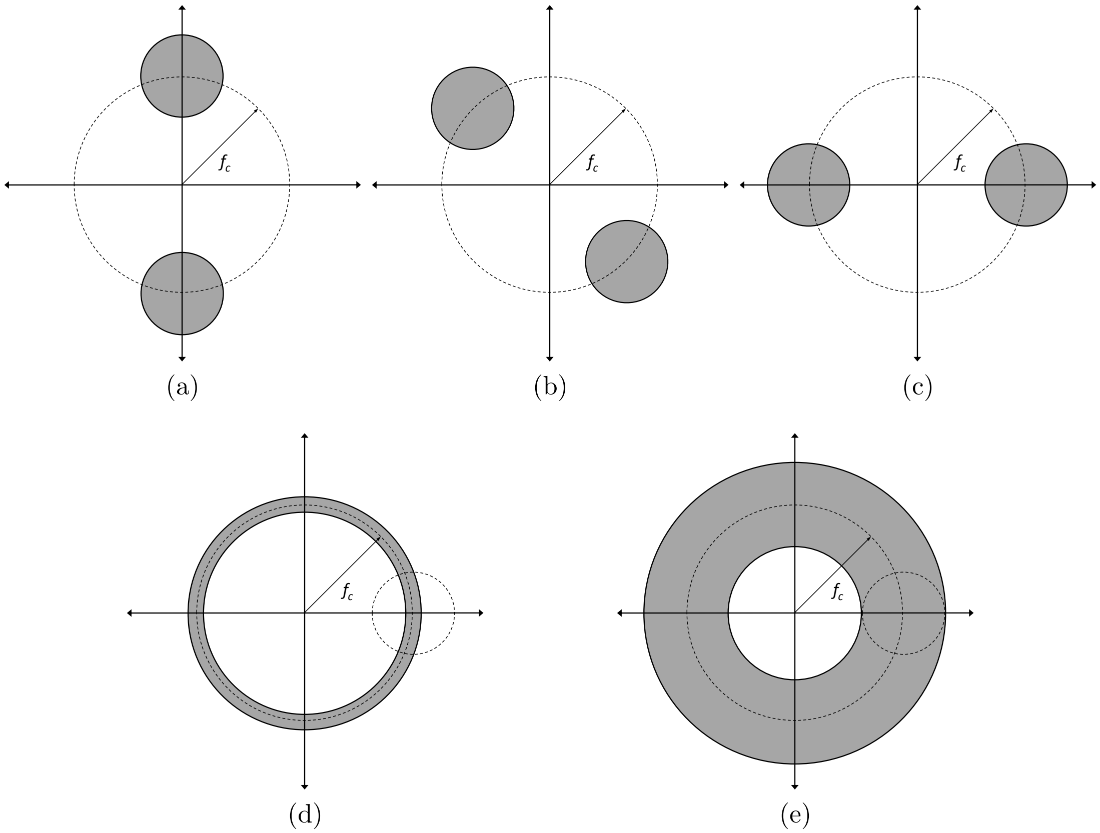


**Figure 6.1**: Scale diagrams of filter spectra. Gray and white regions indicate values of 0 and 1, respectively. Circle with radius equal to the center frequency $f_c$ is shown for reference. For the two isotropic conditions in (d) and (e), one circular passband of the anisotropic condition is shown for reference. (a) Horizontal (0°) (b) Oblique (45°) (c) Vertical (90°) (d) Isotropic, equal spectral area (iso=) (e) Isotropic, equal bandpass radius (iso+).

On each trial, two patches with a given horizontal disparity between them were randomly sampled from a stimulus pattern. The patches were displayed stereoscopically in two configurations side by side, one with crossed and one with uncrossed disparity. Thus, the two stereoscopic patches always equal and opposite disparity, and identical contrast. The patches were displayed in a circular aperture of diameter 1.75°. A Gaussian contrast taper was applied to the patches as well, with standard deviation 0.35°. The contrast taper was always centered within the aperture, which both always remained at zero disparity. Surrounding the patches was a random texture presented at zero disparity, with 20% density each of light

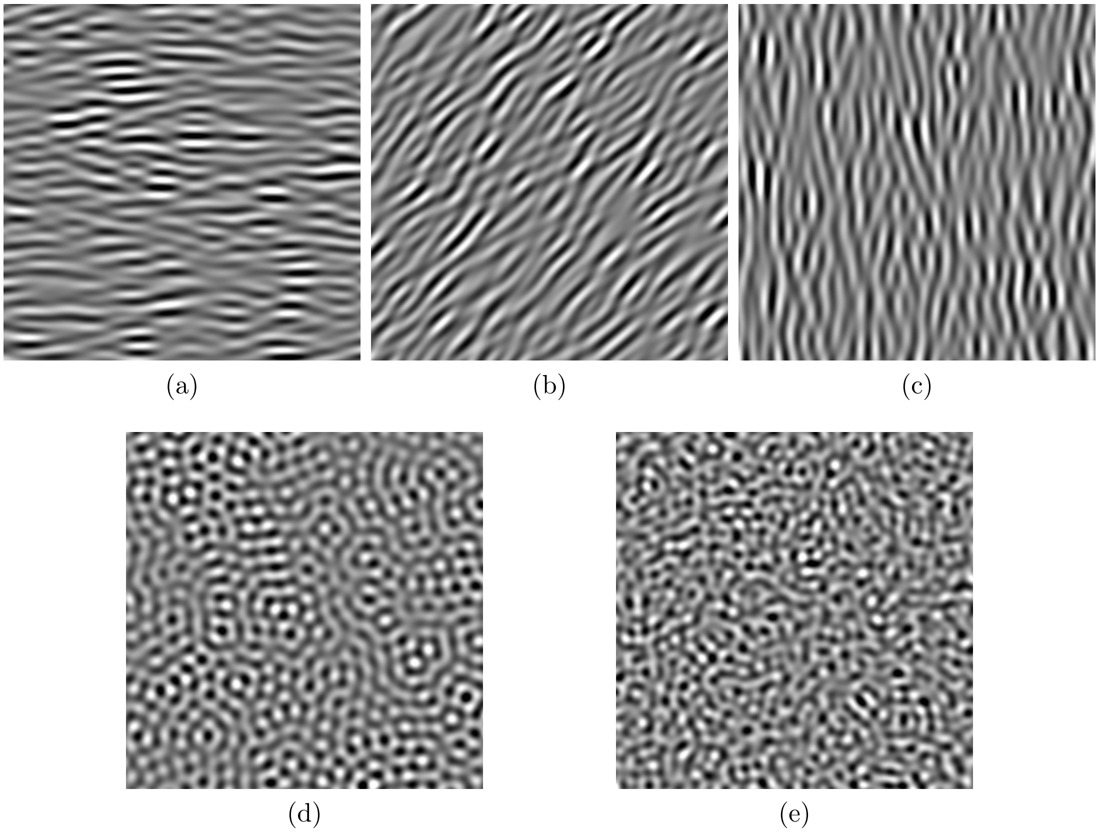


**Figure 6.2**: Example stimuli for the 7 cpd condition. (a) Horizontal (0°) (b) Oblique (45°) (c) Vertical (90°) (d) Isotropic, equal spectral area (iso=) (e) Isotropic, equal bandpass radius (iso+).

and dark dots, and 60% gray dots. This background aids fusion and provides a reference depth level against which to compare the disparate patches. A fixation cross and nonius lines centered between the two stereoscopic patches were visible throughout the experiment and were present to check proper vergence.

The subject's task was a two-alternative forced choice to identify which stereoscopic patch was farther from them (had the uncrossed disparity). The patch with the uncrossed disparity randomly varied between trials as appearing on the left or the right. To begin a trial, subjects pressed a button on a keypad and after 250 ms, the stimuli appeared for 1000 ms and then disappeared. Subjects were then prompted to respond "left" or "right", identifying the patch behind the screen, by pressing the corresponding button on the keypad. While the stimuli were not being displayed, a flat gray region filled the circular apertures in their place.

Though subjects were asked to fixate on the central cross at the beginning of each trial, they were permitted to look around the screen for the duration of the stimulus display. For this task, vergence is not important since verging on one of two patches with equal and opposite disparity still preserves the relative disparity. Similar methodology, stimuli, and tasks were employed in [126, 127]. Though the authors in [127] control for vergence, they found that there results generalized to 2-second exposures.

Prior to the actual experiment, subjects were given a training session with full contrast stimuli, longer display times, and auditory feedback for incorrect responses. As subjects progressed, the display time was shortened to 1000 ms and the auditory feedback was removed, just as in the real experiment.

Each of the 17 test condition was run in a block. The contrast of the patches was adjusted in log steps using an adaptive staircase procedure over 50 trials targeting an 85% correct response rate. Two such staircases were randomly interleaved in each block, resulting in 100 trials for each test condition. Each subject completed two runs of the experiment.

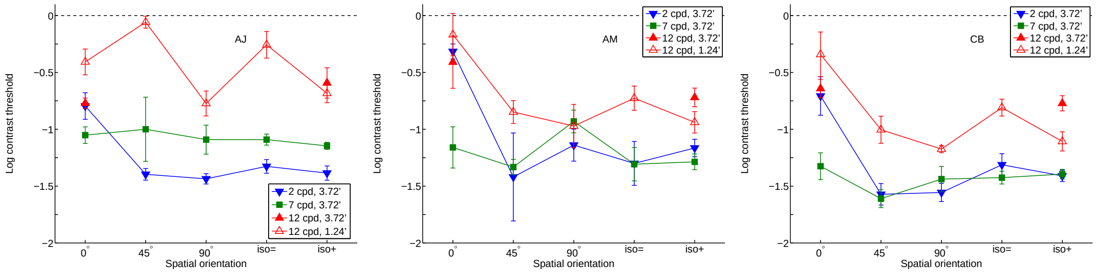
Contrast thresholds for disparity detection were determined by fitting a cumulative Gaussian to the trial responses with a detection threshold of 75%.

### 6.1.3 Subjects

Three subjects, one author and two others naive to the purpose of the experiment, participated. All had normal or corrected visual acuity and good stereo vision.

## 6.2 Results

Results for the three subjects are shown in Fig. 6.3. At 2 cpd, contrast sensitivity for extracting depth is clearly impaired for the $0^\circ$ degree case. The oblique and vertically oriented patterns yield high sensitivity and no improvement



(a) (b) (c)

**Figure 6.3**: Results of the experiment for the three subjects. Mean thresholds and standard errors are shown. (a) Subject AJ. (b) Subject AM. (c) Subject CB.

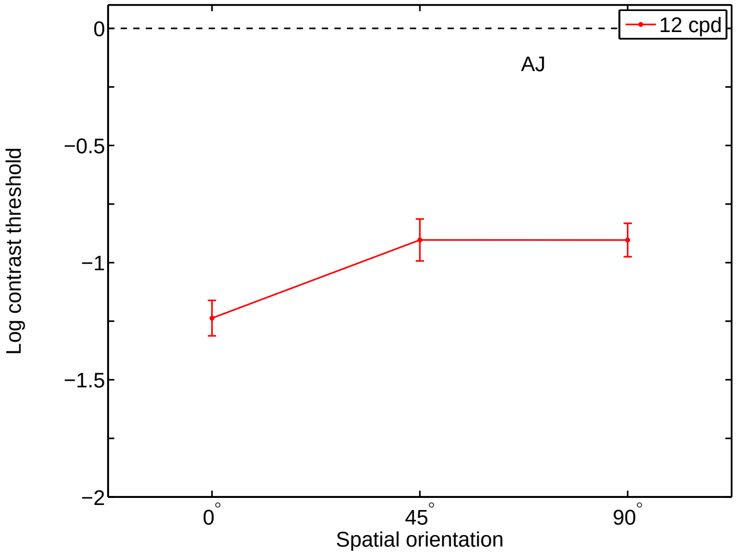


**Figure 6.4**: Detection thresholds for the 12 cpd condition for subject AJ.

is attained by adding Fourier components isotropically.

At 7 cpd, there is relatively little variation in sensitivity between the horizontal, oblique, and vertical orientations. The two isotropic orientations show the same sensitivity across subjects.

Thresholds could not be obtained for 12 cpd, 3.72 arcmin disparity condition for oblique, vertical, and iso= conditions. This may be due to the problem of identifying corresponding features in the two eyes images, a problem exacerbated by the higher density of local features. This center frequency and disparity combination produce a binocular phase of nearly 270°, resulting in a very ambiguous stimulus that is susceptible to the wallpaper illusion. Interestingly, a threshold could be obtained for the horizontal orientation where the horizontally extended contours provide less scope for the wallpaper illusion but there are still sufficient vertical features to support stereo matching. Allocating the frequency components isotropically still proved to be too ambiguous, but adding more components did permit reliable depth detection at the highest contrasts. However, the addition of Fourier components did not greatly improve sensitivity for the horizontal orientation. For all subjects, the horizontal and iso+ conditions resulted in nearly equal thresholds.

The difficulties in extracting depth at the high spatial frequency, high disparity condition led us to test the 12 cpd center frequency with a disparity of 1.24 arcmin, corresponding to a binocular phase of 90°. This quadrature phase is optimal in the context of the phase encoding model for disparity [114, 127]. An anisotropy between horizontal and vertical orientations is again present. Further, the isotropic spectral distribution gives lower sensitivity than the vertically striated pattern, but adding more Fourier components in the isotropic case restores sensitivity to the level achieved by the vertically oriented stimulus alone. The elevated threshold for the oblique orientation for AJ is partially due to decreased sensitivity for oblique orientations, as shown in Fig. 6.4. Though AJ has equal sensitivity to obliques and verticals, the vertical features are rich enough in the 90° condition to allow for depth detection at lower contrasts. Note that in the horizontal condition all subjects detected the 1.24' disparity with lower sensitivity than the 3.72' condition.

The major trends in the data are intuitive. The increased sensitivity for vertical orientations is not unexpected given that these contours most efficiently transmit horizontal disparity information. Similarly, the more elevated thresholds for the iso= condition than for the vertical orientation may reflect the fact that for a fixed spectral area, the isotropic distribution results in an inefficient allocation with respect to disparity extraction. However, other features of the data are less readily understood. Why would the 12 cpd, 1.24 arcmin condition have a higher threshold than the 12 cpd, 3.72 arcmin condition for the horizontal orientation?

The former is the supposedly optimal (quadrature phase) pairing for disparity extraction. Why is the horizontal-oblique threshold difference smaller for 2 cpd than for 7 cpd? Why do the vertical and two isotropic cases have similar thresholds for 7 cpd? We propose a model for how spectral information is used to extract disparity that can explain these phenomena.

Consider a two-dimensional luminance pattern $f(x, y)$ with (continuous) power spectrum $S_f(\xi_1, \xi_2)$. To find the total signal power of the vertical contours (in the luminance domain), we project the power spectrum onto the $x$-axis in the frequency domain and compute the sum:

$$S_f^v(\xi_1) = \int_{-\infty}^{\infty} S_f(\xi_1, \xi_2)\,\mathrm{d}\xi_2. \tag{6.3}$$

The resulting $S_f^v(\xi_1)$ is the power spectrum for vertical contours. However, not all frequencies are weighted equally in disparity computation. From the phase model for disparity encoding and as shown in [127], the visual system is most sensitive to spatial frequencies carrying a binocular phase of 90° with respect to the disparity being recovered. The binocular phase $\theta_b$ is a function of spatial frequency $\xi_1$ and disparity $d$. For a given disparity, it can be calculated for all frequencies in the spectrum. Since sensitivity depends on the binocular phase and peaks at 90°, we adopt a sinusoidal weighting function:

$$w_p(\xi_1, d) = \begin{cases} \left|\sin\left(\theta_b(\xi_1, d)\right)\right|, & |\theta_b(\xi_1, d)| < 180^\circ. \\ 0, & \text{otherwise.} \end{cases} \tag{6.4}$$

The absolute value is to ensure that energy in the negative half of the frequency domain is not inverted. Phases within half a cycle are considered to contribute to stereo matching whereas more ambiguous phases are discarded by the weighting.

For a given disparity, the weighting function $w_p$ is then applied to $S_f^v(\xi_1)$. The result is the vertical contour power spectrum for a certain disparity. The total disparity energy is then computed as the sum of this function:

$$E(d) = \int_{-\infty}^{\infty} w_p(\xi_1, d) S_f^v(\xi_1)\,\mathrm{d}\xi_1. \tag{6.5}$$

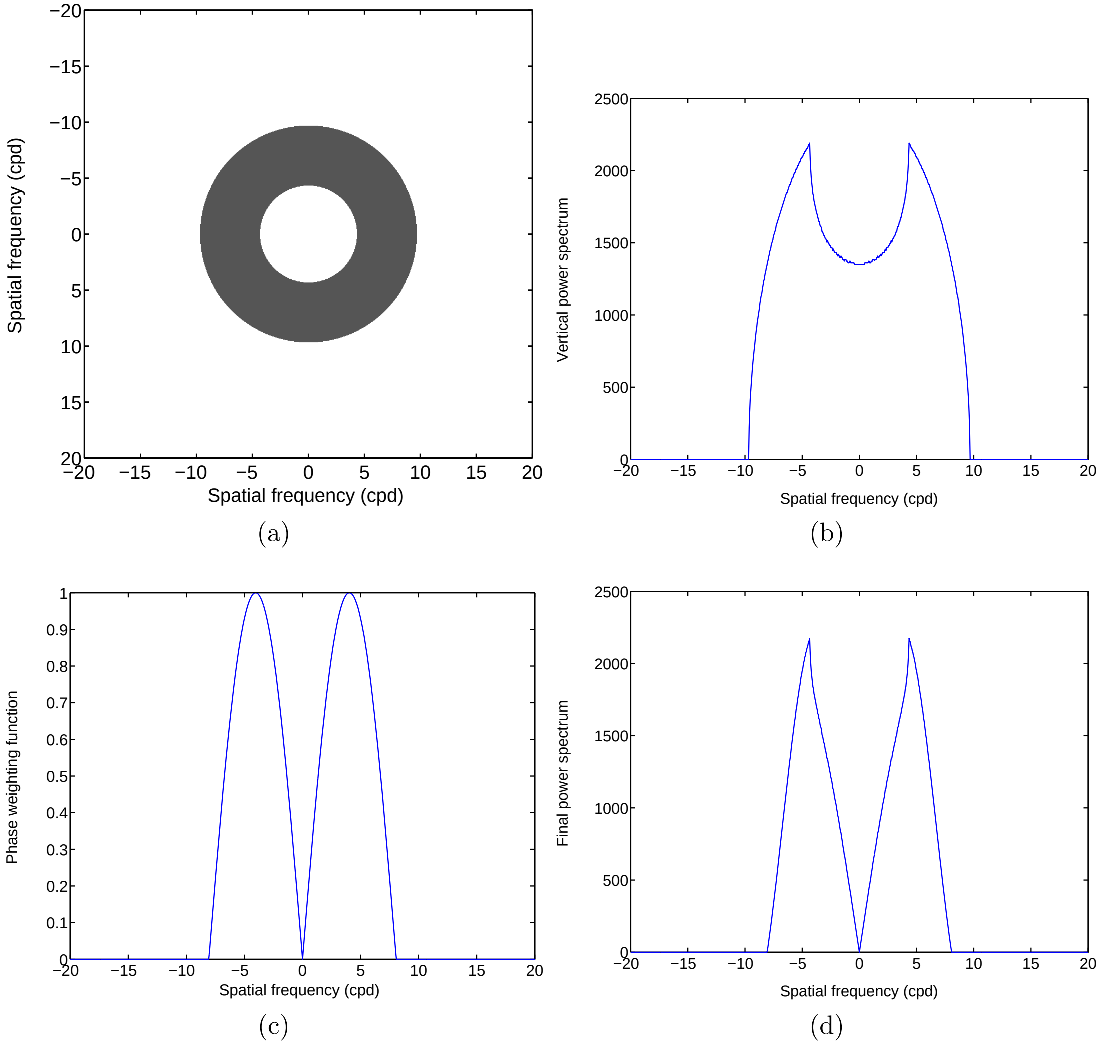
Since the stimuli in our experiment were produced by filtering white noise, the power spectra are given by

$$S_f(\xi_1, \xi_2) = P_0 \left| H(\xi_1, \xi_2) \right|^2, \tag{6.6}$$

where $P_0$ is the input power and $H$ is the filter transfer function. This expression is further simplified due to the fact that our filters are zero-phase, binary masks. Using our filters and Equation (6.5), we computed the disparity energy for the 3 spatial frequencies, 2 disparities, and 5 filter orientations used in the experiment. The step-by-step process of computing the model is shown in Fig. 6.5. We plotted the negative log of the energy in Fig. 6.6, which gives an indication of the relative thresholds modeled by our equations.

First note that the predicted thresholds are relative within each spatial frequency curve since our model does not account for the frequency dependence of contrast sensitivity that may originate from sources unrelated to disparity processing (such as optical contrast losses at high spatial frequency). Thus, each curve may be shifted up or down relative to the others, and the actual negative log energy value does not directly correspond to the real contrast threshold. Second, the predicted results capture many aspects of the measured data, including the previously mentioned subtler phenomena. For the 12 cpd, 3.72 arcmin disparity condition, the model threshold for the horizontal orientation is lowest and is high for the other orientations, explaining why we could not obtain the other thresholds. It erroneously predicts a higher threshold for the iso+ orientation, which should be at about the level of the horizontal orientation threshold. This error probably results from the fact that we have not included contrast sensitivity in the model, which will affect the higher spatial frequencies more, especially because of the larger bandwidths at these frequencies. This deficit is also seen for the same parameter combination for 1.24 arcmin—the actual data show a lower threshold for iso+ versus iso=.

Figs. 6.7–6.8 show how the model predictions match up against the collected data. The individual sensitivities of the subjects and the model have been normalized by plotting the thresholds relative to an anchor data point; no other



(a) (b)

(c) (d)

**Figure 6.5**: Disparity energy model computation. (a) Spectrum of filter for 7 cpd, 3.72 arcmin, iso+ condition. Gray indicates 1, white indicates 0. (b) Projection of filter spectrum onto $x$-axis results in vertical contour power spectrum $S_f(\xi_1, \xi_2)$. (c) Binocular phase weighting function $w_p(\xi_1, d)$ for $d = 3.72'$. (d) Final power spectrum. The sum of this function across frequencies gives the total disparity energy for this filter and disparity.

manipulation has been performed. In Fig. 6.7, data is plotted for the 2 cpd condition and is anchored to the iso+ data point. Note how the model not only captures the trends but also the differences in thresholds between data points. The model predictions are fairly accurate, staying within the error bars for most data points. The slight variations in AM's data are likely due to some astigmatism. Fig. 6.8 shows the predicted model results for all orientations between 0 and 90 degrees for

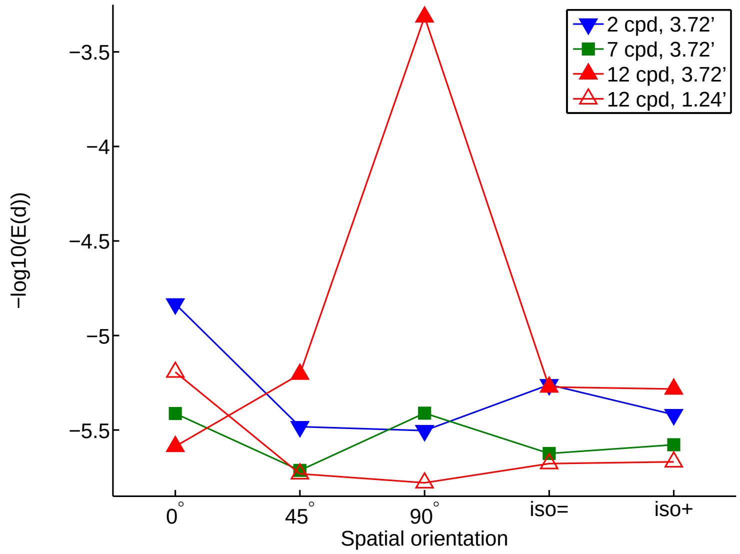


**Figure 6.6**: Relative thresholds predicted by the disparity energy model.

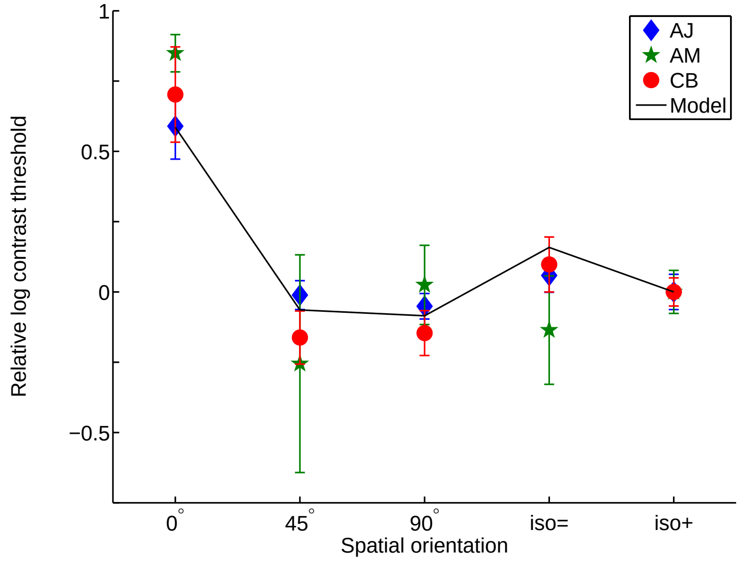


**Figure 6.7**: Thresholds for the 2 cpd at 3.72' condition, relative to the iso+ orientation, as predicted by the disparity energy model and measured experimentally.

12 cpd and 1.24', anchored at the oblique orientation. The experimental results are also shown, and good agreement is again found between the model and two subjects. The nonconforming data of AJ is accounted for by the lack of sensitivity to oblique contours. Otherwise, the trend between the horizontal and vertical orientations is roughly followed.

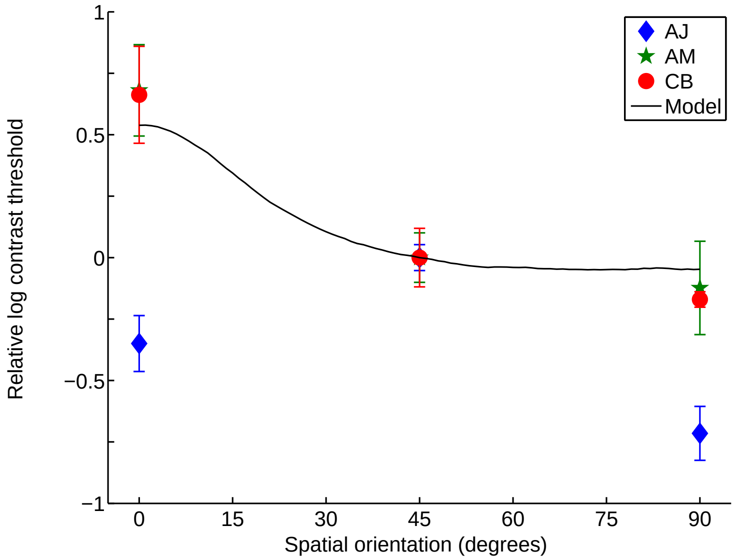


**Figure 6.8**: Relative thresholds as predicted by the disparity energy model for spatial orientations from 0 to 90 degrees for 12 cpd and 1.24'. Experimental results for the 3 subjects and 3 anisotropic orientations are overlayed. All thresholds are displayed relative to the oblique orientation.

## 6.3 Discussion

Our experiments show a clear anisotropy in sensitivity for depth detection from luminance contours. Generally, sensitivity increases as the pattern orientation goes from horizontal to vertical. For isotropic patterns, sensitivity decreases as Fourier components are allocated to orientations away from vertical when spectral support is preserved. Sensitivity is restored if more spectral components are added, but is not increased beyond the sensitivity attained with predominantly vertical Fourier components.

Although vertical contours most efficiently transmit horizontal disparity information, other orientations of luminance patterns still foster stereopsis. This ability is due to the fact that all non-horizontal orientations carry a vertical component, and therefore information relevant to stereopsis. Our disparity energy model suggests that the visual system weights the contributions of these spectral components based on how effective they are at discerning a particular disparity. Both the model predictions as well as the corroborating experimental data are in agreement with the findings of [122–124], where it is found that many orientations transmit disparity information, but vertical does so most effectively.

Our results indicate a greater extent of spatial summation in the vertical direction than the horizontal direction. Integrating along the (non-horizontal) luminance contours that transmit horizontal disparity information is advantageous because it allows the visual system to boost its signal-to-noise ratio (SNR) prior to stereo matching without greatly sacrificing resolution in depth. The opposite anisotropy is seen for spatial integration of disparity values, where sensitivity, and therefore integration extent, is greater when corrugations are extended in the horizontal direction. However, if disparities from horizontal edges are pooled anyway, then why not also perform this integration in luminance prior to stereo matching? Such a scheme would economize on the computation necessary to solve the correspondence problem.

One possible explanation might appeal to the fact that stereopsis is a shift-varying, nonlinear process. Since stereopsis has a global optimization stage following a matching stage, filtering input images prior to disparity computation will generally produce a different result than filtering disparities computed from the raw images. An advantage of the post-filtering scheme for horizontal contours is that the visual system can adequately balance disparity noise and resolution. Initially, disparity can be computed at high horizontal resolution by pre-filtering the retinal images using a vertically elongated kernel. If horizontal edges in depth are sensed, then the estimates can be pooled in this direction to increase sensitivity. In this manner, the visual system can compensate for the lack of efficiency in transmission of horizontal disparity information by horizontal contours while computing depth with high resolution.

## 6.4 Applications

While our experimental and computational results are interesting from a vision science perspective, they can also be applied to video compression and computer vision in several ways.

In mixed resolution compression, sometimes downsampling is performed in one direction, horizontal or vertical, only. Since there are different correlations in the horizontal and vertical directions, the choice of direction is not arbitrary. In [128], the decision of which direction to subsample is governed by a rate distortion curve based on PSNR. Using our model, the rate distortion could also include a term based on depth sensitivity so that the decision is perceptually aware. This modification could result in higher quality videos for a given bandwidth or allow for stronger downsampling in one direction than the other while preserving the depth impression. Another scheme for mixed resolution was proposed in [129], where one view is subsampled horizontally and the other view is subsampled vertically by the same factor. A similar decision is made, based on rate distortion, about which view to subsample in which direction. Again, our computational model for contrast sensitivity for depth detection could be employed here. Further, it may be possible to subsample more vertically and less horizontally based on our results, which could increase quality while maintaining bandwidth.

Oftentimes, a more simplistic representation of stereo data is used to leverage existing 2D coding and transport technology. The two stereo views are subsampled by a factor of two, and combined into a single frame. Then, a standard 2D video codec is used to compress the video. At the decoder, both views are upsampled to full resolution. Clearly, there are two methods to implement this frame-packing scheme: the two stereo views can be subsampled horizontally or vertically. Our results suggest that subsampling vertically is a safer option in terms of preserving the impression of depth. In this manner, the horizontal resolution is maintained and there is more energy left in the vertical contours for stereopsis.

Our results and model can be applied to stereo image metrics and depth map compression. For a given depth map and corresponding luminance image, our model could be used to decide what regions of the stereo image will appear flat to the viewer based on its disparity, contrast, and dominant texture orientation. Such regions can be smoothed out in the depth map, resulting in a lower bit rate. In comparing two stereo images that have undergone certain processing, our model could again be applied to determine the affect of the processing degradation on perceived depth. For instance, the test image could be decomposed by oriented filters, the disparity energy could be calculated, and the depth sensitivity could be estimated. Such applications have been proposed and successfully demonstrated in [130] using contrast sensitivity functions for depth detection. Integrating our model into an existing codec, similar to how depth masking was employed in H.264 in [131], is another possible application of our work. The addition of our model would extend these works to include considerations for orientation.

## 6.5 Conclusion

It has long been assumed that vertical contours are responsible for stereopsis. Previous work has demonstrated that contours at other orientations are also sufficiently rich in features for stereopsis [122, 124], but that sensitivity to vertical orientations is still dominant. In a more extensive, direct experiment, we obtain contrast sensitivity functions for depth detection as a function of bandpass filter orientation. We find clear evidence for contour anisotropy, which also supports previous findings.

However, our results show that the relationship between depth sensitivity and pattern orientation also depends on other factors such as spatial frequency and disparity. Certain combinations of these parameters can produce seemingly anomalous results, such as a greater sensitivity to horizontally oriented patterns. We are able to account for this data through our disparity energy model, which says that vertical energy in the power spectrum of a stimulus weighted by a binocular phase-dependent function is a good indicator of relative sensitivity thresholds. The model determines what allocation of frequency components will result in a strong ability for depth detection. Our model is based on, and provides support for, the phase model of disparity encoding.

# Acknowledgments

The text of Chapter 6 is, in part, a reprint of a paper that is being prepared for publication as: A. K. Jain, A. E. Robinson, D. I. A. MacLeod, and T. Q. Nguyen, “Anisotropic spatial integration in the sensing of horizontal disparity,” *Vision Research*, in preparation. The dissertation author was the primary investigator and author of this paper.

# Chapter 7

# Conclusion

In the case where one eye sees a blurry view and the other sees a sharp view, the human visual system attempts to maximize the information it receives by weighting the view with the higher spatial frequency content more heavily. This phenomenon of binocular suppression is interesting from a psychophysical perspective and also has important consequences for 3D video processing.

## 7.1 Looking Back

We began with investigating the question as to whether blur needs to be temporally balanced between the two eyes in mixed resolution. This was an open question in literature, and one that needs to be answered in order for this compression technique to be viable—if the compressed videos are uncomfortable to view, then mixed resolution compression will not work. We compared the traditional method of mixed resolution, where one eye continually receives the low resolution view, to a balanced blur method where the blurry view alternates between views at each frame. At sufficiently high frame rates, there was no real difference in perceived quality. In terms of visual comfort, the two methods were also equal, except for some content dependence in favor of alternating-eye blur.

We also developed web technology for collecting data for video quality experiments that enabled the fatigue experiment. Our tool makes data collection more error-proof, easier to duplicate and share, allows for simultaneous subject testing, and works with 2D and 3D videos. We have released our software as free and open source.

Another important aspect of a compression algorithm is the ability to restore quality. Having shown it is not necessary to balance blur and given its simplicity for encoding, the single-eye blur method is generally the best choice for mixed resolution coding, and accordingly, we chose to further develop it. Mixed resolution is particularly amenable to super-resolution because for each low resolution frame, there is a high resolution reference. Previous approaches have been successful, but have focused solely on images. Here, we extended example-based approaches to the stereo video domain, resulting in temporally consistent video.

Crucial to our super-resolution work is the use of spatio-temporal stereo matching to allow for temporal consistency, and to increase match accuracy. But to what degree does motion aid the matching process? Prior work in spatio-temporal disparity estimation has been based on the intuition that motion can help disambiguate potential matches, and have empirically or heuristically evaluated this effect and determined parameters. We approached the question from a theoretical standpoint, deriving probability of matching error expressions as a function of motion distribution, noise, and image feature. Our results can be used to tune parameters and develop efficiency metrics for spatio-temporal disparity estimation algorithms.

Finally, after studying temporal aspects of mixed resolution coding and restoration, we sought more efficient depth representations in the spatial domain. Returning to the mixed resolution problem, little attention has been paid to how to perform downsampling for the low resolution view, or for stereo in general. Downsampling affects depth perception, but there has been no real attempt at establishing the relationship. Given the visual system's natural preference for vertical contours for stereopsis, it would be sensible to downsample more in the vertical than horizontal direction in order to preserve depth resolution. We studied this relationship by measuring contrast sensitivity functions for depth detection from oriented bandpass filtered random noise. We found a clear anisotropy in depth detection, and developed a computational model that explains the data.

Our model predicts how sensitive the visual system is for depth detection given an input image. This is a potentially powerful tool that can be used in stereo compression, depth map compression, and stereo metrics.

In total, this work establishes mixed resolution coding as a viable option for compression stereoscopic video. We have shown that such content is comfortable to view even with the simple encoding method of only blurring one eye's view, and that the missing resolution can be recovered at the decoder with good fidelity. In doing so, we develop two computational models: one for spatio-temporal disparity estimation, and one for the perception of depth from filtered random noise stimuli. These models, while important for mixed resolution applications, also have more general use in stereo algorithms and models of vision.

## 7.2 Looking Ahead

Our work can be developed in several ways. While we have focused on the application of mixed resolution to compression, the same concept can be applied in stereo rendering. Given one image, sometimes a companion image needs to be generated to form a stereo pair, as in rendering from a plenoptic camera, database of images, in view synthesis, or video+depth decoding. Rendering the second image at a lower resolution decreases computation time as well as the potential for artifacts. If artifacts can be detected, they could be mitigated with blur.

There is much engineering work left to do in order to implement these results into a codec. For instance, the content dependence of balancing blur could be further investigated, and a codec could be developed to select the appropriate blur scheme and kernel to ensure visual comfort. The super-resolution method could be optimized and integrated into a decoder to enhance quality when necessary. Our computational model for the efficacy of motion in stereo matching could be extended to more complex scenarios, such as those involving occlusion or global optimization schemes. It could also be developed into a metric for video disparity estimation algorithms. Similarly, our model for depth sensitivity could be tested on natural images and made into a metric for rate distortion or stereo quality. These expressions could then be integrated into a codec that optimizes subsampling for depth salience.

In the realm of vision science, our study into stereo anisotropy can be extended by comparing the anisotropies found in the cyclopean domain versus that found in the luminance domain. For instance, measuring sensitivities of cyclopean edges at different orientations when the stimulus is a random dot stereogram modulated by an oriented luminance texture might reveal certain interactions between the two anisotropies.

# Bibliography

[1] C. Wheatstone, "Contributions to the physiology of vision.–part the first. on some remarkable, and hitherto unobserved, phenomena of binocular vision," *Philosophical Transactions of the Royal Society of London*, vol. 128, pp. 371–394, 1838.

[2] L. Lipton, *Foundations of the Stereoscopic Cinema.* Van Nostrand Reinhold, 1982.

[3] I. P. Howard, *Seeing in Depth.* I. Porteous, 2002.

[4] K. N. Ogle, "Induced size effect: I. a new phenomenon in binocular space perception associated with the relative sizes of the images of the two eyes," *Archives of Ophthalmology*, vol. 20, no. 4, pp. 604–623, 1938.

[5] C. Tyler, "Binocular vision," in *Duane's Foundations of Clinical Ophthalmology, vol. 2*, W. Tasman and E. Jaeger, Eds. Philadelphia: J.B. Lippincott Co., 2004.

[6] B. Julesz, "Binocular depth perception of computer-generated patterns," *Bell System Tech.*, vol. 39, no. 5, pp. 1125–1161, Sep. 1960.

[7] ——, *Foundations of Cyclopean Perception.* Univ. Chicago Press, 1971.

[8] A. E. Robinson, A. K. Jain, M. Scott, D. I. A. MacLeod, and T. Q. Nguyen, "Apparent sharpness of 3D video when one eye's view is more blurry," *i-Perception*, vol. 4, no. 6, pp. 456–467, 2013.

[9] R. F. Hess, C. H. Liu, and Y.-Z. Wang, "Differential binocular input and local stereopsis," *Vision Research*, vol. 43, no. 22, pp. 2303–2313, 2003.

[10] G. Westheimer and S. P. McKee, "Stereoscopic acuity with defocused and spatially filtered retinal images," *J. Opt. Soc. Am.*, vol. 70, no. 7, pp. 772–778, Jul. 1980.

[11] L. Stelmach, W. J. Tam, D. Meegan, and A. Vincent, "Stereo image quality: effects of mixed spatio-temporal resolution," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, no. 2, pp. 188–193, Mar. 2000.

[12] A. K. Jain, C. Bal, A. Robinson, D. I. A. MacLeod, and T. Q. Nguyen, "Temporal aspects of binocular suppression in 3D video," in *Sixth Int'l. Workshop Video Proc. Qual. Metrics Consum. Elec. (VPQM)*, Jan. 2012, pp. 93–98.

[13] C. W. Tyler, "Stereoscopic vision: Cortical limitations and a disparity scaling effect," *Science*, vol. 181, no. 4096, pp. 276–278, 1973.

[14] D. Scharstein and C. Pal, "Learning conditional random fields for stereo," in *IEEE Conference on Computer Vision and Pattern Recognition, 2007*, Jun. 2007, pp. 1–8.

[15] D. Scharstein and R. Szeliski. (2010, May) Middlebury stereo vision page. [Online]. Available: http://vision.middlebury.edu/stereo/

[16] P. Aflaki, M. Hannuksela, J. Häkkinen, P. Lindroos, and M. Gabbouj, "Impact of downsampling ratio in mixed-resolution stereoscopic video," in *3DTV-Conference*, June 2010, pp. 1–4.

[17] V. De Silva, H. Arachchi, E. Ekmekcioglu, A. Fernando, S. Dogan, A. Kondoz, and S. Savas, "Psycho-physical limits of interocular blur suppression and its application to asymmetric stereoscopic video delivery," in *19th International Packet Video Workshop*, May 2012, pp. 184–189.

[18] Y. Chen, S. Liu, Y.-K. Wang, M. Hannuksela, H. Li, and M. Gabbouj, "Low-complexity asymmetric multiview video coding," in *Multimedia and Expo, 2008 IEEE International Conference on*, Apr. 2008, pp. 773–776.

[19] J. Quan, M. Hannuksela, and H. Li, "Asymmetric spatial scalability in stereoscopic video coding," in *3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video*, May 2011, pp. 1–4.

[20] H. Brust, G. Tech, K. Mueller, and T. Wiegand, "Mixed resolution coding with inter view prediction for mobile 3dtv," in *3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video*, Jun. 2010, pp. 1–4.

[21] I. Dinstein, M. G. Kim, J. Tselgov, and A. Henik, "Compression of stereo images and the evaluation of its effects on 3-d perception," in *Applications of Digital Image Processing XII*, vol. 1153. SPIE, Jan. 1990, pp. 522–1187.

[22] M. Perkins, "Data compression of stereopairs," *IEEE Trans. Commun.*, vol. 40, no. 4, pp. 684–696, Apr. 1992.

[23] S. Sethuraman, M. Siegel, and A. Jordan, "A multiresolution framework for stereoscopic image sequence compression," in *Image Processing, 1994. Proceedings. IEEE International Conference on*, vol. 2, Nov. 1994, pp. 361–365.

[24] M. Azimi, S. Valizadeh, X. Li, L. Coria, and P. Nasiopoulos, "Subjective study on asymmetric stereoscopic video with low-pass filtered slices," in *Int'l Conf. on Computing, Networking and Communications*, Feb. 2012, pp. 719–723.

[25] S. Liu, F. Liu, J. Fan, and H. Xia, "Asymmetric stereoscopic video encoding algorithm based on subjective visual characteristic," in *Int'l Conf. on Wireless Comm. Sig. Proc.*, Nov. 2009, pp. 1–5.

[26] W. J. Tam, L. B. Stelmach, and S. Subramaniam, "Stereoscopic video: asymmetrical coding with temporal interleaving," in *Stereoscopic Displays and Virtual Reality Systems VIII*, vol. 4297. SPIE, 2001, pp. 299–306.

[27] W. J. Tam, L. B. Stelmach, F. Speranza, and R. Renaud, "Cross-switching in asymmetrical coding for stereoscopic video," in *Stereoscopic Displays and Virtual Reality Systems IX*, vol. 4660. SPIE, 2002, pp. 95–104.

[28] D. Garcia, C. Dorea, and R. De Queiroz, "Super resolution for multiview images using depth information," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 9, pp. 1249–1256, Sep. 2012.

[29] M. Emoto, T. Niida, and F. Okano, "Repeated vergence adaptation causes the decline of visual functions in watching stereoscopic television," *Display Technology, Journal of*, vol. 1, no. 2, pp. 328–340, Dec. 2005.

[30] T. Bando, A. Iijima, and S. Yano, "Visual fatigue caused by stereoscopic images and the search for the requirement to prevent them: A review," *Displays*, vol. 33, no. 2, pp. 76–83, 2012.

[31] S. Yano, S. Ide, T. Mitsuhashi, and H. Thwaites, "A study of visual fatigue and visual comfort for 3d hdtv/hdtv images," *Displays*, vol. 23, no. 4, pp. 191–201, 2002.

[32] M. Lambooij, W. IJsselsteijn, M. Fortuin, and I. Heynderickx, "Visual discomfort and visual fatigue of stereoscopic displays: A review," *Journal of Imaging Science and Technology*, vol. 53, no. 3, pp. 1–14, 2009.

[33] T. Shibata, J. Kim, D. Hoffman, and M. Banks, "The zone of comfort: Predicting visual discomfort with stereo displays," *Journal of Vision*, vol. 11, no. 8, 2011.

[34] S. Jumisko-Pyykkö, T. Utriainen, D. Strohmeier, A. Boev, and K. Kunze, "Simulator sickness – five experiments using autostereoscopic mid-sized or small mobile screens," in *3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video*, Jun. 2010, pp. 1–4.

[35] D. H. Brainard, “The psychophysics toolbox,” *Spatial vision*, vol. 10, no. 4, pp. 433–436, 1997.

[36] D. G. Pelli, “The videotoolbox software for visual psychophysics: transforming numbers into movies,” *Spatial Vision*, vol. 10, pp. 437–442, 1997.

[37] M. Kleiner, D. Brainard, and D. Pelli, “What’s new in psychtoolbox-3?” *Perception ECVP Abstract Supplement*, vol. 36, 2007.

[38] A. B. Watson and A. J. Ahumada, “Blur clarified: A review and synthesis of blur discrimination,” *Journal of Vision*, vol. 11, no. 5, 2011.

[39] A. K. Jain, A. E. Robinson, and T. Q. Nguyen, “Comparing perceived quality and fatigue for two methods of mixed resolution stereoscopic coding,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 3, pp. 418–429, Mar. 2014.

[40] ITU-R, “Recommendation ITU-R BT.500-13: Methodology for the subjective assessment of the quality of television pictures,” International Telecommunication Union, Recommendation, Jan. 2012.

[41] A. K. Jain, C. Bal, and T. Q. Nguyen, “Tally: A web-based subjective testing tool,” in *Fifth Int’l. Workshop on Qual. of Multim. Exp. (QoMEX)*, Jul. 2013, pp. 128–129.

[42] T. Takeuchi and K. K. De Valois, “Perceived sharpness of moving natural images,” *Journal of Vision*, vol. 4, no. 8, p. 490, 2004.

[43] FreeHD3D. (2012, Jun.) Welcome to the world of the 3D Side-by-Side clips. [Online]. Available: http://www.freehd3d.info/

[44] R. C. Blair and W. Karniski, “An alternative method for significance testing of waveform difference potentials,” *Psychophysiology*, vol. 30, no. 3, 1993.

[45] K. Ukai and P. A. Howarth, “Visual fatigue caused by viewing stereoscopic motion images: Background, theories, and observations,” *Displays*, vol. 29, no. 2, pp. 106–116, 2008.

[46] J. Kopf, M. F. Cohen, D. Lischinski, and M. Uyttendaele, “Joint bilateral upsampling,” *ACM Trans. Graph.*, vol. 26, no. 3, Jul. 2007.

[47] K. Seshadrinathan, R. Soundararajan, A. Bovik, and L. Cormack, “Study of subjective and objective quality assessment of video,” *IEEE Trans. Image Process.*, vol. 19, no. 6, pp. 1427–1441, Jun. 2010.

[48] MSU Graphics & Media Lab (Video Group). (2012, Sep.) MSU perceptual video quality tool. [Online]. Available: http://www.compression.ru/video/quality_measure/perceptual_video_quality_tool_en.html

[49] Qt Development Frameworks – Nokia Corporation. (2012, Apr.) Qt – cross-platform application and UI framework. [Online]. Available: http://qt.nokia.com/

[50] B. Lepilleur. (2012, Apr.) JsonCpp Documentation. [Online]. Available: http://jsoncpp.sourceforge.net/

[51] VideoLAN Organization. (2012, Dec.) VLC media player. [Online]. Available: http://www.videolan.org/

[52] Python Community. (2012, Apr.) Python programming language - official website. [Online]. Available: http://www.python.org/

[53] Django Software Foundation. (2012, Mar.) Django – the web framework for perfectionists with deadlines. [Online]. Available: https://www.djangoproject.com/

[54] B. Chesneau and P. J. Davis. (2012, Jul.) Gunicorn. [Online]. Available: http://gunicorn.org/

[55] Ruby on Rails Community. (2012, Jul.) Web development that doesn't hurt. [Online]. Available: http://rubyonrails.org/

[56] Ruby Community. (2012, Jul.) Ruby – a programmer's best friend. [Online]. Available: http://www.ruby-lang.org/

[57] Twitter, Inc. (2012, Jun.) Bootstrap, from Twitter. [Online]. Available: http://twitter.github.com/bootstrap/

[58] The jQuery Foundation. (2012, Apr.) jQuery mobile framework. [Online]. Available: http://jquerymobile.com/

[59] GitHub, Inc. (2013) github. [Online]. Available: https://github.com/

[60] A. K. Jain, L. C. Tran, R. Khoshabeh, and T. Q. Nguyen, "Efficient stereo-to-multiview synthesis," in *IEEE Int'l. Conf. Acoust., Speech, Sig. Proc. (ICASSP)*, May 2011, pp. 889–892.

[61] L. C. Tran, R. Khoshabeh, A. K. Jain, C. Pal, and T. Q. Nguyen, "Spatially consistent view synthesis with coordinate alignment," in *IEEE Int'l. Conf. Acoust., Speech, Sig. Proc. (ICASSP)*, May 2011, pp. 905–908.

[62] A. K. Jain and T. Q. Nguyen, "Video super-resolution for mixed resolution stereo," in *IEEE Int'l. Conf. Image Proc. (ICIP)*, Sep. 2013, pp. 962–966.

[63] J. Tian, L. Chen, and Z. Liu, "Dual regularization-based image resolution enhancement for asymmetric stereoscopic images," *Signal Processing*, vol. 92, no. 2, pp. 490–497, 2012.

[64] M. Gao, S. Ma, D. Zhao, and W. Gao, "A spatial inter-view auto-regressive super-resolution scheme for multi-view image via scene matching algorithm," in *Circuits and Systems (ISCAS), 2013 IEEE International Symposium on*, 2013, pp. 2880–2883.

[65] H. S. Sawhney, Y. Guo, K. Hanna, R. Kumar, S. Adkins, and S. Zhou, "Hybrid stereo camera: An ibr approach for synthesis of very high resolution stereoscopic image sequences," in *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '01. New York, NY, USA: ACM, 2001, pp. 451–460.

[66] E. Hung, R. De Queiroz, F. Brandi, K. de Oliveira, and D. Mukherjee, "Video super-resolution using codebooks derived from key-frames," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 9, pp. 1321–1331, 2012.

[67] B. Zhang, J. Liu, J. Ge, C. Chen, H. Yuan, and W. Liu, "A super resolution reconstruction scheme for mixed spatio-temporal stereo video," in *Audio, Language and Image Processing (ICALIP), 2012 International Conference on*, Jul. 2012, pp. 490–496.

[68] T. Richter, J. Seiler, W. Schnurrer, and A. Kaup, "Robust super-resolution in a multiview setup based on refined high-frequency synthesis," in *Multimedia Signal Processing (MMSP), 2012 IEEE 14th International Workshop on*, 2012, pp. 7–12.

[69] O. Williams, M. Isard, and J. MacCormick, "Estimating disparity and occlusions in stereo video sequences," in *Proc. of 2005 IEEE Conf. on Comp. Vis. and Patt. Recog.*, 2005, pp. 250–257.

[70] J. Zhu, L. Wang, J. Gao, and R. Yang, "Spatial-temporal fusion for high accuracy depth maps using dynamic MRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 5, pp. 899–909, May 2010.

[71] M. Sizintsev and R. Wildes, "Spatiotemporal stereo and scene flow via stequel matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 6, pp. 1206–1219, Jun. 2012.

[72] J. Sanchez-Riera, J. Cech, and R. P. Horaud, "Robust spatiotemporal stereo for dynamic scenes," in *21st Intl. Conf. on Patt. Recog.*, Dec. 2012.

[73] S. H. Chan, R. Khoshabeh, K. B. Gibson, P. E. Gill, and T. Q. Nguyen, "An augmented lagrangian method for total variation video restoration," *IEEE Trans. Image Process.*, vol. 20, no. 11, pp. 3097–3111, Nov. 2011.

[74] W. T. Freeman and E. C. Pasztor, "Learning low-level vision," *International Journal of Computer Vision*, vol. 40, pp. 25–47, Oct. 2000.

[75] W. T. Freeman, T. R. Jones, and E. C. Pasztor, “Example-based super-resolution,” *IEEE Comput. Graph. Appl.*, vol. 22, no. 2, pp. 56–65, Mar. 2002.

[76] C. M. Bishop, A. Blake, and B. Marthi, “Super-resolution enhancement of video,” in *In Proc. Artificial Intelligence and Statistics*, 2003.

[77] A. K. Jain and T. Q. Nguyen, “Discriminability limits in spatio-temporal stereo block matching,” *IEEE Trans. Image Process.*, vol. 23, no. 5, pp. 2328–2342, May 2014.

[78] Nagoya University. (2013, Dec.) MPEG-FTV Project. [Online]. Available: http://www.tanimoto.nuee.nagoya-u.ac.jp/MPEG-FTVProject.html

[79] Mobile3DTV. (2013, Dec.) 3D video database. [Online]. Available: http://sp.cs.tut.fi/mobile3dtv/stereo-video/

[80] Poznan University of Technology. (2013, Dec.). [Online]. Available: ftp://multimedia.edu.pl/CFP/

[81] M. Domaski, T. Grajek, K. Klimaszewski, M. Kurc, O. Stankiewicz, J. Stankowski, and K. Wegner, “Poznan multiview video test sequences and camera parameters,” in *ISO/IEC JTC1/SC29/WG11 MPEG 2009/M17050*, Xian, China, Oct. 2009.

[82] VideoLAN Organization. (2013, Nov.) x264. [Online]. Available: http://www.videolan.org/developers/x264.html

[83] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[84] D. Scharstein and R. Szeliski, “A taxonomy and evaluation of dense two-frame stereo correspondence algorithms,” *International Journal of Computer Vision*, vol. 47, no. 1-3, pp. 7–42, 2002.

[85] J. Neumann and Y. Aloimonos, “Spatio-temporal stereo using multi-resolution subdivision surfaces,” *International Journal of Computer Vision*, vol. 47, no. 1-3, pp. 181–193, 2002.

[86] L. Zhang, B. Curless, and S. Seitz, “Spacetime stereo: shape recovery for dynamic scenes,” in *Computer Vision and Pattern Recognition, IEEE Conference on*, vol. 2, 2003, pp. 367–74.

[87] C. Leung, B. Appleton, B. Lovell, and C. Sun, “An energy minimisation approach to stereo-temporal dense reconstruction,” in *Pattern Recognition (ICPR), Proceedings of the 17th International Conference on*, vol. 4, 2004, pp. 72–75.

[88] M. Gong, “Enforcing temporal consistency in real-time stereo estimation,” in *9th European Conference on Computer Vision (ECCV) Proceedings, Part III*, ser. Lecture Notes in Computer Science, A. Leonardis, H. Bischof, and A. Pinz, Eds. Springer, 2006, pp. 564–577.

[89] S.-B. Lee and Y.-S. Ho, “Temporally consistent depth map estimation using motion estimation for 3DTV,” in *International Workshop on Advanced Image Technology*, vol. 149, 2010, pp. 1–6.

[90] G. Zhang, J. Jia, T.-T. Wong, and H. Bao, “Consistent depth maps recovery from a video sequence,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 6, pp. 974–988, 2009.

[91] A. Hosni, C. Rhemann, M. Bleyer, and M. Gelautz, “Temporally consistent disparity and optical flow via efficient spatio-temporal filtering,” in *Proceedings of the 5th Pacific Rim conference on Advances in Image and Video Technology - Volume Part I*, 2012, pp. 165–177.

[92] K.-J. Yoon and I.-S. Kweon, “Locally adaptive support-weight approach for visual correspondence search,” in *Computer Vision and Pattern Recognition, IEEE Conference on*, vol. 2, 2005, pp. 924–931.

[93] C. Richardt, D. Orr, I. Davies, A. Criminisi, and N. A. Dodgson, “Real-time spatiotemporal stereo matching using the dual-cross-bilateral grid,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, ser. Lecture Notes in Computer Science, vol. 6313, Sep. 2010, p. 510523.

[94] J. Kowalczuk, E. Psota, and L. Perez, “Real-time temporal stereo matching using iterative adaptive support weights,” in *IEEE International Conference on Electro/Information Technology*, 2013, pp. 1–6.

[95] Z. Lee, J. Juang, and T. Nguyen, “Local disparity estimation with three-moded cross census and advanced support weight,” *IEEE Trans. Multimedia*, vol. 15, no. 8, pp. 1855–1864, 2013.

[96] O. Shahar, A. Faktor, and M. Irani, “Space-time super-resolution from a single video,” in *Computer Vision and Pattern Recognition, IEEE Conference on*, 2011, pp. 3353–3360.

[97] Y. Wexler, E. Shechtman, and M. Irani, “Space-time completion of video,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 3, pp. 463–476, 2007.

[98] M. Maggioni, G. Boracchi, A. Foi, and K. Egiazarian, “Video denoising using separable 4d nonlocal spatiotemporal transforms,” in *Proc. SPIE*, vol. 7870, 2011, pp. 787 003–787 003–11.

[99] D. Robinson and P. Milanfar, "Fundamental performance limits in image registration," *IEEE Trans. Image Process.*, vol. 13, no. 9, pp. 1185–1199, 2004.

[100] N. Sabater, J.-M. Morel, and A. Almansa, "How accurate can block matches be in stereo vision?" *SIAM J. Img. Sci.*, vol. 4, no. 1, pp. 472–500, Mar. 2011.

[101] A. O'Connor, S. Krishnan, and V. Vaishampayan, "Accuracy of depth and orientation estimation based on texture characteristics," in *Signal Processing and Communications, International Conference on*, 2010, pp. 1–5.

[102] J. Davis, D. Nehab, R. Ramamoorthi, and S. Rusinkiewicz, "Spacetime stereo: a unifying framework for depth from triangulation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 2, pp. 296–302, 2005.

[103] Y. Swirski, Y. Schechner, and T. Nir, "Variational stereo in dynamic illumination," in *Computer Vision (ICCV), 2011 IEEE International Conference on*, 2011, pp. 1124–1131.

[104] Panasonic. (2013, Sep.) LUMIX ZS30. [Online]. Available: http://shop.panasonic.com/shop/model/DMC-ZS30K

[105] Canon U.S.A. (2013, Sep.) EOS Rebel T4i. [Online]. Available: http://www.usa.canon.com/cusa/support/consumer/eos_slr_camera_systems/eos_digital_slr_cameras/eos_rebel_t4i

[106] Red. (2013, Sep.) Epic Dragon. [Online]. Available: http://www.red.com/products/epic-dragon

[107] B. L. A. Raymond van Ee, "Motion direction, speed and orientation in binocular matching," *Nature*, vol. 410, no. 6829, p. 690694, Apr. 2001.

[108] E. Sjoberg, K. Squire, and C. Martell, "Online parameter estimation of a robot's motion model," in *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, 2007, pp. 735–740.

[109] H. Yuan, Y. Chang, Z. Lu, and Y. Ma, "Model based motion vector predictor for zoom motion," *IEEE Signal Process. Lett.*, vol. 17, no. 9, pp. 787–790, 2010.

[110] H. Tao, H. Sawhney, and R. Kumar, "Dynamic depth recovery from multiple synchronized video streams," in *Computer Vision and Pattern Recognition, IEEE Conference on*, vol. 1, 2001, pp. 118–124.

[111] E. Trulls, A. Sanfeliu, and F. Moreno-Noguer, “Spatiotemporal descriptor for wide-baseline stereo reconstruction of non-rigid and ambiguous scenes,” in *Proceedings of the 12th European conference on Computer Vision - Volume Part III*, 2012, pp. 441–454.

[112] J. Sun, N.-N. Zheng, and H.-Y. Shum, “Stereo matching using belief propagation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 7, pp. 787–800, 2003.

[113] J. Delon and B. Rougé, “Small baseline stereovision,” *Journal of Mathematical Imaging and Vision*, vol. 28, no. 3, pp. 209–223, 2007.

[114] I. Ohzawa, G. C. Deangelis, and R. D. Freeman, “Stereoscopic depth discrimination in the visual cortex: Neurons ideally suited as disparity detectors,” *Science (New York, N.Y.)*, vol. 249, no. 4972, pp. 1037–1041, Aug. 1990.

[115] I. Ohzawa, G. C. DeAngelis, and R. D. Freeman, “Encoding of binocular disparity by simple cells in the cat’s visual cortex,” *Journal of Neurophysiology*, vol. 75, no. 5, pp. 1779–1805, 1996.

[116] G. DeAngelis, I. Ohzawa, and R. Freeman, “Depth is encoded in the visual cortex by a specialized receptive field structure,” *Nature*, vol. 352, no. 6331, pp. 156–159, 1991.

[117] B. G. Cumming, “An unexpected specialization for horizontal disparity in primate primary visual cortex,” *Nature*, vol. 418, no. 6898, pp. 633–636, Aug. 2002.

[118] M. F. Bradshaw and B. J. Rogers, “Sensitivity to horizontal and vertical corrugations defined by binocular disparity,” *Vision Research*, vol. 39, no. 18, pp. 3049–3056, 1999.

[119] M. F. Bradshaw, P. B. Hibbard, A. D. Parton, D. Rose, and K. Langley, “Surface orientation, modulation frequency and the detection and perception of depth defined by binocular disparity and motion parallax,” *Vision Research*, vol. 46, no. 17, pp. 2636–2644, 2006.

[120] C. W. Tyler and L. L. Kontsevich, “Stereoprocessing of cyclopean depth images: horizontally elongated summation fields,” *Vision Research*, vol. 41, no. 17, pp. 2235–2243, 2001.

[121] I. Serrano-Pedraza, C. Brash, and J. C. A. Read, “Testing the horizontal-vertical stereo anisotropy with the critical-band masking paradigm,” *Journal of Vision*, vol. 13, no. 11, 2013.

[122] R. Blake, J. M. Camisa, and D. N. Antoinetti, “Binocular depth discrimination depends on orientation,” *Perception & Psychophysics*, vol. 20, no. 2, pp. 113–118, 1976.

[123] L. K. Cormack and R. B. Riddle, “Binocular correlation detection with oriented dynamic random-line stereograms,” *Vision Research*, vol. 36, no. 15, pp. 2303–2310, 1996.

[124] S. S. Patel, H. E. Bedell, and P. Sampat, “Pooling signals from vertically and non-vertically orientation-tuned disparity mechanisms in human stereopsis,” *Vision Research*, vol. 46, no. 1-2, pp. 1–13, 2006.

[125] J. Mansfield and A. Parker, “An orientation-tuned component in the contrast masking of stereopsis,” *Vision Research*, vol. 33, no. 11, pp. 1535–1544, 1993.

[126] S. J. D. Prince, R. A. Eagle, and B. J. Rogers, “Contrast masking reveals spatial frequency channels in stereopsis,” *Perception*, pp. 1345–1357, 1998.

[127] H. S. Smallman and D. I. A. MacLeod, “Size-disparity correlation in stereopsis at contrast threshold,” *J. Opt. Soc. Am. A*, vol. 11, no. 8, pp. 2169–2183, Aug. 1994.

[128] M. Yu, H. Yang, S. Fu, F. Li, R. Fu, and G. Jiang, “New sampling strategy in asymmetric stereoscopic video coding for mobile devices,” in *E-Product E-Service and E-Entertainment (ICEEE), 2010 International Conference on*, Nov. 2010, pp. 1–4.

[129] P. Aflaki, M. Hannuksela, M. Homayouni, and M. Gabbouj, “Cross-asymmetric mixed-resolution 3d video compression,” in *3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), 2012*, 2012, pp. 1–4.

[130] P. Didyk, T. Ritschel, E. Eisemann, K. Myszkowski, H.-P. Seidel, and W. Matusik, “A luminance-contrast-aware disparity model and applications,” *ACM Transactions on Graphics (Proceedings SIGGRAPH Asia 2012, Singapore)*, vol. 31, no. 6, 2012.

[131] D. Pajak, R. Herzog, R. Mantiuk, P. Didyk, E. Eisemann, K. Myszkowski, and K. Pulli, “Perceptual depth compression for stereo applications,” *Computer Graphics Forum (Proc. Eurographics 2014)*, vol. 33, no. 2, 2014.