Automated detection of defects in 3D movies

Yasin Nazzar

Master of Engineering

Electrical and Computer Engineering

Montreal, Quebec

2015-07-17

Thesis submitted to McGill University in partial fulfilment of the requirements of the degree of Master of Engineering.

ACKNOWLEDGEMENTS

First of all I would like to thank Professor James J. Clark for giving me the opportunity to work under him. The experience was highly memorable and it was very pleasing working under someone with such knowledge, patience and capability to motivate others. He is one of the nicest person I have ever worked with personally. He provided me with the resources and freedom to explore different ideas while maintaining constant guidance to avoid pitfalls.

I am eternally grateful to my parents for their ever lasting support in every aspect of my life and especially in my pursuit of a Master's degree. They embedded, within me, the importance of seeking knowledge, from an early age, which gave me the confidence to come this far. I would also like to thank my brother Nazmus for listening to me babble on about my research and thesis late into the nights, while he was busy with a thesis and research of his own. I am also grateful to the members of my family who are here in Montreal for their constant love and support.

I wish to thank Prof. Jeremy Cooperstock, Prof. Kaleem Siddiqi, Prof. Martin D. Levine, Prof. Tal Arbel, Prof. Doina Precup as well as Prof. James J. Clark for each of their courses. The knowledge obtained from their courses served as the foundation for the work in this thesis. I would also like to thank the staff here at the Centre for Intelligent Machines namely, Jan Binder, Nick Wilson and Marlene Gray for their support. Jan and Nick were always there to help with obtaining new equipment and software while Marlene helped with other administrative issues. I wish to thank my labmates Jonathan, Mehdi, Qing, Amin, Siavash, Niloofar and

Kevin who have helped me with my work, directly, or provided ideas through their insightful conversations. I am thankful to all of my friends here at McGill, including my labmates, who have helped made the journey an enjoyable one.

Lastly, I would like to acknowledge the financial support that I have received from McGill University and my supervisor James Clark throughout the duration of my studies. I am highly appreciative of this support as it allowed me to pursue this Master's program and conduct my work with ease of mind.

ABSTRACT

The aim of this thesis is to provide tools and generate knowledge that assist in the creation of content for 3D movies. Such movies have made a recent comeback to mainstream entertainment and it is vital to assess their quality. Bad stereo content will lead to discomfort for viewers that will hurt the popularity of 3D movies. In this thesis, we focus on stereo window violation as it is one of the common problems related to stereo content. The stereo window violation problem is analysed in the framework of computational stereo vision. We propose a method, for stereo window violation detection, that implements a depth based object tracker that is also aware of objects in focus. Using such a method we are able to identify objects that cause window violation and raise an alert accordingly. Secondly, we investigate conflicting monocular and stereoscopic cues as another problem that ruins the 3D perception and makes watching 3D movies more tiring. The thesis also proposes a methodology to estimate half occlusion regions at a pixel level. Half occlusions are identified as a key feature that allows us to determine depth boundaries. Additionally, half occlusions allow us to narrow down regions of interest to search for other problems such as occlusion and stereopsis conflicts. Our motivation is that the tools and knowledge generated through this thesis will be used on the sets of shooting 3D movies as well as during post production and as an aid for 2D-to3D conversions. Doing so will allow easier and automated assessment of stereo quality and lead to the creation of better quality movies uplifting the popularity of 3D movies for the long run.

ABRÉGÉ

L'objectif de cette thèse est de fournir des outils et de développer des connaissances qui aideront à la création de contenu pour les films en 3D. Ces films ont connu un récent regain de popularité et il est maintenant essentiel d'évaluer leur qualité. Un contenu stéréo de piètre qualité peut causer de l'inconfort chez les spectateurs, ce qui nuira à la popularité de la technique. Dans cette thèse, nous nous penchons sur la violation de fenêtre stéréoscopique puisque c'est un des problèmes fréquemment rencontré avec le contenu stéréo. Le problème de la violation de fenêtre stéréo est analysé à l'aide d'outils issus du domaine de la vision artificielle. Nous proposons une méthode pour sa détection qui met à profit une technique de suivi de la profondeur des objets de la scène dont la mise au point procure une image nette. L'utilisation d'un tel procédé permet d'identifier les objets qui causent une violation de fenêtre et de déclencher une alerte en conséquence. Nous étudions aussi un second problème pouvant nuire à la perception 3D et au confort de visionnement : les conflits d'indices de profondeurs monoculaires et stéréoscopiques. La thèse propose une méthodologie pour estimer les régions de semi-occlusion à un niveau de précision de l'ordre du pixel. Les semi-occlusions sont identifiées comme un élément clé, permettant de déterminer les discontinuités de la profondeur. En outre, les semi-occlusions permettent d'identifier les régions sujettes à d'autres problèmes tel que le conflit occlusion-stéréopsie. Notre motivation est de fournir un ensemble d'outils et de connaissances qui sera utilisé tant sur les plateaux de tournage des films en 3D, que pendant la post-production ou encore, pour aider au cours du processus de

conversion 2D à 3D. Cela permettra d'évaluer plus facilement et automatiquement la qualité de la stéréo et conduira à la création de films de meilleure qualité, édifiant la popularité des films en 3D à long terme.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1
Introduction

1.1 Background

The current movie industry places a heavy emphasis on 3D movies. These movies have taken a step further than traditional movies by incorporating the perception of depth to enhance the immersive experience. Depth perception is created and altered using stereoscopy. Stereoscopy is a technique that is used to create a perception of depth by taking a pair of images which have a different perspective of the same scene and presenting them to the left and right eye of a viewer. Such image pairs can be made by creating a copy of a current single image and adding horizontal displacements of the pixels based on the depth of the objects in the scene. Another method used by film creators is using a stereo camera set-up to acquire a pair of images from slightly different viewpoints. There are different ways in which the stereo cameras are set-up for such a purpose but the goal for these set-ups is to use a pair of lenses to simultaneously capture the same scene with a slight variation. Once the image pair is obtained a variety of display methods, such as anaglyph, polarization, etc. can be utilized to present these images to the viewer's eyes.

Even with the most artistic renderings, the perception of depth created using stereoscopy is still only a fake replication of the human visual system. This makes it all the more important to ensure the process is done as well as possible. A variety of artefacts can arise while creating stereoscopic content that generate stimuli not

usually experienced by the human visual system. Some of these have been known for a while as can be seen in the paper of Woods *et al.* [59]. Most of the problems mentioned in that work can be adjusted for by choosing the right parameters for the camera and display. Parameters include quantities like distance between the cameras, their alignment, etc. This will be explored further in the following sections. However, even with the perfect set-up, there are still possibilities of shooting scenes that are difficult for humans to comprehend. This will only deteriorate the viewer's interest in watching 3D movies instead of creating a new and more enjoyable movie watching experience. Development of computer vision techniques for detecting and identifying the latter type of problems is the core focus of this thesis.

Computer vision can be broken down to a variety of sub-fields focusing on performing tasks such as object recognition, object detection, motion estimation, pattern recognition, etc. All of these sub-tasks aim to achieve, at various levels of complexity, some of the functionalities of the human vision. Moreover, the solutions generated from the various sub-fields can be extended to a variety of real world applications, both independently as well as in conjunction with other solutions. For example, a robot with an object recognition capability can be made to follow a certain path or another object while motion estimation capabilities can help it understand how much it has travelled and its relative location with respect to its starting position.

The typical human vision utilizes the pair of eyes to obtain slightly different views of the same scene. The differences are generally a horizontal displacement of objects from one eye to another. These horizontal displacements are processed in our visual cortex to create a perception of depth [14]. Depth perception allows

us to perceive a three-dimensional view of our world. This is something that film-makers aim to provide. They aim to create a perception of depth to allow for a more immersive experience of movie watching and making the experience even more enjoyable. Although traditional movies contain a variety of depth cues such as depth of focus, relative sizes, motion parallax, etc. they do not always provide an immersive experience.

Stereo vision is a field within computer vision which focuses on developing algorithms that duplicate the stereopsis capabilities of human vision. Techniques in stereo vision lend themselves quite well to analysing the pairs of images obtained through stereoscopy. Using these methods we can develop understanding of the scenes by estimating the perceived depth of a particular scene or decide if the scene composition is likely to be a poor viewing experience for the viewers. Additionally it can allow us to build systems to analyse and assist production of stereoscopic content on-scene as well as during post-production.

Some of the specific problems that will be looked at, in this thesis, are stereo window violations, conflicting depth cues and pseudoscopy. Stereo window violation is a problem that causes the depth perception to break. It creates sudden jumps in the perceived depth and sometimes an inability to fuse a pair of images into a single 3D image. This problem occurs with objects that appear to be floating in front of the movie screen. As these objects cross the edge of the screen the depth perception breaks. Details of the problem are discussed in Chapter 3. Stereoscopy generates the depth perception through the horizontal displacement in a pair of images. However, that is not the only cue that our visual system uses to perceive depth. There are

a variety of ways to create stereoscopic content. Stereoscopic content is sometimes created using 2D-3D image conversion as well as compositing. These methods aim to introduce stereo depth cues into the images, however if not done properly these artificially created depth cues may conflict with other monocular depth cues such as occlusions and shadows. Pseudoscopy is a type of content where a stereo image pair gets swapped and is presented to opposite eyes than the ones originally intended. This process reverse the depth perception and introduces conflicting depth cues. This is explored in later chapters.

1.2 Thesis Overview

This section discusses the structure of the thesis. Chapter 2 contains a literature review on relevant topics following the introduction in Chapter 1. The thesis itself is motivated by the usage of computer vision to help create better stereoscopic content. Therefore, the literature review first covers the processes used to create such content. Native stereo content creation through stereo camera set-ups as well as 2D-to-3D conversions are shortly discussed in the review. Reviewing the methods of creating the content itself gives us a better understanding on how to analyse them. Following that we look into some of the current work that has been done to solve or explore some of the problems in stereo movies. Stereo window violations as well as various conflicting cues are part of this review. From this we move on to topics within computer vision. Stereo correspondence is introduced and state of the art methods in the literature are explored. This is followed by the exploration of half occlusions, which are also an integral part of the computational stereo literature. They are not only present in all stereo content but they prove to be useful features in the analysis

of certain problems. Chapter 3 contains the detailed description of our proposed framework for detecting stereo window violations. The chapter begins with a formal description of the problem itself. An overview of our framework is discussed to gain an insight to the process. Each component of our process is then thoroughly discussed with contributions highlighted within. A small discussion follows the methodology to introduce the dataset that was created to test this methodology. The results are presented along with a discussion. Chapter 4 focuses on the topic of half occlusions. We begin with definitions of half occlusions follow by an overview of how they may be detected. The solution is tested on the Middlebury Stereo Dataset to make it comparable to other methods in literature. The discussion of the results explore the value of half occlusion in analysing other stereo problems. Instead of focusing purely on detecting half occlusions we explore the usefulness of half occlusions as a feature. Chapter 5 summarises all of the contributions made throughout along with closing statements to indicate future work.

CHAPTER 2
Literature Review

The aim of this thesis is to assist in the creation of stereo movies by providing tools to assess the quality of these movies during production or post-production. To do so, an understanding of the 3D film making process as well as an understanding of possible viewer experience is required. The proposed methods to achieve the objective of the thesis are based on computer vision techniques. Thus it is equally important to revisit previous work done in computer vision, specifically stereo vision, and obtain an understanding of that as well. This chapter of the thesis will review previous work on stereo movie creation, quality assessment of stereo movies, rectification and correspondence in stereo vision and half occlusions in stereo imagery.

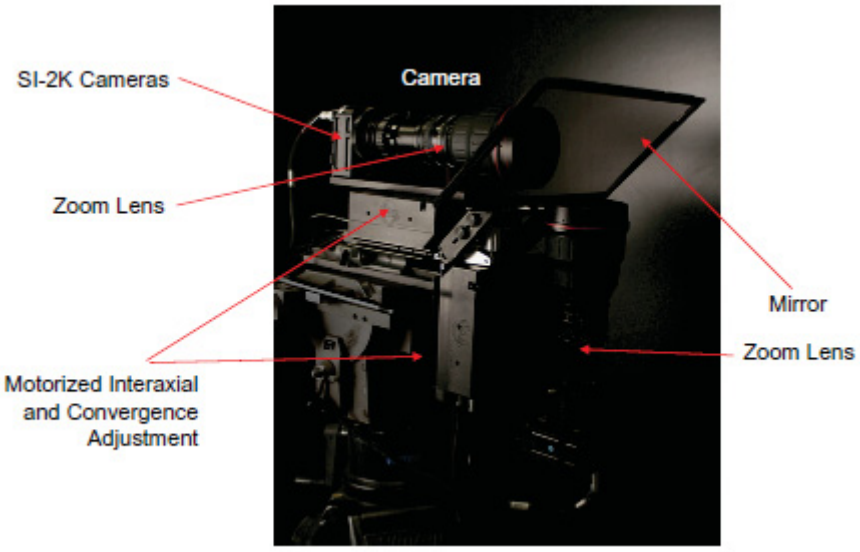
2.1 Stereo content creation

Stereoscopic content can be created for a variety of purposes. Ranging from short films in amusement parks to broadcasting sports and other TV content in 3D using stereoscopy. Regardless of the domain, the processes of creating stereoscopic content is very similar.

The most natural way of creating stereoscopic content is by shooting using a stereoscopic camera set-up. This involves capturing two sets of images as done by the human eyes. However, in practice the camera set-up is not an exact replication of the eyes. Not all set-ups are based on two cameras set up side-by-side to emulate the human eyes. Other common set-ups for filming 3D movies, as discussed in [32],

include having a single sensor with lens attachments to get different views, using a pair of lens mounted at 90 degrees to each other with the incoming light going through a half-silvered mirror angled at 45 degrees. Each set-up has its own pros and cons but the half-silvered mirror set-up, similarly to Figure 2–1 (a) is more popular for feature films. The single sensor configuration, as seen in Figure 2–1 (b) is not viable for the movie industry as image resolution is lost by attempting to create two view from a single sensor. The side-by-side setup, like the one in Figure 2–1 (c), works well for wide landscape shots. A parameter to take into consideration while filming a stereoscopic content is the interocular distance between the cameras, also referred to as baseline or interaxial distance. The interocular distance represents the separation between the two fields of view. A greater interocular distance creates a greater 3D effect, especially for close up scenes. Wide lenses allow capturing of more light and a wider field of view which can be valuable for visual and artistic purposes. Using a side-by-side arrangement with wide lenses typically results in too large of an interocular distance, limiting the artistic freedom to take close up shots. Along with the camera setup, a variety of other mechanisms go into commercial stereo-rigs with half-silvered mirrors, which allow for video stabilization, proper alignment and changing the interocular distance and convergence [33]. The ability to use wide lenses as well as being able to control all these parameters justifies its popularity.

Shooting with a stereo camera rig is not the only way of creating 3D movies. A film can be shot using a conventional 2D camera and then converted to a 3D movie. This is done by manually painting in the depth of a scene and then using it to shift the objects in an image to create a second view. It is not always necessary



(a) A half-silvered mirror camera

(b) A stereo lens attachment on a single sensor camera

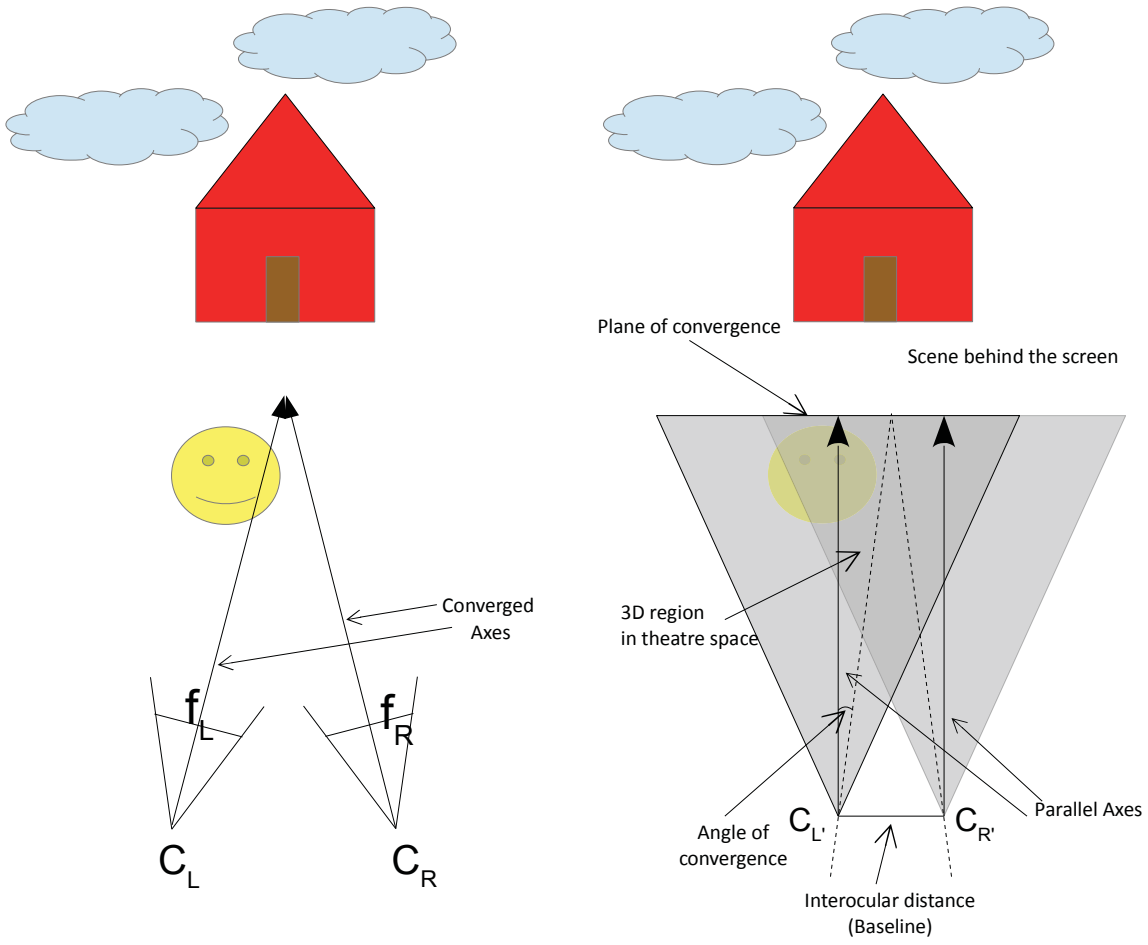
(c) A side by side camera

Figure 2–1: Examples of different stereo camera setups [33]

to fully paint in the depth as a lot of work has been done to assist the process such as providing sparse representations of depth as seen in [57]. Other methods aim to make the process even more convenient by allowing automatic conversion of 2D footage into 3D films. The work of Zhang *et al.* [61] shows one such endeavour that utilizes cues other than stereopsis to estimate the depth of a scene. Monocular cues

for estimating depth include geometric constraints, relative size, perspective, focus, etc. For videos, as is common for 2D to 3D conversions, motion parallax provides a strong cue for depth estimation. Once these monocular cues are used to get an estimate of the scenes depth then a similar process of shifting objects in the scene can be used to obtain a stereo pair. The conversion process introduces a problem of its own, which is the need to fill in the newly exposed background after shifting the foreground objects to create the stereo pair. Converted movies are still prone to the problems that occur in a movie natively shot using a stereo rig. Hence the discussions in this and the following sections of the thesis that address these issues apply to both forms of stereo content.

Understanding the geometry of the scene being shot will help our analysis of the potential problems. The simplest way to do so would be to start with a simple configurations of two cameras with a fixed interocular distance. The two cameras can either be parallel or converged depending on the axis of the camera from its optical centre to the focal point. The convergence determines the distance at which the fields of view of the two cameras align. Objects at this distance from the camera have zero parallax. Cinematographers often prefer to coincide the screen to match zero parallax. This makes it easier to frame scenes and monitor the depth budget. Any object nearer to the camera, compared to this point, will appear in front of the screen. Objects further than that point will appear behind the screen [33]. Therefore, convergence plays a key role in determining how the 3D effect will look to viewers alongside the interocular distance. Figure 2–2 (a) shows a simple representation of a scene being shot using a pair of cameras.



(a) Scene representation of a converged camera setup(b) Scene representation after image rectification (equivalent to a parallel setup)

Figure 2–2: Stereo scene geometry

Regardless of the cameras convergence, the images are put through a post-processing step that performs image rectification. This rectification process provides

many benefits to good content creation, details of which are further discussed in Section 2.3. However, it is important to note that the geometry of the image formation process, after the rectification, is equivalent to that of images acquired using a parallel camera setup. A parallel setup essentially has a point of convergence at infinity. Therefore, everything being shot in a scene would appear floating in front of the screen. This could be a painful experience as well as a poor choice artistically. For this purpose a technique known as Horizontal Image Translation (HIT) is used in post-production to alter the point of convergence for parallel setup and also for a variety of other purposes [8]. Figure 2–2 (b) shows a simplified representation of the scene after rectification.

2.2 Problems in stereo

2.2.1 Stereo Window Violation

Window violations are one problem that is very common to stereo cinematography. It occurs when an object appears hovering in front of the screen and moves in a way such that it crosses the boundary of the screen. When such an event takes place, the depth perception of this object is distorted. The distortion takes place in the form of sudden jump in depth perception as well as an inability to properly fuse a 3D image in the boundary regions. The depth jump causes the object crossing the boundary to appear pushed back to the depth of the screen. The problem itself is caused by the limitation of having a screen and its boundary at a fixed depth. When the object crosses the screen, it is cut-off due to the occluding boundary. This can be seen in Figure 2–3 (a). The occlusion cue is stronger and eventually forces the depth perception of the object to be pushed back to the screen level. The

boundary, itself, and regions beyond it do not have any disparities and that is the reason why the depth perception is pushed back to screen level where there are no disparities. Without careful consideration of scene structuring it is very easy to generate a violation. Repeated occurrences of this violation is neither comfortable for viewing nor is aesthetically pleasing. Therefore, cinematographers work constantly on fixing occurrences of window violations or altering scene structuring to avoid such violations.

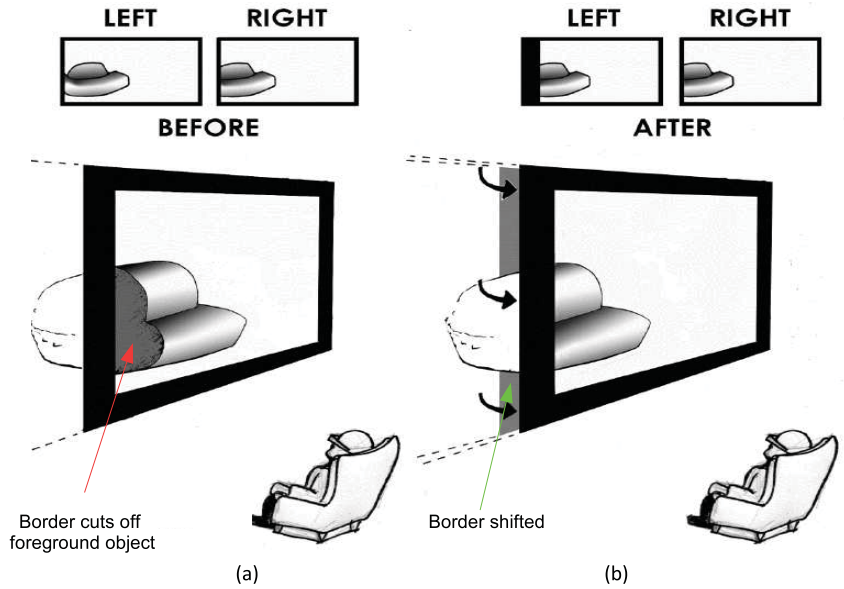


Figure 2–3: (a) Scene with a stereo window violation where the border cuts off an object in front of the screen. (b) Example of using a dynamic floating window to add disparity to the borders and shift it forward in depth. Stereo window violation is avoided by matching the border's depth to that of the foreground object or making the border appear even closer to the audience than the foreground object.[18]

In literature, a common solution to solving the stereo window violation problem is to use a floating stereoscopic window [32, 18, 36]. Stereo window violations are also related to half occlusions as they are caused by monoscopic regions near the edges of

the screen. A portion of the image can been seen only on one view but not the other. The part not visible occluded by the border of the screen. For content appearing in front of the screen, as in the case of stereo window violations, the left view would have the monoscopic regions near the left boundary and the right view would have its monoscopic regions near its right boundary. By introducing the floating stereoscopic window, we change the depth of the screen borders by adding extra black pixels to these monoscopic regions. By doing so, there are no mismatches between the objects in front of the screen. The border itself is given some disparity resulting in a change in its depth perception. The objects in front will now be occluded by a border which is at the same depth or nearer than them in appearance. Figure 2–3 (b) shows an example of using floating windows to correct for stereo window violation. There are variations in how the floating window is defined. It could be established in advance where a fixed portion is cropped out to make the borders to be the nearest object in the scene. Alternatively, it can be dynamically adjusted for different scenes based on the scene structure and geometry. Even though there are ways of solving the problem in literature, there is little work on detecting stereo window violations without manual input. The floating windows themselves would need input from a stereographer to be properly established. The cropping introduces a loss of content that may be vital to the storyline of the movie. Therefore having an automated detection system would add to the capabilities of producing quality stereo movies with greater ease.

2.2.2 Conflicting Depth Cues

A major issue when creating stereoscopic content is ensuring there are no conflicts with other cues related to depth perception. Monocular cues such as colour, occlusions, texture, focus, etc. can create a strong depth perception [31, 38, 37, 53]. Even if our brain is able to fuse the stereo pair, having conflicting cues will hinder its ability to perceive depth properly. A common manifestation of occlusion-stereopsis conflict can be observed in the case of stereo pair reversal (pseudoscopy). This happens when the left and the right views of the stereo pair get swapped for one another. The situation can be mimicked by swapping the lenses in 3D glasses during viewing. Swapping a stereo pair that has been compensated for distortions and ensured to have an acceptable disparity range will still allow us to fuse the pair into a single 3D image. However, our ability to perceive depth will be hindered as the brain receives conflicting signals. The swapped parallax of objects that are supposed to be all the way in the back would now indicate that they are to be perceived hovering in front of the screen. Similarly, objects in front will have a parallax that will force them all the way behind the screen. However, the object, that is originally supposed to be in front, still occludes the background. This creates a confusion in our brain which jumps back and forth between different cues. The confusion can lead to fatigue for our visual system and worse lead to a wrong perception of depth. Since, the depth perception still exists users may be unable to tell that the stereo content is faulty. But long exposure to such content will not yield an enjoyable experience as the fatigue may lead to nausea or headaches. A similar conflict may arise when performing

conversions. In a reversal, the entire image is at risk of the occlusion-stereopsis conflict but in the case of conversion only a portion of the image may be affected. Some conversions are done manually while the depth is hand painted in. This is an expensive and time consuming operation and not usually done unless the highest quality is required. With the lack of true 3D content availability, 3D television vendors may opt to use some of the available real time conversion techniques to create content and generate popularity for the televisions. These real time conversions are usually not of the highest quality and the depths across the entire image may not be estimated accurately leading to conflicts. Such a risk also exists when doing compositing for 3D scenes. Compositing is the process of adding new elements to an existing scene. This can include adding a different background on an existing scene or adding a character into the scene using computer generated imagery. Both of these processes are also prone to conflicts in colour cues from the two views. Through the conversion and compositing processes, features such as shadows may not be recreated perfectly in both images leading to difference in colour for stereo correspondences. The mismatches can make depth perception more difficult.

2.3 Computational Stereo Vision

Computational stereo vision concerns itself with the analysis and extraction of information from binocular pairs of images (obtained by stereo cameras or conversion). In the literature, the pairs are usually referred to as left and right images (because they are captured by left and right cameras).A lot of the techniques can be extended to more than two images (multi-view stereo), but this thesis focuses only on binocular stereo only due to the nature of our target problem. There are two

main problems in the domain of stereo vision. One is the problem of stereo correspondence. Since we have a pair of images of a scene acquited from two different viewpoints, it is of interest to be able to match points from one image to another. The second problem under stereo vision is that of reconstruction. The reconstruction problem aims to resolve the 3D structure of a scene given correspondences and the geometry of the stereo system used to obtain the stereo pair. This means being able to generate a 3D map of a viewed scene. For our purpose, solving the correspondence problem is of primary interest and is explored further in this literature review [54].

The primary difference between the pair of stereo images is the shift in location, of objects, from one image to another. This shift is commonly referred to as disparity in computer vision literature and as parallax in 3D cinematography literature. The two terms might be used interchangeably across this thesis with disparity being used for computer vision techniques and parallax being used for cinematic/perceptual contexts. Estimating the disparity map of a scene allows us to interpret the depth of objects. Solving the correspondence problem is not a simple task, with much of current research being done to improve the accuracy and efficiency of the solutions. Issues such as regions being present in one view and not the other, lack of textures making matching difficult and other factors make solving the correspondence problem challenging. However, a variety of solutions exist in the current literature to form the foundation of our solutions to the target problems in 3D movies.

2.3.1 Overview of the Stereo Correspondence Problem

The literature consists of a variety of methods that address the correspondence problem. Even though the specific implementations vary greatly, the broad structure

for each of these methods remain similar. The final goal of the solutions is to estimate a binocular disparity map.The first step in the process is to perform rectification. To establish these correspondences, each pixel in the right image has to be matched with a pixel in the left image. The matching can be done by leveraging local appearance information such as color, texture, etc. However, for each pixel in the left image, searching for a corresponding match across the two dimensions of the right image is a computationally taxing process. However, leveraging the properties of the epipolar geometry, guiding the stereo setup, we can reduce the search space to one dimension [49].

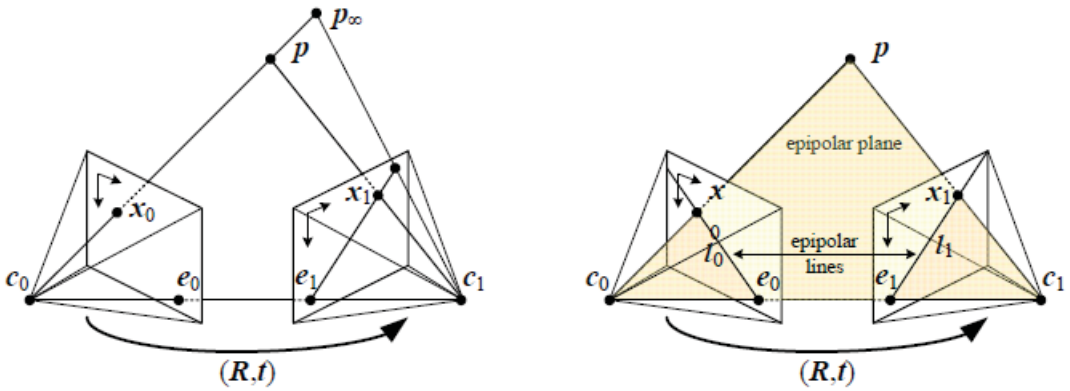


Figure 2–4: Epipolar geometry for binocular stereo [49].

The projection of points in 3D space onto the image plane lies along the epipolar line. This is referred to as the epipolar constraint. In order to match pixels from the left image to the right, we need to search only along the epipolar lines. The epipolar geometry is defined relative to the camera pose and calibrations. This geometry can be estimated from seven or more corresponding points and with the use of the fundamental matrix. The fundamental matrix is a 3×3 matrix that encodes the extrinsic

and intrinsic information of the cameras. The extrinsic parameters contain information about the transformation from the 3D world coordinates to the 3D camera coordinates. These parameters are the rotation matrix and translation vector. The intrinsic parameters encode information such as focal length, pixel size of the sensor [54]. Rectification is the process of warping the pair of images such that the epipolar lines align with the horizontal axis of the image. It is ideal to perform rectification on images obtained using cameras whose extrinsic and intrinsic parameters are known. In order to obtain these parameters a calibration is performed. Multiple views of a known pattern type are captured which can be used to estimate the required parameters. The outcome of the process also provides the coefficients of radial and tangential distortions which are used to model the effect of distortions in real lenses. Using all the parameters from the calibration, the effects of distortion are removed and the images are warped to ensure the stereo pair differ only in horizontal disparity. Popular toolboxes exist in Matlab, as well as in OpenCV, that perform accurate and efficient calibration [4]. The work in [17] and [47] discuss in further detail the process of rectification using images obtained by calibrated cameras. Rectification can also be performed on uncalibrated images but the process is more complicated [29]. Since the calibration is a one time process it would be ideal to perform the calibration before setting up the scene for shooting and then performing rectification on the calibrated images. The geometry of images obtained, after rectification, is equivalent to that of images acquired with a parallel camera setup. This is a property that will be explored later in the thesis. Calibration and rectification forms the first step for all correspondence algorithms. It also helps define the relationship between

the disparity and scene depth quite easily as seen in Equation 2.1.

$$d = f\frac{B}{Z} \tag{2.1}$$

From Equation 2.1, d is the disparity, f is the focal point of the camera, B is the baseline distance or the interocular distance mentioned in previous sections and Z is the depth. Hence we can see there is a inverse relationship between the disparity and the depth. Because of this very simple relationship the depth map and disparity map are often used interchangeably in the literature when finding the solution to the correspondence problem.

Assuming we have a pair of rectified images, the next step would be to establish a cost function. The cost function captures how well the corresponding points match each other. The final step involves minimizing the cost function to obtain the disparity estimate. Scharstein and Szeliski explore the general structure of such algorithms in their work [42]. Based on their work we learn how the computation, as well as optimization, of the cost function can be local or global. The quality of the solution depends on the specifics of defining the cost and its optimization to obtain the disparity. The following subsection explores some specific stereo correspondence algorithms.

2.3.2 Local Stereo Correspondence

Local stereo algorithms make use of pixel-based matching costs. The nature of the cost functions either form a similarity or difference measure around a local window. The simplest functions utilized in these local window based approaches include Sum of Squared Differences (SSD), Absolute Difference (AD), Normalized

Cross-Correlation (NCC) as seen in [22, 27, 40]. The three measures discussed are dependent on the color information in the pair of the images. The varying gains in the two cameras might cause differences in local color appearance, hence there exists similarity measures that aim to match the gradients of the images[41].

Regardless of the measure being used, we want the cost function to represent the cost of assigning a disparity d to a pixel p in an image. Hence, the cost function is defined over all pixels in an image across the range of possible disparity values. Equation 2.2 shows the definition of a cost function using the sum of squared differences over a local window.

$$C(x, y, d) = \sum_{x,y \in N} [I_L(x, y) - I_R(x, y - d)]^2 \tag{2.2}$$

The cost is calculated over a neighborhood N around the pixel location (x, y). The size of the neighborhood is determined by the window size and the neighborhood is typically a rectangle centered on the pixel in question. The squared difference is calculated using a pixel in the left image and the pixel in the right image that is horizontally shifted by d pixels from the left pixel's location. If the range of disparity includes positive and negative values the horizontal search can be on either side of the pixel location (x, y). The significance of the sign of disparity is discussed in Section 3.1. By calculating the cost over this local window, this method implicitly performs an aggregation of the costs. Some methods do the aggregation step separately where the cost is calculated over pixels only and then aggregated based on some required criteria. The aggregation step or cost computation over a window helps reduce the effect of high frequency noise and produces smoother cost volumes. Once this cost

volume is calculated the disparity map can be obtained by following a Winner-takes-all(WTA) scheme. In such a scheme, for each pixel location, the disparity value d that has the lowest cost is assigned to that pixel. If the cost function were to use normalized cross correlation as a measure then the disparity with the high cost (similarity in this case) would be assigned to the pixel. Using bigger window sizes leads to smoother results while sacrificing the resolution of the disparity estimates. Local methods are simple to implement and optimize. Hence, they can produce disparity estimates quickly. The usage of multiple simple operations in computing cost and optimization allows for parallelization of the process.

2.3.3 Global Stereo Correspondence Algorithms

Even though local stereo algorithms are simple and quick to implement they have certain disadvantages that results in less accurate estimates. The window is generally rectangular in shape around the pixel. The size of the window is decided upon at the beginning and remains fixed across every pixel location. Windows that are too large may fail to capture finer details in the images, while windows that are too small will have difficulty matching larger structures in the scene. Hence an alternate approach to stereo correspondence arose in the form of global algorithms.

Global algorithms differ from local methods in terms of their definition of cost functions as well the optimization. The correspondence problem is formulated as a labelling problem in the global framework. The goal is to assign a label to each pixel in the image, the label in this case being a disparity value. We have a discrete set of points that need to be assigned a label from a discrete set as well. A Markov Random Field (MRF) is ideal to model such a scenario. When using an MRF, the

cost function is represented using an energy function. The energy function consists of a data consistency term (which is similar to cost functions defined in local methods) along with a smoothness term. The smoothness term is what sets the global approach apart as it ensures neighbouring pixels have similar labelling even if noise inflates the data cost at certain pixels. It is also not restricted within a square region around the pixel. A general energy function for such an MRF can be seen in Equation 2.3.

$$E_{total} = E_{data} + \lambda E_{smooth} \tag{2.3}$$

A regularizing parameter,λ, governs the contribution of the data consistency term versus the smoothness term in establishing the label. Once the energy function has been established minimizing this function would yield the optimum labelling (disparity map).

The formulation of energy functions for problems and details about its implementations are discussed in [5, 28, 50]. The pixels in the image are represented as nodes of a graph that are connected to each other in a grid form. A range of possible disparities are chosen as the set of labels. The data term correspond to the likelihood and it is the cost of assigning a disparity to a pixel. This cost measure is often similar to costs used in local methods as can be seen in Equation 2.4.

$$E_{data} = \sum_{p} D_p(l_p) = \sum_{p} [I_L(x, y) - I_R(x, y - d)]^2 \tag{2.4}$$

The data cost is represented in the graphical MRF by a link from the label to the node. The smoothness cost is represented by the links between neighbouring pixels in the regular grid lattice that forms the MRF. The smoothness cost, $V_{pq}(l_p, l_q)$

is defined for the label of pixel p at location (i, j) and pixel q at location (s, t) that is a horizontal/vertical neighbour of the pixel p. The neighbourhood around p is represented by $\mathcal{N}$. The authors in [50] show the Potts model as a useful way (especially for stereo) to represent the smoothness cost as it penalizes any different pairs of labels uniformly. The smoothness cost using the Potts model is shown in Equation 2.5.

$$E_{data} = \sum_{p,q \in \mathcal{N}} V(\Delta l) = \sum_{p,q \in \mathcal{N}} min(|\Delta l|^k, V_{max}) \tag{2.5}$$

with $k = 1 or 2$ and $V_{max} = 1$. There are a variety of optimization techniques that allow us to minimize the energy function of these models and lead to solutions quite close to the global minimum. Boykov *et al.* [6] mention some of these techniques in their work before introducing their own contribution of using the graph cut method.

One approach for achieving local minimum of the energy function is the Iterated Conditional Modes (ICM). This is a greedy technique where each pixel is assigned a label that results in the largest reduction in energy until a convergence to a local minimum is achieved. Simulated annealing is another method for minimizing the energy. It gained popularity due to its ease of implementation as well as its ability to minimize any arbitrary energy function. However, it takes exponential time to run and hence is too slow to be used for practical implementations. When implementing a faster annealing for practical applications, the resulting minimization turn out to be very far from the global optimum. Boykov *et al.* proposed two algorithms based on graph cuts that provide a faster minimization of the energy and achieve results closer to the global optimum.

The two main algorithms introduced in [6] are the α expansion and the $\alpha - \beta$ swap. The α-expansion and the $\alpha - \beta$ swap allow a large number of pixels to update their labels simultaneously contrary to the previous methods mentioned. In an α-expansion step, any set of image pixels can change their label to be α. In an $\alpha - \beta$ swap, pixels previously labeled α are assigned a new label β or the reverse in some cases. These two basic steps form the core of the α-expansion and the $\alpha - \beta$ swap optimization algorithms. Given a MRF configuration, there is an exponential number of possible expansion and swap moves, therefore a naive implementation of the algorithm would require exponential time. An efficient method of choosing the appropriate expansion and swap moves can be found by the graph cut method. The MRF graph is connected to source-sink terminals. Each pixel is connected to a label in the source and a label in the sink. The weights between these connections represent the cost of assigning a particular label for that site (imposing data consistency). The pixels are also connected to each of its neighbours and the cost of assigning different labels to adjacent sites forms the weights for these connections (imposing smoothness). For the α-expansion algorithm the source label is chosen as α while the sink represents all labels that are not α. For $\alpha - \beta$ swap, the source label is α while the sink is β. By finding the minimum-cut/max-flow [5] of this graph, the energy function is reduced. This cut breaks the graph into two disjoint sets with sites connected either to the source or the sink. The labels of the pixels are updated based on which label (source or sink) they are connected to. Pixels connected to the not-α label retain their old labels instead of being assigned a new one. This process is repeated for each α label or each $\alpha - \beta$ pairs respectively until a minimum cut cannot

be found that results in a minimization of the energy. The advantage of using such graph cut methods, for global optimization, is that the αexpansion can be performed in linear time and the $\alpha - \beta$ swap in quadratic time (in practical implementations can often be done in linear time). The drawback for these optimization algorithms is that they do not work as well for other generalized priors that exert a piece-wise smoothness assumption.

2.4 Half Occlusion

As mentioned earlier stereo correspondence can be a difficult problem. Difficulties in correspondence can be due to a lack of texture or the presence of significant blurring. This loss of detail makes it difficult to match one view to another. Other factors such as varying gains among cameras, existence of specularities in one view or another or vertical alignment problems can all make correspondence difficult. However, sometimes correspondence is not only difficult but impossible because a matching does not exist. This is the case in the presence of half occlusions. The presence of half occlusions cannot be avoided as it is inherent to the stereoscopic content creation process. Occlusion is usually caused by an object appearing in front of another and thus making it hidden from view. In a stereo setup, there are two views with different perspective and so the portion of the image which gets hidden is different in each of the view. Therefore, one image contains a portion of the occluded object which does not appear on the other. Since these regions appear exclusively, they do not have any matching to the other view, making correspondence impossible. With this realization, we can understand that certain pixels will not have true correspondences and leverage the knowledge of half occlusion to yield other information

such as object boundaries, depth discontinuities, etc. The following sections provide a review of work done on detecting half occlusions.

2.4.1 Metrics to assess half occlusions

As mentioned before, half occlusions lead to poor stereo correspondences around those regions. A range of approaches have been used in literature to address the issue of half occlusion. Some methods aim to find the goodness of match from the correspondence. The quality of match is then used to make a decision on the presence of half occlusion. These methods aim to evaluate the quality of the matches by looking at each pixel, of an estimated disparity map, and its neighbours. Most of these methods begin by first evaluating the disparity map with the existence of noisy estimates around the half occlusion regions. Egnal and Wildes [15] compiled and compared a few methods that try to estimate half occlusions. One such way involves evaluating the quality of the disparity map by looking at the modality of the disparity histograms in a horizontal line around a pixel [30]. The method proposes that around regions of a half-occlusion boundary we would have disparities of the occluding surface and the occluded surface. This would lead to a bimodal distribution of the horizontal disparity. The measure for bimodality was computed as the ratio of the two local peaks [45]. The ratio is defined as

$$Bimodality = \frac{maxD_1}{maxD_2} \tag{2.6}$$

where D_1 and D_2 represent the largest and second largest peak in the disparity histogram.

Another method looks at adjacent regions of good matches and bad matches to detect the presence of half occlusions. The goodness in match is going to be high until a half occlusion is encountered where the measure will fall. This jump in match goodness can be a useful metric in finding half occlusions [44]. The error to check for Match Goodness Jump is defined as

$$Error = max\left(\bar{C}_x - \bar{C}_{x+w}, \bar{C}_x - \bar{C}_{x-w}\right) \tag{2.7}$$

where x is the horizontal coordinate of the pixel, $\bar{C}$ is the summed(aggregated) matching cost within a window size w.

The previous two metrics are a quality assessment of the disparity estimate. Finding half occlusions using those metrics alone would not be ideal as the quality of the disparity may be influenced by factors other than half occlusions. They do not incorporate any specific knowledge about the nature of the half occlusion problem. The two views of a stereo pair only have a slightly varying perspective where most parts are similar, except the half occlusion regions. Therefore, the disparity estimate in the left image (left to right matching) should be negatives of the disparity estimate in the right image (right to left matching). The left/right checking [58, 10, 52] utilizes this information to check the left and right disparity estimates. Pixels failing this checking are to be labelled as half occlusion. The error criteria to check if the left and right disparity estimates have matchings is as follows

$$Error = x_R - (x'_L + d^L_{x'_L}) \tag{2.8}$$

In Equation 2.8, x_R is a pixel in the right image that matches $x'_L = x_R + d^R_{xR'}$, x'_L being the estimated matching location in the left image based on the right horizontal disparity $d^R_{xR'}$. Using the knowledge of the half occlusions we can constrain our solution. Half occlusion regions do not have a true match but regions with true matches must follow their respective ordering. A pixel to the left of a point in the left image would also appear to the left of the same point in the right image. The false matching caused by half occlusions can cause this constraint to be violated [60]. The ordering constraint can be quantified as

$$Error = max(0, d^R_{xR} - d^R_{x"R}) \tag{2.9}$$

where x" represents the rightmost match acquired thus far. The final metric is the occlusion constraint [26, 19]. The principle behind this method is that near object boundaries the disparity jumps from the occluder to the background. This jump leads to a jump in the matching point in the opposite image as well leading to a group of unmatched pixels in between which can be labeled as half occluded pixels. This measure is similar in concept to the match goodness jump but instead of considering the cost, we look at jumps in the disparity estimates instead. The occlusion constraint can be enforced by using the following error function

$$Error = max(0, d^L_{xL+1} - d^L_{xL}) \tag{2.10}$$

where d^L_{xL+1} is the disparity of the pixel to the right of d^L_{xL} in the left image. Each of these metrics are simple low level methods of identifying possible half occlusions

regions. One or more of these quantities may be incorporated together to form high level solutions and make final decisions on half occlusion.

2.4.2 Bayesian approach to Half Occlusion Detection

Many stereo correspondence algorithms operate without using any information on half occlusions. Some of them, utilize metrics similar to those seen in Section 2.4.1 to perform post-processing corrections. However, Belhumeur and Mumford [3] incorporate half occlusions into their process of finding stereo correspondences. Their algorithm is designed with a global optimization in mind. They take a Bayesian approach to the problem by defining probabilities for the data term and prior probabilities for smoothness. The data term represents the costs of matching pixels from the left to right. The probabilities are then converted to an energy function similar to what we have seen in Section 2.3.3. However, their formulation of the final energy function is different from the one previously discussed. The major distinction is the inclusion of half occlusions into the energy function. Two boundary energy functions implemented via line functions in horizontal and vertical direction are also introduced into the overall energy function to better define object boundaries. Edge information is used to guide these boundary energy functions. The half occlusion is not only included as a new term in the energy function but it also influences the data term of the energy function. An initial estimation of the possible half occlusion regions is obtained using one of the metrics seen from 2.4.1. Belhumeur and Mumford utilize the ordering constraint to obtain their estimate of the half occlusion. The number of half occlusion pixels are counted as a cost towards the occlusion energy term. Additionally, the half occlusion pixels are not included when computing the cost for the

data term. Since, half occlusions do not allow for proper matching, avoiding them in the data matching term would allow for more accurate cost representations. The smoothness and boundary functions are extended to two dimensions including both the horizontal and vertical directions. The energy function is then minimized using a combination of dynamic programming and simulated annealing.

CHAPTER 3
Detection of Stereo Window Violation in 3D Movies

This chapter of the thesis formally defines the problem of stereo window violation in 3D movies. It is followed by the proposed framework and detailed discussion of each component that allows us to detect such problems in 3D movies. Lastly, the experimental setup used to test out framework and validate its performance is discussed.

3.1 Stereo Window Violation

To understand the problem of stereo window violation we must first discuss the concept of a stereo window. The stereo window, as used in the context of 3D cinematography, corresponds to the binocular stereo window that is formed when using two cameras. A similar concept applies to movies that are converted from 2D to 3D since a pair of images are created for that purpose to simulate shooting using two cameras. In a two camera setup each camera has its own field of view. Regardless of whether we use a side-by-side configuration or a mirror-rig configuration the cameras each capture a different viewpoint of the scene. However, these fields have overlapping regions. Within this overlapping region is a plane where the two camera's fields of view converge. This point of convergence can vary based on how the cameras were setup (converged versus parallel) and by the usage of horizontal image translation to alter the convergence. The plane of convergence defines the stereo window where the disparity or parallax is zero. This stereo window plane is made to coincide with

the movie screen in a theatre [32, 33]. Figure 3–1 illustrates the stereo window for a simple two camera setup with a slight toe-in for convergence.

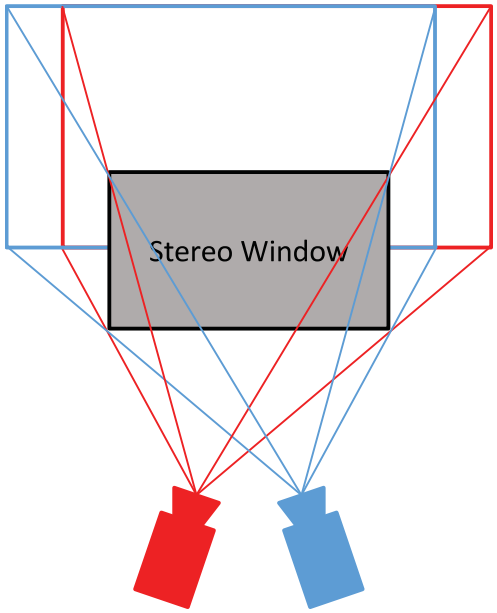


Figure 3–1: Two cameras with a slight toe-in with their fields' of view and convergence plane defining the stereo window

To tackle stereo window violation we also need to understand how objects appear in the theatre space. The parallax in the scene dictates where each object will appear with respect to the screen. This effect is purely perceptual as all of the content is being displayed on the screen itself. The perceptual effect may alter slightly based on the seating location of the viewer. However, we analyze the problem from the perspective of a viewer seated at a location for best viewing experience. As the content is displayed on the screen our eyes focus (this process is also called accommodation) upon it while it retains the ability to independently converge on something else. This ability to decouple the focus and convergence is inherent in

most humans, without which 3D movies as we know it today would not be possible. As mentioned earlier, the movie screen coinciding with the stereo window has zero parallax. Therefore for objects appearing at the screen depth, the pixel location (x, y) in the left image I_L is matches to the pixel location (x, y) in the right image I_R. But when the parallax is negative, the pixel location (x, y) in left image I_L matches to the pixel location $(x - |d|, y)$ in the right image I_R, where $|d|$ represents the amount of parallax or horizontal disparity. Similarly, for positive parallax, the pixel location (x, y) in left image I_L matches to the pixel location $(x + |d|, y)$ in the right image I_R. The majority of current theatres use polarization techniques to superimpose both images on the screen which are then filtered through glasses providing individual views to each of our eyes. For negative parallax, matching points in the right image occur to the left of the point in the left image. Our right eye looks at the right image and our left eye looks at the left image. For images with negative parallax, our eyes converge in front of the screen creating the perception of objects floating in the theatre space. The same concept applies for positive parallax, making objects look like they are behind the screen. Figure 3–2 shows a schematic illustration of objects with positive and negative parallax and how they would appear perceptually in the theatre space.

Looking back at Figure 3–1, we can see there are some regions where the fields of views for the cameras do not intersect at all. Any object falling within those regions are only visible to one of the two eyes. This mismatch causes our perception to alternate between the stimulus from the left image and the right image leading to difficulties in inferring depth and forming a 3D image. This effect is called

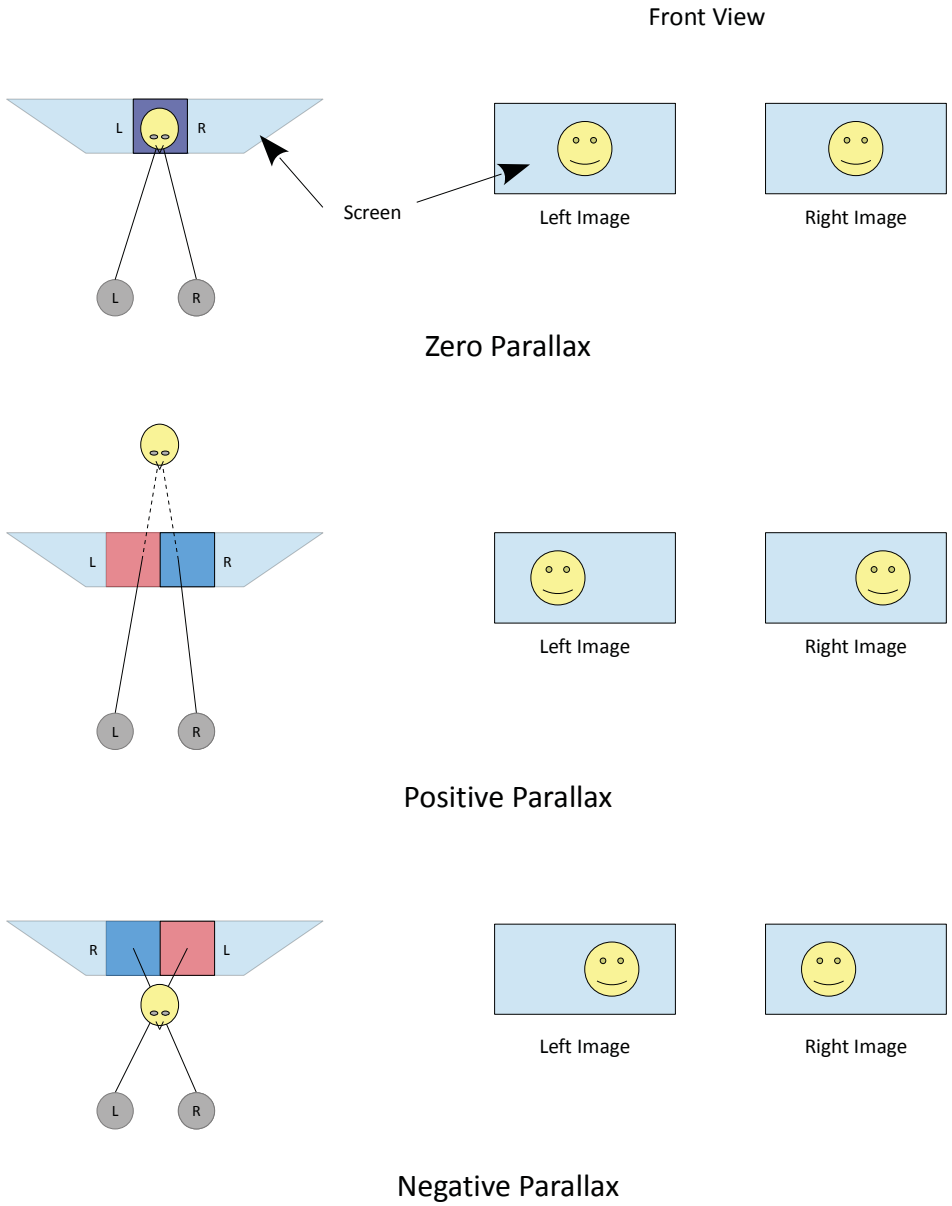


Figure 3–2: Different types of parallax and their perception with respect to the screen

retinal rivalry in literature [48]. Retinal rivalry is undesirable in general, but even more so when it occurs in regions in front of the screen. Stereo window violation is linked to retinal rivalry for objects appearing in front of the screen. When an object having negative parallax, appearing in front of the screen, moves in way such that it crosses the boundary of the stereo window (the screen) it results in a stereo window violation. An illustration of this situation is given in Figure 3–3. As the object crosses the screen a portion of it is lost to one eye. This effect can also happen to objects appearing behind the screen as well. However, for the former case an additional issue is the fact that the object in front of the screen gets occluded by the screen boundary. This is an unnatural effect for the human visual system and makes it difficult for the brain to reconcile this information leading to an uncomfortable viewing experience. Our eyes are unable to fuse the two images together, resulting in a sudden depth jump. This jump is caused by a stronger conflicting occlusion cue of the screen boundary, resulting in the object appearing at screen depth [2]. These sudden depth changes can be taxing for the visual system and also ruin the aesthetics of the scene. If the stereo window violation is caused by a moving person this artefact may cause his arm, crossing the boundary, to appear at screen depth whereas the rest of him appears floating in front at its original location. This artefact would completely ruin the immersive experience of the 3D movie. To avoid visual fatigue and to retain the visual aesthetics of the movie it is important to identify scenes containing stereo window violation and eliminate them.

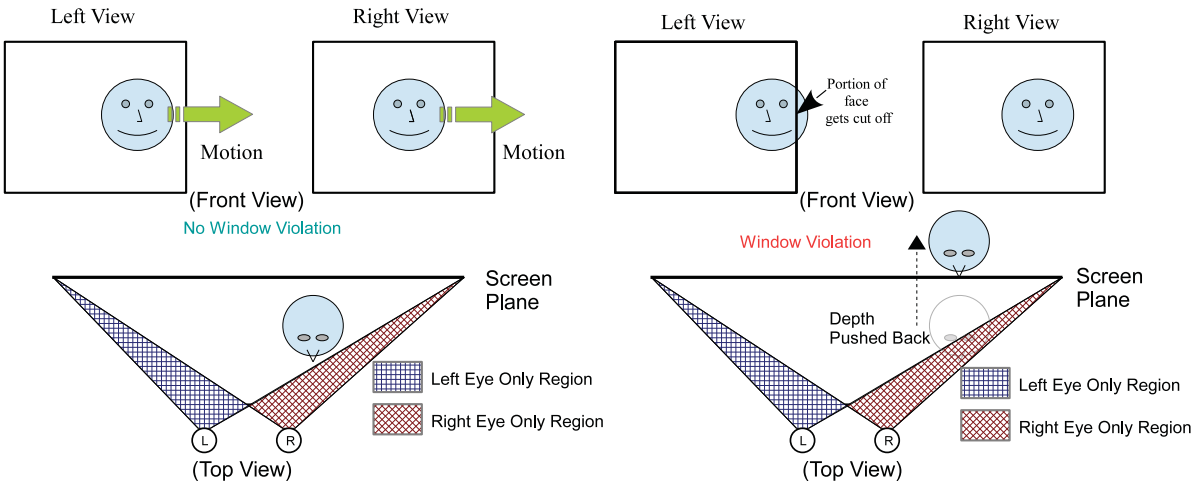


Figure 3–3: Schematic representation of stereo window violation

3.2 Proposed framework for the detection of stereo window violation

This section of the thesis gives a broad overview of the components in our proposed framework for detecting stereo window violation. Each component is then discussed in detail to demonstrate how they fit together to form the detector.

3.2.1 Overview

Section 3.1 gives us a good platform to build our solution as it helps understand the exact nature of the problem. Stereo window violation is a problem that occurs for objects appearing in front of the screen. Our aim is to provide a detection system that flags each frame that contains such a violation or maybe in proximity of having such a violation. To know which objects are in front of the screen, we first need to do estimate the depth of the scene. The first step of our solution is to compute a dense disparity map using a binocular stereo correspondence algorithm defining disparities over positive and negative parallaxes. After obtaining a dense disparity estimation,

we do not want every pixel that is in front of the screen to be a candidate for causing stereo window violation. Since our target problem is for movies, we propose using the focus cue to find regions that could cause stereo window violation. Focus is used as a very strong cue by cinematographers to direct the attention of viewers to certain part of the scene. This is something movie-goers are accustomed to, as they immerse themselves into the movie experience. So, we can ignore a character out of focus going off screen because he is not essential to the story and is not meant to be looked at by the viewer. Having these two independent estimates still gives us many pixels with certain disparities and a certain amount of focus. Ideally we wish to be tracking objects in the scene. We need to establish coherence between pixels based on their disparity and focus estimations. Hence, the final step involves clustering the pixels using the disparity, focus and the original image intensities as features to find segmentations of objects. These objects can then be tracked to monitor for stereo window violation. Figure 3–4 shows a flowchart representation of the components in the detection process.

3.2.2 Depth estimation from binocular disparity

A variety of disparity estimation algorithms exist in the literature and they have been categorically discussed in Section 2.3. The aim of our complete solution is to provide on-set assistance for cinematographers or people working on 2D-to-3D conversion. Therefore, along with providing an accurate depth map we would also like our method to be as efficient as possible. Local methods are generally simpler methods requiring minimal effort in the optimization phase and are generally highly parallelizable. But the quality of their estimates are usually far off from globally

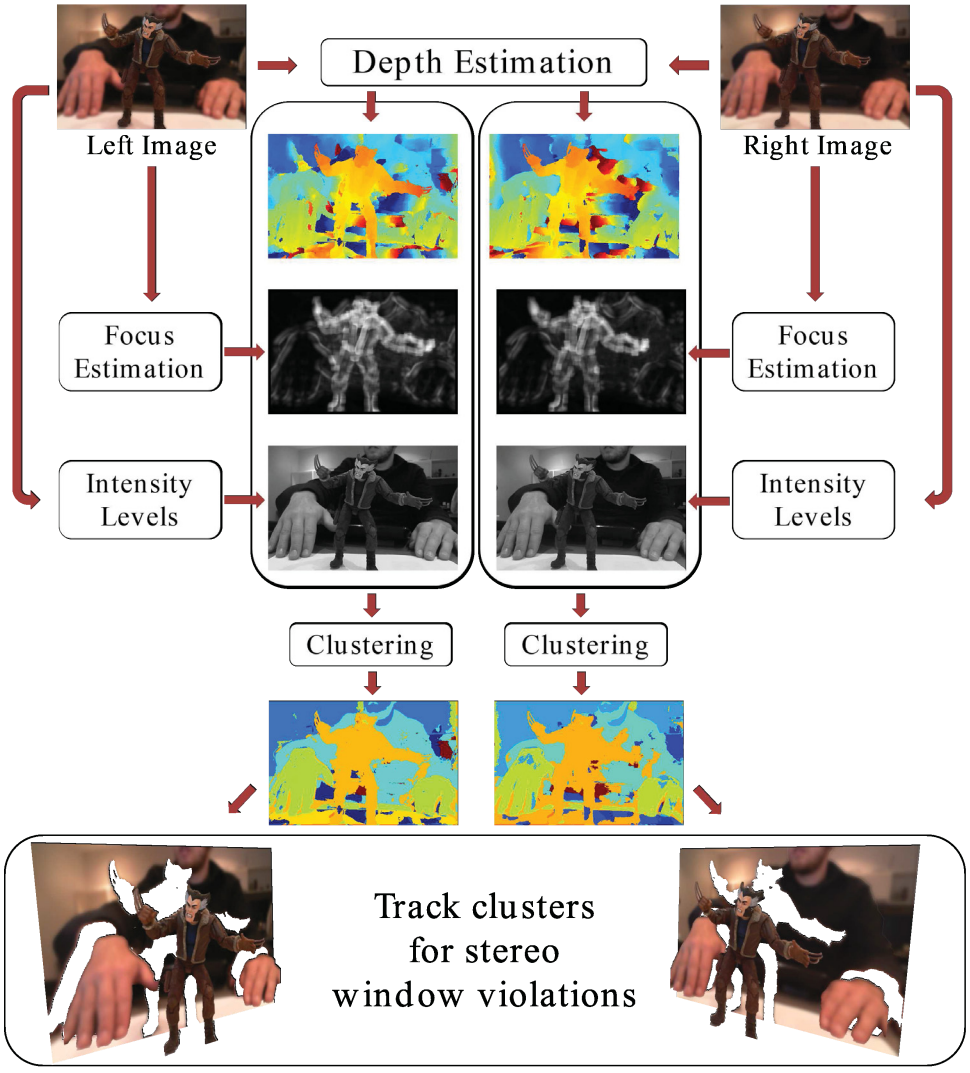


Figure 3–4: Flowchart of our proposed methodology

optimized methods. The results tend to be either "blocky" or "unsmooth" based on the window size as cost functions are defined and aggregated over rectangular windows.

An aggregation or filtering operation that does not uniformly filter across the rectangular window is preferable. This would allow us to retain the simple and parallelizable approach of local methods while the results would be more accurate as the variation within the rectangular block can be captured in a non-uniform filtering. He *et al.* introduced a novel filtering operation called the guided filter in [23]. The guided filter has a couple of useful properties. Firstly, the filter operation is edge preserving as well as gradient preserving. The edge preservation property is derived from using the original intensity image to guided variable weighting across the window around a pixel. The gradient preservation makes this a more useful filtering operation than the bilateral filter that can cause gradient reversals. The guided filter algorithm has a running time complexity of $O(N)$, depending only on the number of pixels in the image. Changing the size of the filter or changing the range of image intensities does not effect the running time of the algorithm. These properties made the guided filter a useful tool for a variety of computer vision algorithms. In [39] this filter can be seen applied to a range of problems such as optical flow, image segmentation and stereo. Hosni *et al.* [25] further explore utilizing this filter to implement a local stereo correspondence algorithm. Their implementation is based on GPU architecture with the aim of achieving real-time performance. The edge preserving property allows for smoother disparity estimates with well defined boundaries. This along with the

real-time performance reported in [25] makes their technique the ideal tool for detecting stereo window violation. The disparity estimation used for our stereo window violation detection is based on the work of [25]. We need to compute the disparities relative to both the left and right images to track stereo window violation. The following section describes the steps required to compute the disparity map for the left image. The same steps can be used to compute the disparity for the right view by replacing the left image with the right image and vice versa in all of the equations.

A. Cost Volume Construction

As we have seen in the literature review, the first step of a stereo correspondence algorithm is to define a cost function. The cost function is defined over a range of disparities d across all the pixels in an image p. The image itself is two dimensional and defining the cost per pixel for all possible disparities form a cost volume $C(x, y, d)$. In a stereo correspondence algorithm the cost is defined by the dissimilarity measure between a pixel p at location (x, y) and pixel q at location $(x - d, y)$. Hosni *et al.* recommends using the truncated absolute difference of colours and gradients as a dissimilarity measure. Although colour is the most intuitive feature to use for matching pixels, it can be influenced by illumination changes and varying gains by the sensors of the two cameras. Hence incorporating gradient into the measure makes the function robust against such problems. The absolute difference in colour, denoted as M, is defined in Equation 3.1

$$M(x, y, d) = \sum_{i=1}^{3} |I_L^i(x, y) - I_R^i(x - d, y)| \tag{3.1}$$

where $I^i(x, y)$ represents the value of the i$^{\text{th}}$ colour channel (for a three dimensional colour space such as RGB) at pixel location (x, y). The absolute difference of gradients, denoted G, is express similarly in Equation 3.2.

$$G(x, y, d) = |\delta_x(I_L(x, y)) - \delta_x(I_R(x - d, y))| \tag{3.2}$$

where $\delta_x(I_L(x, y))$ is the horizontal gradient of the image I_L at pixel location (x, y). Combining the two differences the cost volume is defined as follows.

$$C(x, y, d) = \alpha \dot{m}in(T_c, M(x, y, d)) + (1 - \alpha)\dot{m}in(T_g, G(p, d)) \tag{3.3}$$

The constant α controls the contribution of the colour dissimilarity versus the gradient dissimilarity. T_c and T_g are the constants for truncating the dissimilarity measure of the colour and gradient respectively to define the overall truncated absolute difference. Figure 3–5 b) shows a sample cost volume computed using this method.

B. Cost Volume Filtering

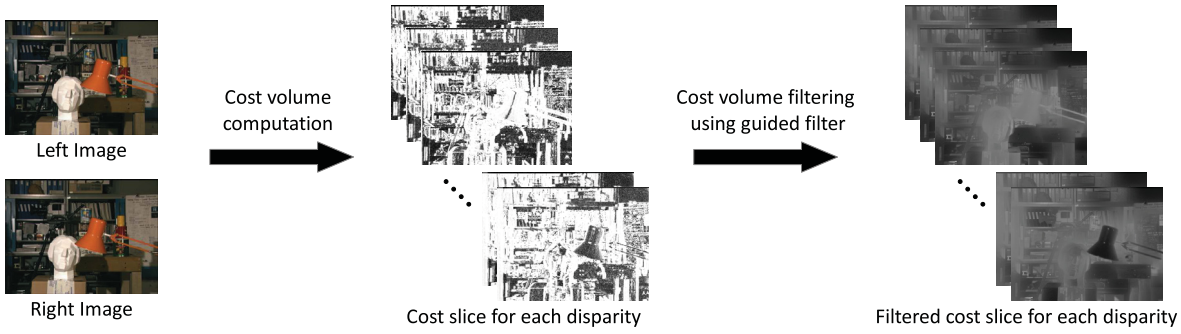


Figure 3–5: Stages of disparity estimation using the guided filter

The cost volume, as seen in Equation 3.3, is defined locally on a per pixel basis. Such a definition, can lead to noisy estimates in disparity as it fails to capture bigger structures in the image. Thus a cost aggregation over a window around these pixels is necessary. Instead of using a uniform aggregation over a rectangular window, Hosni *et al.* utlizes the guided filter to perform the aggregation. Using the guided filter allows us to utilize rectangular windows while preserving the edges using non-uniform weighting. The output of the guided filter as defined by [23] is seen in Equation 3.4.

$$I'_i = \sum_j W_{ij}(I) g_j \tag{3.4}$$

In the above equation, i and j represent the indices of the pixels instead of their coordinates. I' represents the filtered image, g being the guiding image and I the image to be filtered. The variable weights W_{ij} are computed around the pixel index i and the product is summed over the window and assigned to pixel index i in the filtered image I'. In the stereo correspondence problem, the guided filter is to be used for filtering the cost volume. The cost volume is filtered one slice at a time for each disparity value. The filtered cost volume is computed as follows

$$C'(p,d) = \sum_q W_{pq}(I) C(q,d) \tag{3.5}$$

$C'(d)$ is a slice of the filtered cost volume, guided by the image I (left image when computing the left disparity and the right image when computing the right disparity) for disparity value d. In order to compute the weights, we look back at [23] where the guidance image I is used to filter the guided image f (cost volume slices in our case). When using the image as the guide, we have the option of choosing either

the colour image or the grayscale image. The colour image produces better results although computationally more expensive. The weights W_{ij} is defined for a colour guiding image as follows

$$W_{ij} = \frac{1}{|\omega|^2} \sum_{k:(i,j)} \Big(1 + (I_i - \mu_k)^T (\Sigma_k + \epsilon U)^{-1} (I_j - \mu_k)\Big) \tag{3.6}$$

In equation 3.6, the mean μ_k and covariance matrix Σ_k are computed within a window ω_k with dimensions $r \times r$ centred at pixel k of the guiding image I. Other parameters include ϵ, which is a smoothness parameter and $|\omega|$ is the number of pixels in the window. I_i, I_j and μ_k are vectors of size 3×1 as each pixel contain information for the three colour channels. Explicitly computing the weights for the guided filter is computationally expensive. Instead the following linear operations can be performed to obtain the filtered output with greater efficiency.

$$a_k = (\sigma_k + \epsilon U)^{-1} \left(\frac{1}{|\omega|} \sum_{i \in \omega_k} I_i f_i - \mu_k \bar{f}_k \right) \tag{3.7}$$

$$b_k = \bar{f}_k - a_k^T \mu_k \tag{3.8}$$

$$q_i = \bar{a}_i^T I_i + \bar{b}_i \tag{3.9}$$

In Equations 3.7,3.8 and 3.9, f_i represents pixel i of the guided image f (cost slice in the stereo problem), $\bar{f}_k = 1/|\omega| \sum_{i \in \omega_k} b_k$ is the mean of the guided image in the window ω_k. The filtered output is represented by q_i. These set of operations are used to filter each slice of the cost volume to produce the filtered cost volume. Figure 3–5 c) shows the result of filtering the cost volume.

C. Disparity Map Estimation

The advantage of using a local method for stereo correspondence is the ease of optimization. The cost volume was initially computed over a range of disparities and then filtered using the guided filter. The Winner-Takes-All strategy can then be used to get the disparity estimate for each pixel. This is a common strategy utilized in other local stereo methods as well. The disparity for each pixel p is obtained as shown in Equation 3.10

$$d_p = arg \min_{d \in D} C'(p, d) \tag{3.10}$$

D is the set containing all possible disparity values, which depends on the initial range of disparity chosen. Figure 3–6 shows the final disparity estimate obtained using this method along with comparisons from methods mentioned in Section 2.3.

3.2.3 Focus Estimation

The next phase of our proposed method to detect stereo window violations is to find the areas of focus in the image. Focus is a means by which cinematographers draw attention of the viewers to different parts of the scene. It has been and continues to be used to add to the storytelling aspect of a movie. The study in [9] provides a perceptual reasoning behind utilizing focus in our method. Brown *et al.* note that regions that have higher spatial frequency, especially with adjacent regions with lower spatial frequency, appear closer in depth. Higher spatial frequency can be attributed to higher texture from focused imagery with lower spatial frequency coinciding with the out of focus areas. The conclusion of their study also indicate that the cue from spatial frequencies can create a stronger perception of depth than the stereopsis cue.

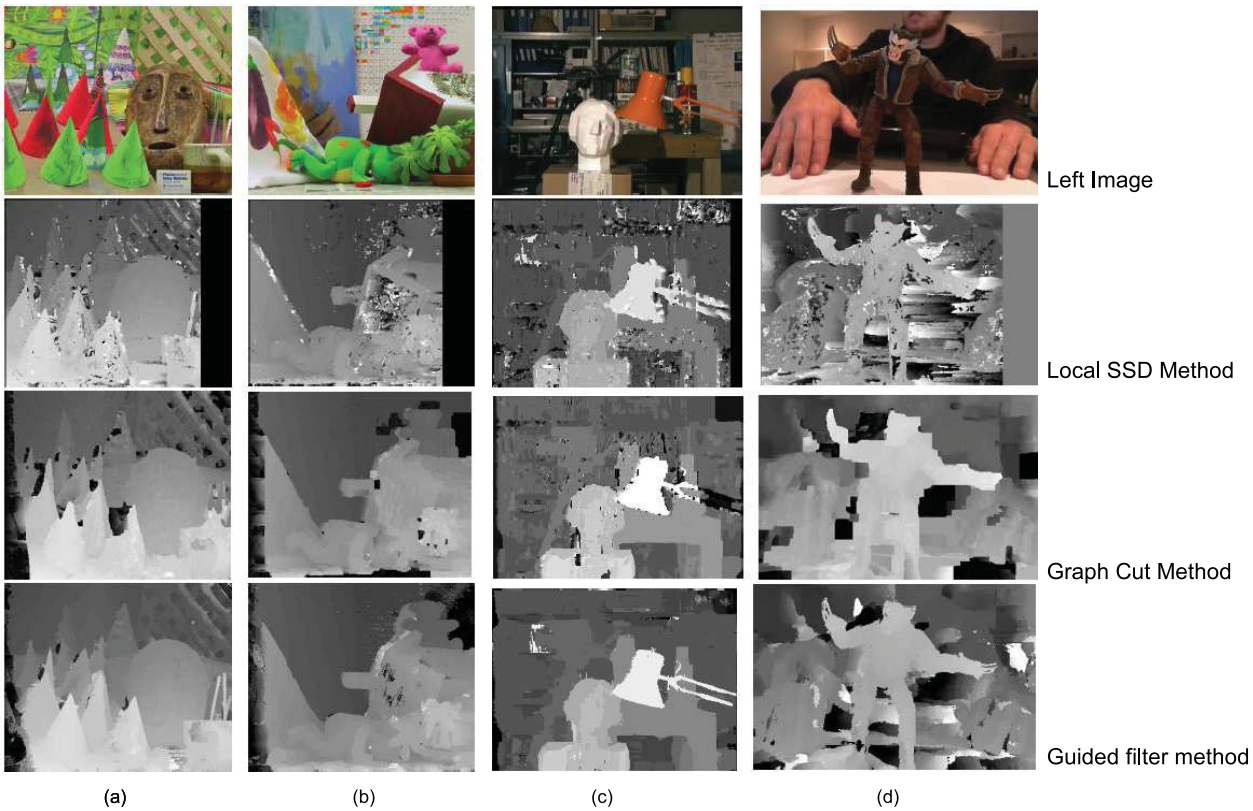


Figure 3–6: Disparity estimation comparison of different methods. Local Sum of Squared Differences (SSD) (2^{nd} Row), Graph Cut (3^{rd} Row) and Guided Filter (4^{th} Row). Column (a)-(c) are images from Middlebury Stereo Dataset and column is an image from our experimental data

This makes it important for us to identify the areas of focus to detect the presence of stereo window violations. Our proposed methodology works with just a pair of stereo images and no external parameters supplied. Hence, no measure or indication of the scene focus is provided along with the images. We must explore the literature to determine ways of estimating focus from images.

The problem of finding the area of focus can be reformulated as finding the blurry (out of focus) regions in the image. By determining the regions that are blurry and the extent of that blur, we essentially have an estimation of focus areas as well. Some methods explored in literature aim to classify an overall image as blurry. Along with the classification an extent of the blur is also stated. Such methods are seen in [51] and [35]. Each paper follows its own approach, [35] taking a probabilistic approach to blur detection whereas [51] analyses the edges in the image using the Haar wavelet transform. The reported results indicate that the method works well in determining whether the image is blurry or not. However, classifying an entire image as being blurred or sharp along with a measure to report the extent of this blur is not useful for our purpose. This does not allow us to determine which regions within the image are out of focus and which regions are in focus. Su *et al.* [46] looks at blur detection from this context and identifies amount of blur in regions of the image. They are able to achieve this by performing a singular value decomposition (SVD) analysis of the image. An image I can be decomposed using SVD and represented as $I = U\Lambda D^T$, where U,V are orthogonal matrices and Λ is a diagonal matrix. The original image I can also be represented as a summation of multiple rank 1 matrices commonly referred to as eigen-images.

$$I = \sum_{i=1}^{n} u_i \lambda_i v_i^T \tag{3.11}$$

In Equation 3.11, u_i, v_i are the column vectors of U, V and the λ_i are the diagonal terms of Λ (also called singular values). This decomposition is utilised in image compression where the compressed image I_k is formed by summing the first k eigen images. The Image I is represented as a weighted summation of the eigen images,

where the singular values represent the weights. The singular values of an SVD operation are arranged from largest to smallest. So by using the first k weights, the small weights at the end are discarded and an approximation is obtained without losing too much detail. This is similar to what happens during image blurring. The large scale details of an image are retained (such as rough shapes) while smaller scale details are discarded. Interpreting this in terms of eigen-images, the smaller singular values that relate to small scale details bear smaller weights for blurred images. This leads to the conclusion that the first few most significant eigen-images therefore have higher weight for a blurry image compared to those of a clear image. This can be extended to finding the amount of blur in regions of a single image. The image can be analysed in local patches around each pixel and SVD is used to calculate the singular value for the patches. The amount of blur for a pixel can be expressed by the following ratio

$$\beta_k = \frac{\sum_{i=1}^{k} \lambda_i}{\sum_{j=1}^{n} \lambda_j} \tag{3.12}$$

In Equation 3.12, λ_i represents the i$^{\text{th}}$ singular value calculated within a local patch $\omega_b \times \omega_b$ for each pixel. Therefore, the measure of blur is the ratio between the first k most significant singular values and all n singular values. The choice of k, depends on the size of window. When SVD is performed on the patch of size $\omega_b \times \omega_b$, the total number of singular values is also equal to ω_b. Thus, k has to be less than the window size. In our experiments, it was noticed that the first two singular value (for a window of size 9) contained more than 90 percent of the information required to represent the shape/structure in patches that are blurry. Incorporating more singular values would cause somewhat textured and sharper regions to be classified as blurry.

High values of k would reduce the variation in the computed metric and make it difficult to distinguish blurred versus focused regions. A low value for k was chosen based on these observations. This methods allows us to measure the amount of blur within the regions of the image for each image being analysed. A slight modification to Equation 3.12 will allow us to estimate the areas of focus instead of blur. This is shown in Equation 3.13.

$$F_p = 1 - \frac{\sum_{i=1}^{k} \lambda_i}{\sum_{j=1}^{n} \lambda_j} \tag{3.13}$$

Both the Equations 3.12, 3.13 are calculations for a single pixel only. These equations have to be used for each pixel in the image to obtain pixel-wise dense estimates of blur or focus. Figure 3–7 shows the results of estimating the focus in an image using this method.

3.2.4 Stereo Window Violation Detection

Our final goal in this solution is to be able to track objects that appear in front of the screen and are in focus. From the previous sections, we obtained pixel-wise estimates of the disparity as well as focus. But the pixels themselves do not represent objects and tracking each individual pixel for a potential window violation does not make sense. Even if fifty pixels were causing a window violation it would be hardly noticeable among the large number of pixels in the screen. Therefore, for our final step we find segmentations of meaningful objects that can be tracked. The segments are found by clustering pixels together.

The clustering technique used in our method is the mean shift clustering [12]. Mean shift is a very popular non-parametric clustering technique that has been used for segmentation, clustering, tracking, space analysis and other applications. Unlike

(a) Original Image (b) Regions of focus

(c) Original Image (d) Regions of focus

Figure 3–7: Focus estimation Results

some other clustering techniques, the mean shift method does not require us to supply the number of clusters. The technique analyses an image in feature space. For example, when segmenting a grayscale image, the intensity values would serve as a one-dimensional feature space. Similarly, for a colour image each colour channel serves as a feature forming a three dimensional feature space. The technique can perform clustering using a variety of relevant feature dimensions. For our purpose, we used a three dimensional feature space to form our clusters, but the features for

our clustering were not the colour channels. The first two features were the disparity and focus estimates. Disparity is a useful feature to direct the segmentation. We threshold the disparity to remove pixels that appear behind the screen. Objects in real-life are piecewise smooth and points on an object are expected to have similar disparities. Grouping together pixels based on their disparity value alone is not sufficient. One reason is that the quality of the disparity map may be degraded due to the issues such as half occlusion, low texture, repetitive features, etc. Additionally we want our objects, for tracking, to be both in front of the screen as well as in focus. Hence the focus estimates are also utilized as a feature for clustering as well. Along with those two, the third feature that was utilized is the grayscale intensity values of the image (left image for left disparity map and vice versa). This feature is useful for guiding the boundaries of the segments by incorporating intensity based edge information into the clustering process. All three colour channels could have been utilized instead of just the grayscale intensities but they add computational complexity and provide minimal improvement from using just the grayscale values. The following steps were used in our implementation of the the mean shift clustering algorithm to produce object segmentations:

1. A random point is chosen as the centre of the searching window and its feature vector is chosen as the mean.
2. A new mean is calculated using all points within a distance defined by the *bandwidth* b_w.
3. The new centre for search is moved to the newly computed mean.

4. Steps 2-3 are repeated until the shift of the mean is very small (e.g. 0.1% of bandwidth)
5. All points traversed by the searching window are set to be of the same cluster.
6. A new random point is picked from the remaining points and steps 2-3 are repeated again while avoiding points already assigned to a cluster.
7. Clusters with centre distance less than half the bandwidth are merged.

The bandwidth is the only parameter that affects the result of the clustering. The bandwidth is essentially a threshold for the distance from the centre of the searching window to the points around it. The distance is measured in feature space and is usually as an absolute difference or squared difference. If the feature spaces are normalized a single bandwidth is sufficient for clustering, otherwise varying bandwidth for each feature space may be needed for better results. Choosing a larger bandwidth reduces the number of clusters and a smaller bandwidth increases number of clusters. Our choice of feature space allows objects that have a high disparity and are in focus (less blur) to be formed into a cluster. This is ideal to track objects that may cause stereo window violation. The clusters were used to create binary masks and segment the objects. The contours of the segmented object were determined and a bounding box was calculated which was used to track the objects from frame to frame. For the problem of stereo window violation, the size of the object being tracked is also important. Small objects such as flying birds, floating leaves, balls, etc. would not be an issue for audiences and raising a flag for these items would be an unnecessary hindrance to stereo content generation. Objects smaller than a given size thresholds are thus exempted from being tracked. Due to the nature of

the disparity, stereo window violations occur when objects in the left image cross the right screen edge or when the objects in the right image cross the left screen edge. Our detection system issues a warning message when an object is approaching either of these edges and issues a window violation message as soon as a significant portion has crossed the edge. The system is also able to track multiple objects that are in front of the screen and monitor them for potential violations.

3.3 Experimental Setup and Results

The individual components of our framework were first tested individually before testing the overall system as a stereo window violation detector. The stereo correspondence techniques used to extract disparity were first tested on the Middlebury Stereo Dataset [42, 43, 24]. The dataset consists of a variety of stereo images, usually with indoor scenes and controlled lighting. Utilizing this dataset gave us an early indication as to which of the stereo correspondence techniques from literature would be useful for our purpose. For testing focus estimation, images were obtained from the Flickr website. The images were chosen to all have some objects in focus. However, the scenarios in these images are limited and do not exactly emulate the cases of stereo window violation. To test different variations of stereo window violations we created our own dataset. A pair of Microsoft LifeCam Cinema cameras were fixed together to form a stereo rig. The cameras were placed in a parallel, side by side, configuration with a interaxial distance of 3 cm and a slight toe-in. Prior to filming our test scenes, the cameras were calibrated. The calibration process allows us to estimate the coefficients for radial and tangential distortions caused by

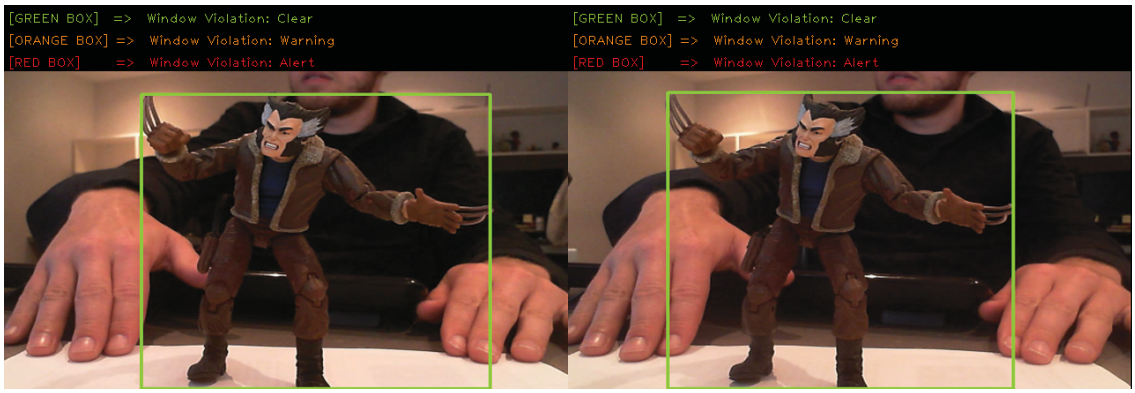
the lenses. These coefficients can be used to eliminate (minimize as much as possible) the effect of distortions in each of the cameras. The process was done using existing tools in OpenCV for calibration [7]. The images obtained using this stereo rig also underwent the stereo rectification process using tools from OpenCV. This ensures that the epipolar lines are along the horizontal scanline, allowing to search for stereo correspondences along a single dimension. The resolution of our camera setup supports 720p videos. However, rectification leads to a loss of pixels as some portions of the image are cropped as part of the process. Small clips depicting various occurrences of stereo window violation were recorded using this setup.

To recreate scenes with stereo window violation we need to have objects that would perceptually appear in front of the screen. Horizontal image translation were used to control the zero disparity plane and the location of the stereo window in the depth dimension. The first short sequences depicts a figurine that appears in front of the stereo window. The figure is then slowly moved across the screen and it exits the screen from the right border. It comes back into the screen, is moved across and then exits via the left border. Both of the situations as it cross the edges triggers a window violation in the left and right views respectively. This scene had a somewhat static background with one prominent object, in front of the stereo window, moving across the screen. The next scenario introduces another character into the scene which is also in focus. One character starts of in front of the stereo window while another one is behind. The character in front moves across the screen whereas the character behind approaches the screen and crosses it to appear in front of it. Our system successfully tracks the character in front of the screen on its own, and as soon as

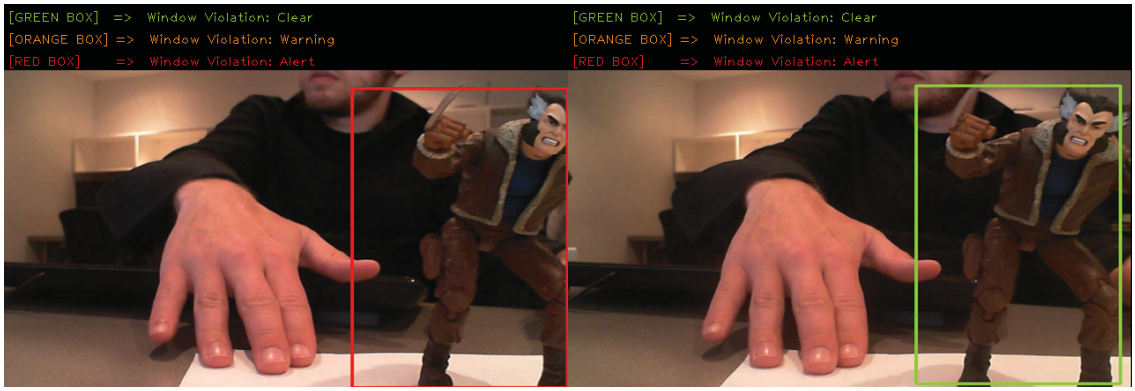
the second character cross the window threshold a second tracker is applied keeping track of both for possible window violations. Other scenes included introducing focus on characters only, and a character starting at a position of window violation instead of just moving across and triggering it. Overall, a test set consisting of 2234 frames was used to validate our system with various conditions. The output of our system is represented by a bounding box around any objects that appear in front of the screen. The boxes have a colour associated with them to indicate the status. The statuses are as follows:

1. Red = Violation: Object, being tracked in front of the screen, is causing a stereo window violation.
2. Green = Clear: No stereo window violation by objects being tracked inside the bounding box.
3. Orange = Warning: Object being tracked is approaching the edge and there is a possibility of a stereo window violation.

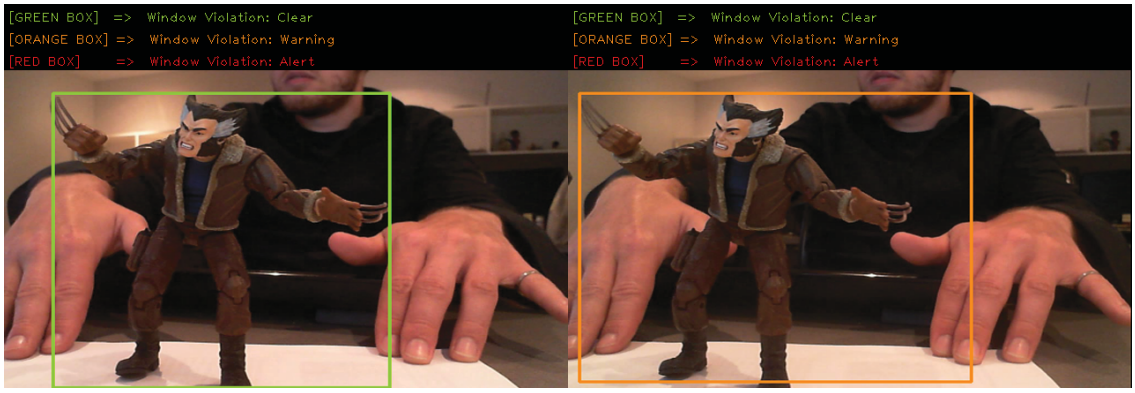
Along with the clear indicator and the alert for a violation, a warning is introduced to indicate approaching violations. Each of the frames being tested were hand labeled and the outputs from our method was compared against them. The numerical results behind the detections are seen in Table 3–1. The numbers indicate that our system does very well at detecting violations and its absence. The test cases for the warning status were few and being a borderline condition made its accuracy fall a bit more compared to the other two. Figure 3–8 and 3–9 show sample results of our window violation detector for a couple of scenarios. Overall, this method is a reliable indicator for stereo window violations.



(a) No Violations.



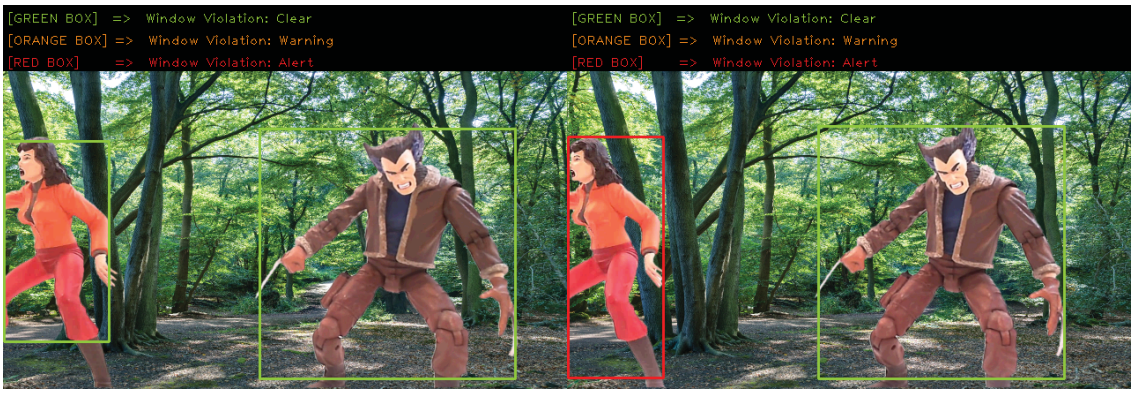
(b) Violation on the right edge found of the left view.



(c) Warning for approaching violation on the left edge of the right view

Figure 3–8: Stereo window violation detection with single object appearing in front of the screen.

(a) No Violations. Only one object being tracked as the other is still behind the screen.



(b) Violation on the left edge is found in the right view by one object while the other object is clear of violation as it appears in front of screen.

(c) Warning for an object in front of the screen approaching the right edge.

Figure 3–9: Stereo window violation detection with multiple objects.

		True Condition		
		Violation	**Clear**	**Warning**
Detection from algorithm	**Violation**	535	16	13
	Clear	12	1627	1
	Warning	3	1	26
Total Test Frames		550	1644	40
Hit Rate		97.27%	98.26%	65%
False Positive Rate		1.72%	2.26%	0.19%

Table 3–1: Results of SVW

3.4 Target Application in 3D movies industry

The ability to create 3D movies has existed for quite a long time. However, content of such nature was not very popular during its early days. Very few movies were made using stereoscopic technology, even fewer in the case of feature films. The prevelance of 3D was retained through shorter films or video sequences to induce thrills as a novelty item. A major factor behind the failure of this technology to catch on with the viewing public was the inability to create high quality 3D content. Without quality stereo production "3D" was reduced to a gimmick rather than being a component of storytelling that creates a more immersive experience. Early stereo movies were created using analog technology. Many common issues or pitfalls regarding stereo content were not fully appreciateed at that time and the analog format made post processing a difficult proposition. Hence, the quality of the content produced was not high and more often created a strenuous experience for viewers than an enjoyable one. All of these factors contributed to the falling popularity of 3D content in mainstream media. However, stereoscopic content has been rejuvenated with the emergence of digital technology and 3D movies have made

a return to mainstream media. Many feature films in recent times have adopted this 3D technology. Much 3D content is being created through filming using stereo rigs as well as conversion methods. Specialists in stereography can look at the digital footage of scenes being shot on set or during post production. Their expertise can be used to judge if the content has any artefact that will make the viewing experience uncomfortable. The motivation behind the work in this thesis is to provide tools that help such specialists or help production crews that lack such a specialist the ability to detect a major problem such as stereo window violation. Having such a tool on set will make it easier to ensure proper framing is done. This can avoid actions such as cropping in post production to ensure stereo defect correction and instead preserve pixels from the original shot. Early access to the quality analysis of the scenes can avoid having to bring actors back in for shooting a scene that was not correctable in post production. With this realization our contribution in this thesis is aimed at developing tools to assist the production of stereo content. This chapter addresses the specific problem of stereo window violation which is a very prevalent problem while creating stereoscopic content. The results indicate that our methodology can serve as an excellent tool to avoid the problem.

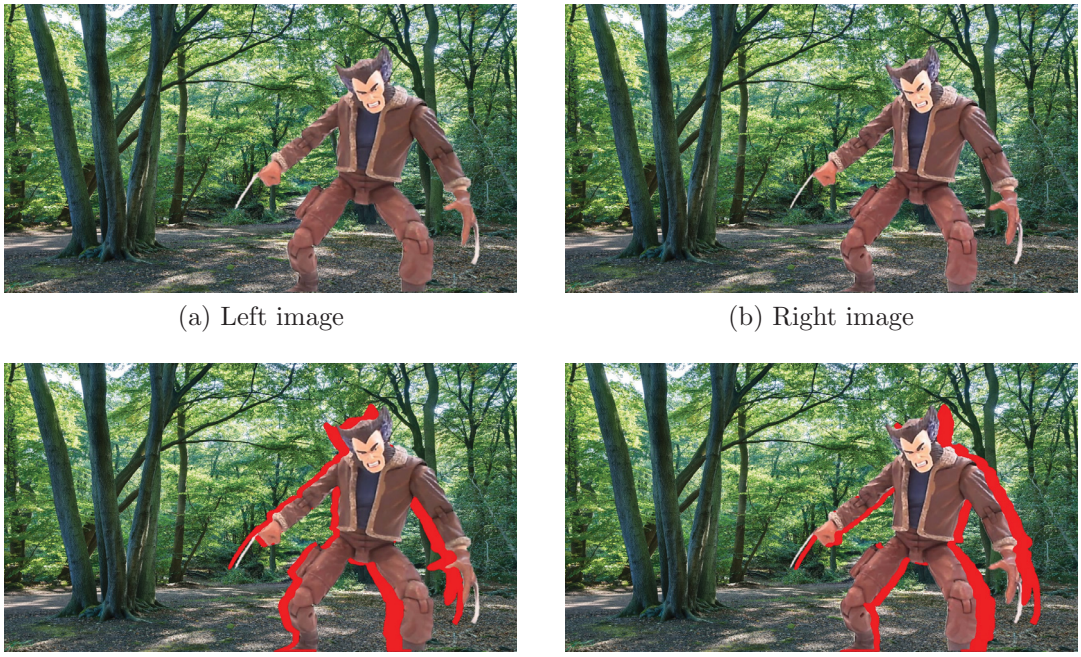
CHAPTER 4
Half occlusion and its application to stereo defect analysis

This chapter of the thesis discusses the issue of half occlusion. Half Occlusion is formally defined and its relationship to stereo movies is established. The proposed solution to finding half occlusions in a pair of stereo images is discussed. We also look at how half occlusions can help identify certain problems in stereo movies and improve stereo correspondence accuracy.

4.1 Half Occlusion

Half occlusion is a feature that is present in most stereo image pairs, regardless of the type of setup being used to create the content. Although half occlusions lead to difficulties in stereo correspondence, they can be regarded as more of a feature than an artefact. This is because half occlusions can help define depth discontinuities resulting in improved segmentation of objects. Half occlusions can also provide cues to help identify where we should look for other stereo artefacts such as pseudo stereoscopy or occlusion-stereopsis conflicts. Further into this chapter we explore why half occlusions, once indentified, can be a blessing rather than a hindrance. Half occlusions can be considered an extension of occlusions to stereo imagery. Any object that appears in front of another object and hides it from view creates an occlusion. The occluding object in front hides the occluded object in the background. Occlusions are present in each of the individual images in a stereo pair. Each of the stereo images represent a slightly different perspective of the same scene. The

location of objects is shifted in each view based on the disparity. Objects that are closer will have a higher disparity than the ones further away. Since the larger disparity creates a larger shift, the occluding object will occlude a different portion of the background in each view. As a consequence of the same effect, a different portion of the background is revealed in each view which does not have a matching to the other view. This region is only occluded in one of the images hence it is referred to as half occlusion. Half occlusion is similarly defined in many of the works throughout literature [34, 3, 21, 15]. In order to find these half occlusions we need to identify these regions that appear in only one of the views. Figure 4–1 shows half occlusion regions highlighted in the stereo image pair.



(a) Left image

(b) Right image

(c) Left image with annotated half occlusion

(d) Right image with annotated half occlusion

Figure 4–1: Example of Half Occlusion regions from composite images

The geometry of the camera setup with respect to the scene plays a key role in determining the half occlusion regions. The illustration in Figure 4–2 shows how the geometry dictates the constraints on half occlusion. The left camera, capturing the left stereo image, has a greater view of the background to the left of the occluding object. Greater view in this context means more of the background is seen by the left view at this location compared to the right view. Hence, regions without valid matchings, due to half occlusion, appear to the left of the occluding object in the left stereo image. Similarly, in the right stereo image, these regions would appear to the right of the occluding object. This constraint holds for a converged camera setup or a parallel setup. The parallel setup may be by choice to begin with or could be from the result of image rectification. The width of the half occlusion region is determined by the relative depth of the occluding object and the background. A greater difference in depth would mean a greater shift of the occluding object in front leading to larger regions of half occlusions. These properties are core to the definition of half occlusion. We see how they are utilized for detecting half occlusions in the following section.

4.2 Proposed Framework for detecting half occlusions

Half occlusions are very highly dependant on the depth of a scene. Hence most of the approaches in the literature that are directed towards finding half occlusions also require the depth of a scene to be computed. Finding half occlusions can be treated as a post processing step after computing the depth map. Alternatively, half occlusions can be incorporated into the process of estimating the depth by treating it as one of the possible labels. The search space for the number of possible occluding

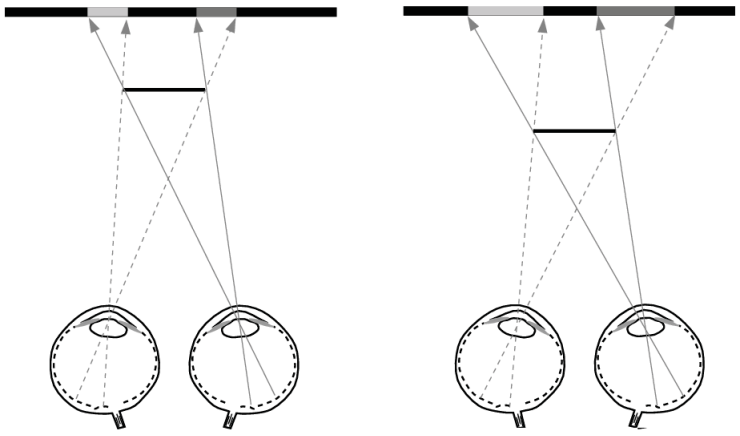


Figure 4–2: Geometry of half occlusion [56]. The distance between the near and far objects influence the size of the half occlusion.

pixels is as large as the disparity range and this can add to the complexity of the solution. Another viable approach is to utilize an iterative algorithm that computes half occlusion as a post processing step but then utilizes it for improving the disparity estimate.

4.2.1 Overview

The dependency between depth and half occlusions makes an iterative technique very useful. If we are given fully accurate depth maps, for both views, it is simple to obtain a perfect estimation of the half occlusions. Our analysis is based on the usage of only stereo vision techniques. There is no access to depth information from other sources such as lasers, Kinect sensors, etc. The depth estimate obtained by the most sophisticated of stereo correspondence algorithms (especially those that do not account for half occlusions) are still far from perfect. There could be mismatches caused by slight illumination variances, lack of textures, specularities, etc. Even if we assume that a filming crew can monitor the factors behind these problems and

control them such that their effects are minimized, the presence of half occlusions will still create inaccuracies. The half occlusions will always leave regions that do not have true stereo correspondences. By knowing the location of the half occlusions, we can compute better disparity maps with sharper boundaries. However, to determine the half occlusions we need to have good quality disparity maps. This serves as the motivation for choosing an iterative approach. We can first estimate the depth of a scene from an initial disparity map and then use it to obtain our half occlusion regions. These half occlusions can reinforce our disparity maps and then lead to better estimates of half occlusion in subsequent iterations. Figure 4–3 shows the components in our framework for detecting half occlusions.

4.2.2 Half Occlusion Estimation

The first step of finding half occlusions would be to determine the depth of the scene. In computer vision, we do this by estimating the binocular disparity map. In order to find half occlusions we need to compute the disparity map for both sides, doing the left to right stereo correspondence as well as the right to left stereo correspondence. By definition, the left disparity map represents the correspondences to points in the right view of the stereo pair. The disparity is the horizontal shift in the pixel location required to find the matching pixel in the other view. Hence, by performing this shift we should be able to reconstruct the one of the views using the disparity map of another. Figure 4–4 shows that such a process would not produce a complete reconstruction. There would be gaps in the reconstruction because some pixels do not have matches in the other view. These hollow gaps are due to half

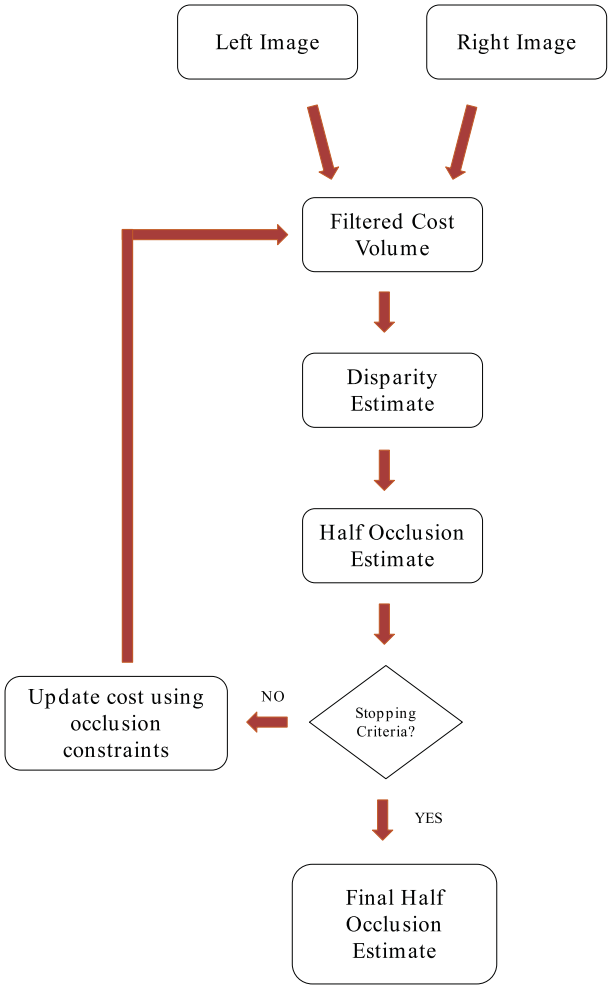


Figure 4–3: Flowchart outlining the process of half occlusion detection

occlusions creating regions with no correspondences.

$$I'_L(x + d_R(x, y), y) = I_R(x, y) \tag{4.1}$$

$$I'_R(x + d_L(x, y), y) = I_L(x, y) \tag{4.2}$$

In Equations 4.1 and 4.2, I'_L and I'_R represent the reconstructed images created using the disparity estimates d_R and d_L respectively. The reconstructed images are

initialised to zeros in each of the color channel. Hence, this form of reconstruction results in the creation of hollow regions. The half occlusions estimates are represented by boolean values as follows

$$O_L(x,y) = \begin{cases} 1 & \text{if } I'_L(x,y) = 0; \\ 0 & \text{otherwise.} \end{cases} \tag{4.3}$$

$$O_R(x,y) = \begin{cases} 1 & \text{if } I'_R(x,y) = 0; \\ 0 & \text{otherwise.} \end{cases} \tag{4.4}$$

Through equations 4.1 - 4.4 all three colour channels are used for the reconstruction and checking for hollow regions of half occlusion. This process only utilises the information about regions with no matching and identifies them. With a perfect disparity map, this would have been sufficient to determine the half occlusions. However, since we do not have access to error free disparity maps we need to utilize other properties of half occlusions. A small deviation in disparity estimation may lead to many small hollow pixels. But before confirming these as half occlusion we can utilize the property that being a half occluded point means there has to be a corresponding occluding object. Some of these properties are incorporated as constraints to allow for better disparity estimation which in turn improves half occlusion estimation.

4.2.3 Incorporating Occlusion Constraints to disparity estimation

Our approach to applying occlusion constraints is based on the work in [62]. Zhang*et al.* utilize an MRF based approach for stereo correspondence. On top of that, they perform oversegmentation on the source images to form super pixels. The

(a) Original left image

(b) Original right image

(c) Reconstructed left image

(d) Reconstructed right image

Figure 4–4: Reconstructing images from disparity estimation to reveal half occlusions

MRF with its data and smoothness terms are then defined in terms of these superpixels. The advantage of using superpixels over the traditional pixels is the reduction of computational complexity when performing the optimization. When applying the occlusion constraints, their energy functions are modified to incorporate that information. We have mentioned how computing the disparity map is an important first step of determining half occlusions without elaborating on how this disparity map is to be computed. We can follow the approach of Zhang *et al.* and construct an MRF

and incorporate the occlusion constraints to the energy function. However, the main motivation of this thesis is to provide tools to analyse stereo content. To that end, it is more practical to extend the work from the previous chapter instead of starting from scratch with a completely new approach. It will be more efficient to use the intermediate information from the stereo window violation detector to estimate our half occlusions.

Looking at equations 4.3 and 4.4 we can see that pixels that are visible to both views would be marked as not occluded. However, for each half occlusion there has to be a corresponding pixel on the other view that occludes this pixel. This occluding pixel has to be one that is visible to both views. Another key factor is that the occluding pixel has to have a larger disparity than the half occlusions regions. Only a closer pixel should be able to occlude a pixel that is further away while the opposite is completely invalid. Based on these rulings, we evaluate the validity of our boolean occlusion estimates and apply constraints to our depth estimate. Given a pixel (x, y) in the right image is regarded as a half occlusion, the disparity is constrained based on which of the following criteria it falls under[62]:

$$G_1 = \{L | O_L(x + L, y) = 0, d_L(x + L, y) > L)\} \tag{4.5}$$

$$G_2 = \{L | O_L(x + L, y) = 0, d_L(x + L, y) < L)\} \tag{4.6}$$

$$G_3 = \{L | O_L(x + L, y) = 1\} \tag{4.7}$$

Equation 4.5 shows a case for a valid half occlusion where the corresponding point of half occluded pixel in the right image is a visible pixel in the left image that has

a greater disparity than the pixel in the right image. Equation 4.6 is similar except that the corresponding point in the left image has a smaller disparity than the half occluded point in the right image. This is a case of invalid half occlusion as having a smaller disparity would mean the occluding pixel is behind the occluded pixel which is not possible. Equation 4.7 shows a scenario where the corresponding point in the left image also has no matching. Both of the pixels in the left and right cannot be without matching, as an unmatched point must be occluded by a visible pixel in the other view. Therefore, the labels have to be adjusted to account for this situation. In [62], the energy functions of the MRF are modified with these scenarios in mind. In our case, we can follow a similar approach but instead modify the cost volume that was calculated in Equation 3.3. This is where our approach deviates from Zhang *et al.*. Instead of working at a superpixel level we apply the constraint and adjust the cost volume at a pixel level. The following equations show the modification to the cost volume:

$$C'(x,y,d) = \begin{cases} \min\limits_{d \in G1} C(x,y,d) & \text{if } d \in G_1; \\ +\infty & \text{if } d \in G2; \\ C(x,y,d) & d \in G_3. \end{cases} \tag{4.8}$$

We obtained the half occlusions from our initial estimation of the disparity map. Using the constraints in Equation 4.8 we can adjust the cost volume and then obtain an updated disparity map that accounts for the half occlusions. We iteratively update the disparity estimates and the half occlusion estimates until the change becomes minimal. Even with the iterative approach the half occlusion estimation will not always be perfect due to the presence of noise in the disparity estimate. Therefore

three post processing steps were applied to the half occlusion estimates from the iterative process. The first post processing step involves applying a morphological operator to the estimate. Specifically a morphological closing (dilation followed by an erosion) operation is performed to close up holes in the estimates of half occlusion. Following this, a size filter is applied to the output to remove small noisy specks in the estimate. The last step of post processing involves utilizing some information regarding half occlusions as opposed to general image processing techniques. Half occlusions are guided by the location of the camera and the difference in depth between the occluding and occluded surfaces. The width of the half occlusion region is dictated by these factors. We can utilize these information to enforce the width of the half occlusion in terms of pixels in our own estimate and improve the results. The stereo rectification process yields a pair of images that mimic the geometry of images captured using parallel configuration whereas the disparity serves as a representation of the depth. Hence, the width of the half occlusions, in terms of pixels, can be represented by the disparity difference between the occluding surface and the occluded surface. But the occluded surfaces do not have a reliable disparity estimate as they do not have any matching points. The closest pixel that is not a half occlusion is used to compute this disparity difference. Using a single pixel is sensitive to noise. To solve this issue, an average over 3 consecutive reliable pixels is used instead of the single pixel. In the left half occlusion regions, this search for reliable pixels is conducted to the left of the half occlusion region and to the right of the half occlusion regions in the right image. This is because half occlusions exists on the left side of depth boundaries on the left image and on the right side of the

depth boundaries in the right image as has been touched upon previously. Utilizing these final steps we obtain our estimation for half occlusion regions in both the left and right images, while obtaining a refined disparity estimate in the process.

4.3 Experiment

Testing for half occlusions detection also requires access to stereo image pairs. Therefore, similar to Chapter 3 we utilize the Middlebury Stereo Dataset for testing. This dataset contains ground truth disparity values that can be used to obtain ground truth values for half occlusions. Many methods in the literature use images from this dataset allowing possible comparisons in results to existing methods. Figure 4–5 shows the results of half occlusion estimation for stereo image pairs from the Middlebury Dataset.

Along with this standard dataset we utilized our stereo camera setup discussed in Chapter 3 to record a small sequence in front of a white screen. Using the white screen allowed us to perform compositing and add our own background to the image. Using horizontal image translations the background was shifted backwards to ensure it would always be the occluded surface in the scene. As the object captured in our video moves through the scene, it covers a different portion of the background allowing us to establish a test set with known half occlusions. The results of estimation on that data is shown in Figure 4–6.

4.4 Motivation and usage

Half occlusions are usually present in stereo imagery. Their presence will affect the quality of stereo correspondence in a negative manner. Therefore it is vital to not ignore them and instead make an effort to detect these regions. The first immediate

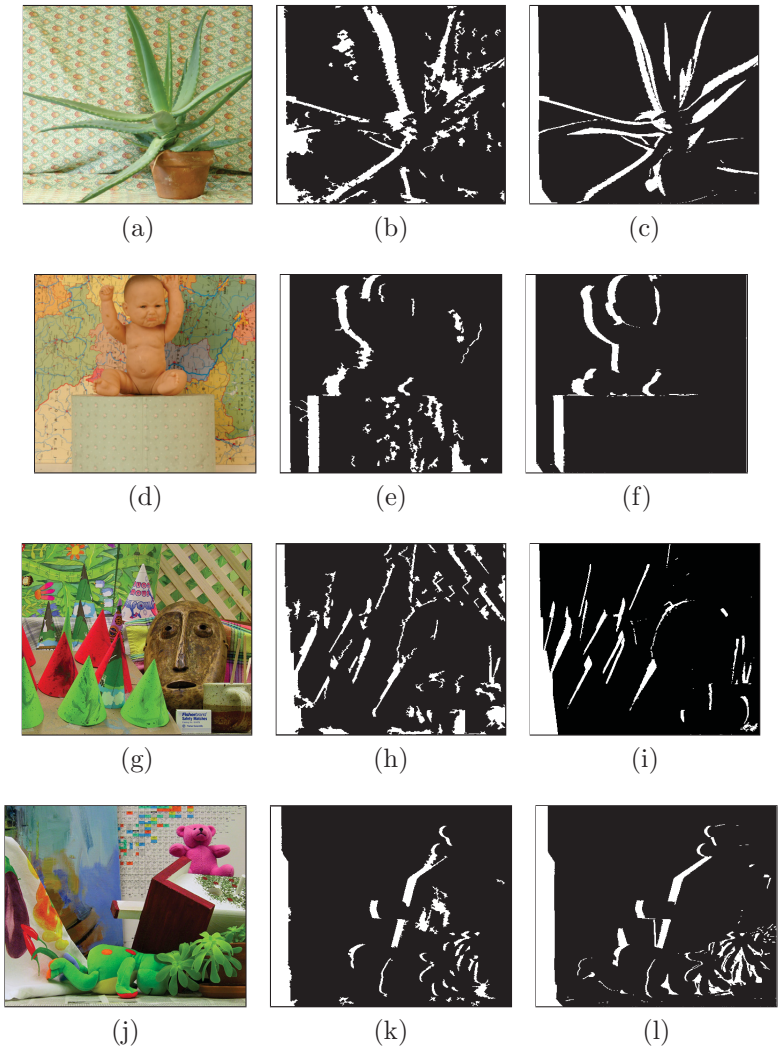


Figure 4–5: Half occlusion results on Middlebury Dataset. First column shows left view of the stereo pair. Second column shows the estimation of half occlusions on the left view using the proposed method. The third column shows the ground truth half occlusions for these images.

benefit of identifying half occlusions would be the reinforcement to the disparity estimation. Blurry and poorly defined edges can be made sharper if half occlusions are known. Sharper depth boundaries would be beneficial to a variety of stereo

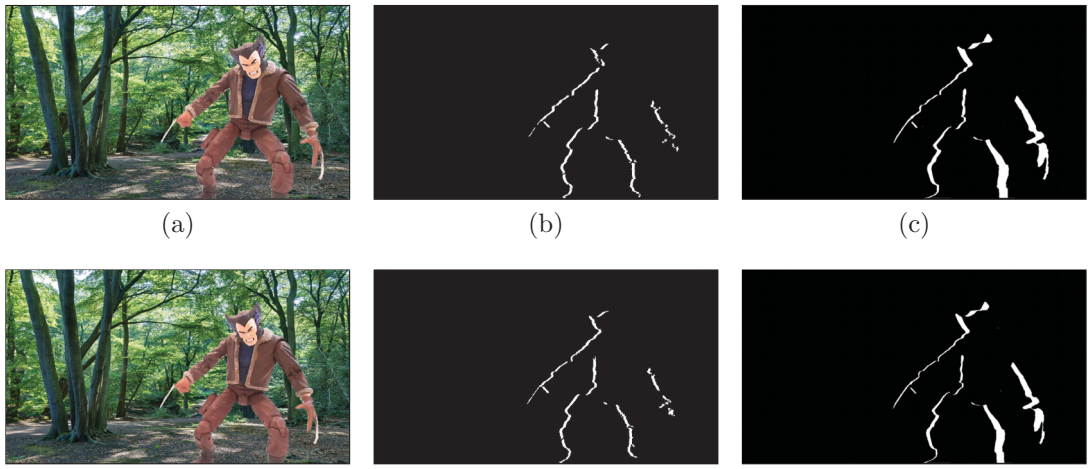


Figure 4–6: Half occlusion results from composited images created using our stereo rig set-up. First column shows left view of the stereo pair. Second column shows the estimation of half occlusions on the left view using the proposed method. The third column shows the ground truth half occlusions for these images.

related issues. It will allow better object segmentation and tracking. Moreover, a few stereo problems manifest themselves on these depth boundaries. Utilizing the knowledge of half occlusions and depth boundaries can help avoid problems of pseudo-stereoscopy. Pseudo-sterescopy can be defined as a form of faulty stereo content that contains conflicting depth cues, incorrect calibration, etc. Depth perception from such content is not completely lost, however its faulty nature can lead to incorrect interpretations. Such content creates strain on the users and may eventually break the depth perception as the problems get worse. The simplest example of pseudo-stereoscopy is when the left and right views are swapped. Similarly the left and right lens of the 3D glasses used when viewing 3D content can also be swapped. Doing so does not completely destroy the depth perception but it creates a strong

occlusion-stereopsis conflict in the depth cue. Objects that are in front are supposed to be at the back according to stereopsis cues, whereas its occlusion cue tells the brain that it should be in front. The brain might swap back and forth trying to decide how to properly perceive the scene. The process will put the brain under stress while continuously breaking the depth perception of 3D movies. Even without the swapping of the left right views, pseudo-stereoscopy can manifest when elements within the scene are not at the right depth. This problem can originate during 2D to 3D conversions or similarly in 3D compositing. During these processes, artists insert new elements into a scene that has already been shot. Creating proper stereo cues for the composited object is vital to avoiding pseudo-stereoscopy. This chapter of the thesis is motivated by the need to detect such problems. We identify half occlusions regions as a vital feature in determining depth boundaries and hence identifying regions that need to be analysed to detect faults such as stereo reversals, occlusion-steropsis conflicts, etc.

4.4.1 Detecting Pseudoscopy using Half Occlusions

In this section we explore the detection of pseudoscopic videos, where the left and right stereo pairs are reversed when presented to the viewer's eyes. Our solution to detecting pseudoscopy is based on using half occlusions. We have defined half-occlusions as regions that are visible to only one eye and not to the other. These regions create difficulty for stereo vision as they do not have any matches or corresponding points in the other view, making it impossible to compute binocular disparities. However, they prove to be very useful when used to detect if an image is presented stereoscopically or pseudoscopically. Figure 4–7 shows the location of half

occluded regions in a simple scene consisting of a flat object in front of a back-plane. In a stereoscopic presentation the left eye sees the left-half-occlusion region (which is the region which the left eye can see but the right cannot) at the left edge of the front object. The right eye sees the right-half-occlusion region (which is the region which the right eye can see but the left eye cannot) at the right edge of the front object.

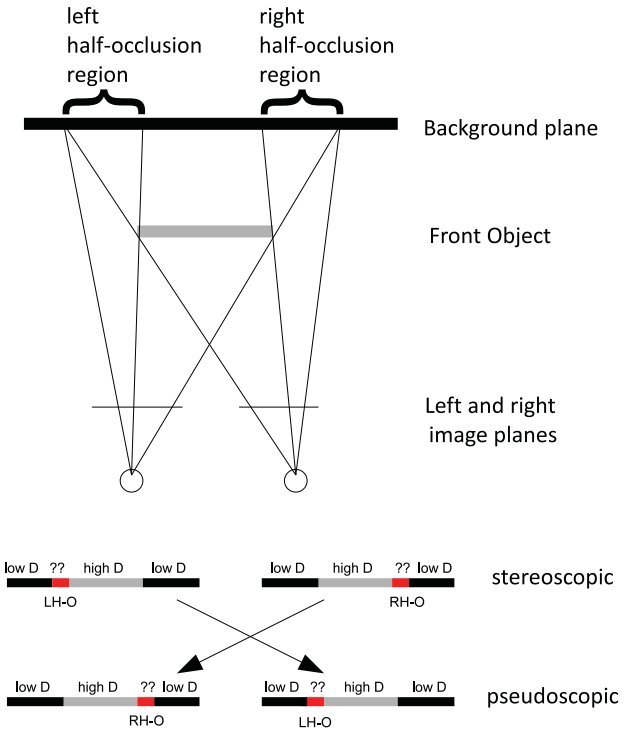


Figure 4–7: The left and right half-occluded regions in a simple scene of a small planar object in front of a planar background. Shown at the bottom are depictions of the left and right images in stereoscopic (top) and pseudoscopic (bottom) display. The black regions correspond to images of the background and the gray regions to the images of the foreground object. The red regions correspond to the half-occlusions. The disparity levels for the image regions are also shown. The disparity of the near object are higher than that of the background. There is no disparity defined in the half-occlusion regions.

In the case shown in Figure 4–7 of an object in front of the background, it can be seen that, in the case of stereoscopic display, the left half-occlusions are to the left side of the foreground object, while the right half-occlusions are to the right side. This pattern is reversed in the pseudoscopic display. If we knew we were looking at this type of a simple scene then we could determine whether the display is stereoscopic or pseudoscopic just by taking the difference of the x-coordinate of the left-half-occlusions and the right half-occlusions. If this difference is negative then the display is stereoscopic. If it is positive then the display is pseudoscopic. However, one could have a scene such as that shown in Figure 4–8, in which the background plane is being viewed through a small hole in the foreground plane. In this case the locations of the left and right half occlusions are reversed from what they were in the case of the foreground object occluding the background. Thus it would seem that a simple differencing of the left and right half occlusion locations would not help in detecting pseudoscopy. But, in most scenes encountered in practice during the filming of 3D movies, the number of foreground objects occluding background objects is generally much higher than the number of holes in the foreground objects. Thus, the situation in Figure 4–7 dominates over that of Figure 4–8, and we can reliably use the left-right occlusion location differences to determine whether the imagery is presented normally or pseudoscopically. In practice we would compute the centroids of all of the left and right half-occlusion pixels in the imagery and use the difference of the centroids in deciding whether the display is pseudoscopic. It is interesting to note that, when a scene is viewed pseudoscopically, the occluding foreground objects appear to be background objects viewed through holes that are

the exact shape of these objects. It may be the unnatural preponderance of such holes perceived in pseudoscopic imagery that leads to the feeling on the part of the viewer that something is wrong with the imagery.

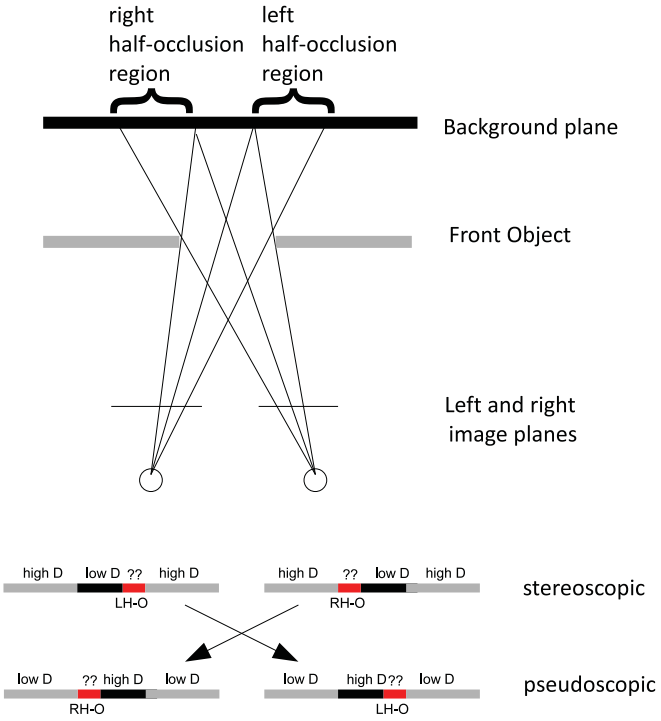


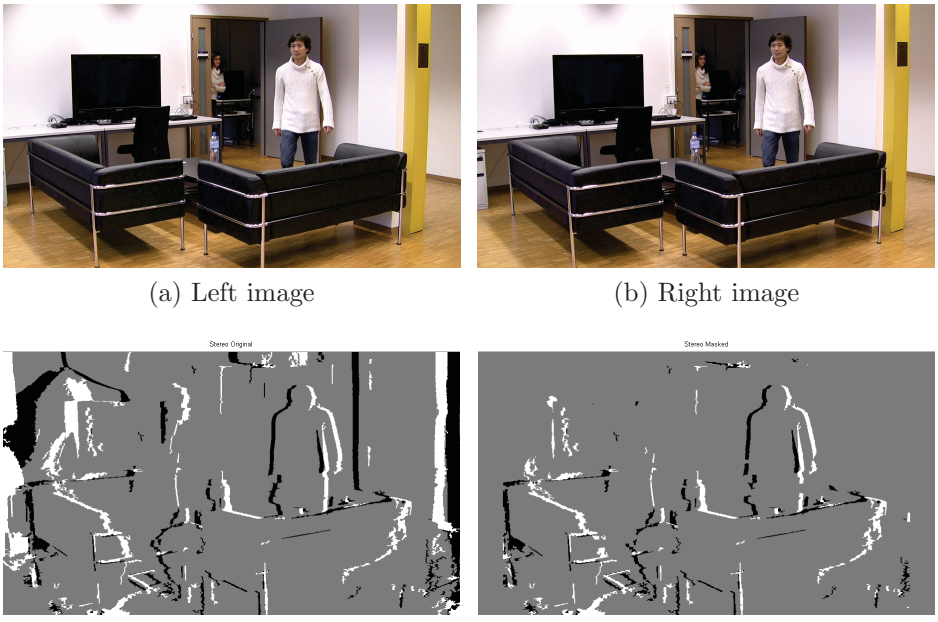
Figure 4–8: The left and right half-occluded regions in a simple scene of a planar background being viewed through a hole in the foreground plane. Shown at the bottom are depictions of the left and right images in stereoscopic (top) and pseudoscopic (bottom) display. The disparity levels for the image regions are also shown.

Even in naturally occurring scenes, there may be roughly equal numbers of holes and occluding objects. In this case our left-right occlusion difference method will not work. This type of situation would reveal itself by a relatively low difference value. For such cases we need to use another approach. One such approach suggested by Akimov *et al.* [1] is to observe the appearance (say of the intensity, or of the texture)

of the visible part of the half-occlusion regions as compared with the foreground and background regions. While it is true that we cannot compute a meaningful disparity value for the half-occlusion regions, we can see that the visible half of the half-occluded region generally looks similar to the background region rather than to the foreground region. This is because it is the background that is being occluded and hence not visible in the other image. In the pseudoscopic display this effect is reversed and the visible part of the half-occluded regions appears similar to that of the nearer object (the one with higher disparity). This is the case no matter whether we have the situation of a foreground occluding object or that of a hole. Thus, this method of pseudoscopy detection will work in situations where we have an equal preponderance of occluding objects and holes. The computational complexity of texture similarity that is needed for the half-occlusion texture comparison method is generally much greater than that of the left-right half occlusion difference method. When real-time operation is required our method would be more useful over the texture matching method.

Our method of detecting pseudoscopy is similar to Akimov *et al.* in terms of using half occlusions. However, we utilize a centroid based approach, which is computationally simpler, instead of computing edge maps and weighted distance. The first step of our method begins with the computation of disparity estimates. Following this we compute the left and right half occlusion regions of the image pair. The computation is performed following the same process as in Section4.2. Disparity estimation itself is a difficult problem to solve accurately and this dataset contains a lot of videos with regions of low textures which make the process even more

difficult. The poor estimation of disparity would carry over to the estimates of half occlusion which would worsen our ability to detect pseudoscopy. Therefore, we use the method discussed in Section 3.2.3 to identify regions with low texture and mask out those regions. This helps us to remove areas of poor disparity estimation from our consideration. Low textured regions are usually part of continuous surfaces or areas without depth discontinuities. Therefore, this masking process helps eliminate many false half occlusion detections as well. We used 52 videos over 5 different datasets [11, 13, 16, 20, 55] to test our methodology. Figure 4–9 shows the effect of masking out half occlusions to retain more reliable estimates from a single pair in our test set.



(a) Left image (b) Right image

(c) Half occlusion (Original) (d) Half occlusion (Filtered)

Figure 4–9: A single frame of a stereo pair and the effect of filtering half occlusion estimates using a blur mask. In (c) and (d) the black pixels are the left half occlusions and the white pixels are the right half occlusions.

The results of masked half occlusions are then used to form a histogram of the horizontal coordinates of the pixels marked as half occlusions. Using the histograms we can compute the centroids of the left and right half occlusions for each stereo pair. As mentioned before, if the difference of the left and right centroids is negative then the image pair is classified as stereo. For a positive difference the pair would be classified as pseudo. Figure 4–10 shows an example of such a histogram. The histogram was computed from a stereo image pair and we can see that the difference of the centroids would be negative indicating that the pair is presented stereoscopically.

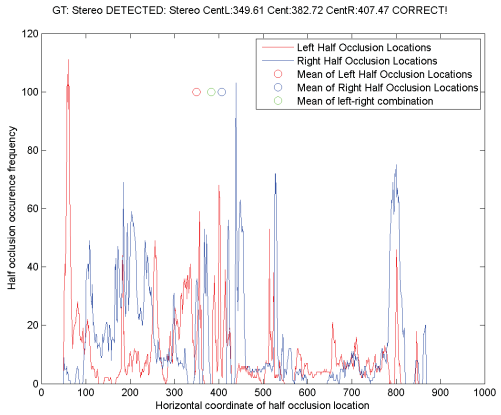


Figure 4–10: Histogram of half occlusion locations on the horizontal axis

The centroids are quite well separated in the histogram making the decision more obvious. However, in some cases, especially when the number of holes and foreground objects are similar the difference of the centroids would be smaller. The Recall/Precision curve in Figure 4–11 shows the effect of a threshold on the centroid differences on the performance of pseudoscopy detection.

It is important to note that even with the use of masking for half occlusions, the quality of initial disparity estimates still carry over to the half occlusion estimates.

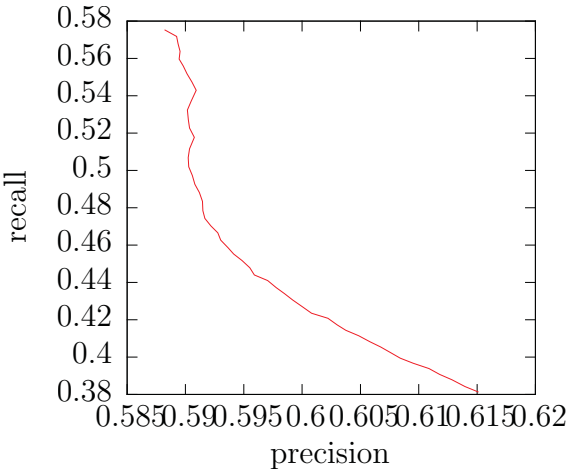


Figure 4–11: Recall vs. Precision of the pseudoscopy detection using the centroid method as a function of the centroid difference. The curve shown is computed over all the frames from each of the 52 videos.

Issues such as illumination changes, lack of texture, high frequency noises are present on certain videos within the dataset. For such videos, detection of pseudoscopy is poor for all frames over the video sequence. This accounts for the low precision range of 58%-61% seen in the curve. If we ignore some of the extreme cases of poor half occlusion detections our precision range improves significantly to 74%-81% as seen in Figure 4–12 and a smaller threshold is sufficient to make the decision. The videos with poor half occlusions were determined through qualitative assessment as well as looking at looking at precision-recall values that were lower than 5%. For some of these extreme cases, both precision and recall were 0% throughout the video. Approximately 10 videos out of 52 were considered to have very poor estimates of half occlusion.

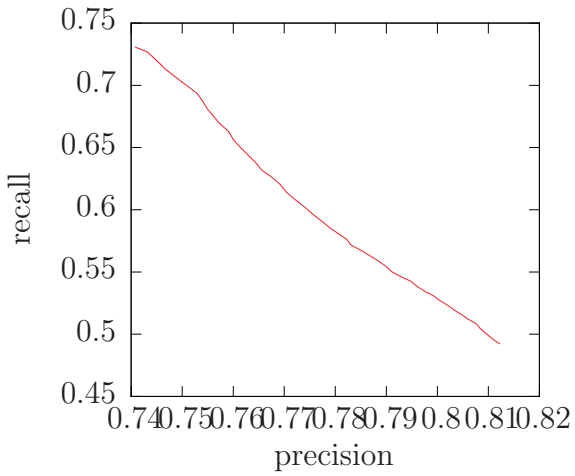


Figure 4–12: Recall vs. Precision after eliminating videos with a smaller video set. The curve shown is computed over all the frames from 42 different videos.

4.4.2 Detecting Depth Cue Conflict using Half Occlusions

Current movies generally contain significant amounts of Computer Generated Imagery (CGI). Introducing CGI requires careful work as the addition of new items into a scene will cause interactions with other items such as shadows and occlusion. Doing this for stereo content is even more challenging as these influences have to be correctly replicated in both views. Compositing can leave the content prone to the introduction of conflicting depth cues especially conflicts between occlusion and stereo cues. Figure 4–13 (a) shows a scene with a character in front of a background. Through compositing we wish to introduce another character which would be behind this original character but in front of the background. The composited image is scene in Figure 4–13 (b). The original character in the scene occludes the new one which in turn occludes the background. The composite image in Figure 4–13 (b) does not have any conflicts between occlusion and stereo cues at that point. However, if the

character from behind were to move up in the depth plane as part of the movie sequence while the character in front stayed at the same depth location it could create problems. The same problem could occur when inserting the new character with only the monoscopic occlusion cue in mind and not taking care of the stereo cue. Figure 4–14 shows a case of wrong insertion or character moving in front with the occlusion cue not adjusted to match that of the stereo cue. Based on the stereo disparity the object being occluded should appear in front of the character that is occluding it. This will create a occlusion-stereopsis conflict that ruins the depth perception and creates fatigue or discomfort.

(a) Original scene

(b) Scene with compositing

Figure 4–13: An example of stereo compositing without conflicts in stereo and occlusion cues.

Figure 4–14: Occlusion-Stereopsis conflict introduced in compositing

Half occlusions can provide useful information to detect these situations. Half occlusions themselves define object boundaries and hence exploring near those regions can identify conflicts. We compute the half occlusions in the original scene before doing any compositing. After inserting the new object, the disparity of the new scene is estimated. Following this we look in the regions of half occlusions from the original scene and check if there are updates to the depth estimates in those regions. If there are updates in those regions caused by the insertion we can compare its depth to that of the character in the original scene. This can help identify potential conflicts in depth cue. Figure 4–15 shows the analysis of this hypothesis on the left view of the images from Figures 4–13 and 4–14. Figure 4–15 (d) and 4–15 (f) represents the half occlusion regions fused on top of the depth map of the composite image. The pink regions highlight the half occlusions. Half occlusions regions are meant to be at a greater depth than the objects whose boundary they form. However, if we look at the pink regions in the image, we can see in Figure 4–15 (d) the depth being greater. Thus there is no conflict between the occlusion and the stereo cue. But the same cannot be said for the pink regions in 4–15 (f) where the depth in the half

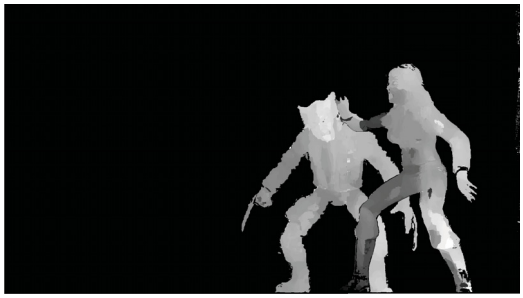
occlusion regions is closer than the occluding character. A conflict is created between the occlusion and the depth cue for this particular image. From these images we can see that half occlusions can be a useful tool for locating such possible conflicts.

(a) Disparity of scene from Figure 4–13(a)

(b) Half occlusion of scene from Figure 4–13(a)

(c) Disparity of scene from Figure 4–13(b)

(d) Half occlusions overlayed on disparity of composite image



(e) Disparity of scene from Figure 4–14

(f) Half occlusions overlayed on disparity of composite image

Figure 4–15: Half occlusions to indicate possible conflict in depth cues (Only the left view of the images are shown here)

CHAPTER 5
Conclusion

This thesis is focused on applying the knowledge from the field of computer vision to the production of good quality stereo movies. Stereo movies have made a recent comeback to the spotlight and although quite a few good movies have been released, there have also been ones that were not always up to the mark. Our motivation extends towards utilizing the computer vision techniques to automate some of the analysis of these stereo content. Additionally, we expect the techniques for analysis will also reveal important guidelines for producing better quality stereo movies. Therefore in the early part of the thesis we focused on the area of stereo vision which is most relevant for analysing stereo content. Along with the exploration of stereo vision, we also touch on the existing knowledge on stereo cinematography. This allowed us to identify common problems and ones which can be addressed using computer vision.

Our first contribution in the thesis is developing a method for detecting stereo window violation, which is one of the most common problems in stereo movies. It is caused by objects that appear in front of the screen and cross the borders of the screen. Most approaches attempt to correct stereo window violation at post production by utilizing techniques such as floating windows. Using such a method involves cropping which may not be favourable at all times for aesthetic reasons. Alternative to cropping would be shooting the scene again to accommodate necessary

objects or characters within proper boundaries. However, conflicts in schedules may prevent re-shooting a scene while the film is in post production and undergoing corrective measures. Moreover, current methodology requires input from trained and expert stereographers to identify such problems. This process itself may also be slow, expensive, and such experts are low in availability. Therefore we propose a method that automates the detection of stereo window violation. The proposed methodology is based on computer vision techniques as it begins by estimating the depth in a scene using binocular disparity. Areas of focus are also computed to determine regions of interest for viewers. The mean shift clustering technique is used to combine the previous computations to obtain segmentation of objects to track. By tracking objects that are in front of the viewing screen, we can automate the alert for stereo window violations. Although the contribution is aimed at 3D movies as its target application, the methodology can also be applied to other applications that require depth based segmentation or tracking. The results obtained showed very high reliability in detecting violations even with multiple objects being tracked.

Ideally we would like our solution for detecting stereo window violations to be available as a tool on the set of 3D movie shootings. For this purpose, our process must produce real-time performance in terms of its speed. Processing pairs of high definition frames and obtaining disparity estimates of sufficient quality is computationally expensive. The clustering and focus detection further adds to the complexity of the computation. The high complexity presents itself as a challenge against achieving real-time performance. This is one area of focus for future work. Some complexity can be reduced by down-sampling the input frames. Down-sampling would sacrifice

the resolution of the disparity estimation but the reduction in the size of the stereo pairs will greatly reduce computational burden. For this particular application, too great of a resolution is not required. Objects that are very close to the screen (with disparities of 1 or 2 pixels) would not cause huge depth jumps in the presence of stereo window violation. The break in depth perception would not be as significant as something which is much further inside the theatre space. Therefore, trading off a slight amount of disparity resolution to obtain greater speed would be useful. Additionally, the down-sampling process can help reduce some high-frequency noise in the images and lead to better disparity estimates. A further boost in speed can be achieved by utilizing a GPU based implementation on platforms such as CUDA. Some of our early attempts have shown great potential for speed up using such an approach.

Another major hindrance to good quality stereo content is the occurrence of conflicting depth cues. Stereo content can be created in a variety of ways all of which try to mimic our visual system and attempt to create a perception of depth. However, this depth perception is merely an illusion attempting to emulate the visual system as much as possible. With the focus mainly on stereo cues during the creation of 3D movies, some content may conflict with other cues such as occlusion, saturation, shadows, etc. Such problems are more likely to happen when 3D movies are created using 2D-to-3D conversion techniques or when compositing is required in 3D movies. The thesis also considered the presence of half occlusions which are present in all stereo content. We explore its role as a deterrent to obtaining more accurate disparity estimation. We also examined its usefulness in determining object

boundaries. Knowing object boundaries can help improve the disparity estimates as well as provide useful information for analysing other issues with stereo content. The half occlusion regions are identified using an iterative approach that estimates the disparity first and then the half occlusions. The estimates of half occlusion are then used to constrain the cost function used to compute the disparity. This constrained cost function yields , better results for disparity, which in turn is used to obtain better estimates for half occlusions. We proposed that half occlusions may be a useful feature for identifying occlusion and stereo cue conflict when performing compositing in stereo. We analysed this by creating test data to replicate such problems and initial testing indicates half occlusion may be useful for analysing the quality of stereo content. Knowing the half occlusions allowed us to identify if there was a conflict and also identify which regions to look into. There is room to explore this further, to understand if half occlusions can be leveraged to identify other problems or narrow the regions of interest to search for defects.

Pseudoscopy is another big defect in 3D movies that is not always easy to spot especially for those without much experience. However, it still causes strain to our eyes and visual system. Pseudoscopy is the presentation of stereo content where the left and right image pairs get swapped when being presented to the viewer's eyes. This causes the depth perception to be altered in a way that is not natural, introducing various conflicting cues as well. In our study we observed that half occlusions were a great feature in identifying this problem as well. For stereo content half occlusions maintain an ordering that is broken when the image pair is swapped.

We presented an approach that utilizes this property to identify whether a video is being presented pseudoscopically or stereoscopically.

Within this thesis we have seen how useful half occlusions can be. Half occlusion regions have been used to identify possible regions of conflicting depth cues as well as determine whether a video is being shown pseudoscopically. However, what we also encountered how difficult it is to obtain accurate half occlusion estimations. The poor half occlusion estimates have a significant effect on our results for finding pseudoscopy. Therefore, this is an issue that must be addressed in future work. The difficulty in obtaining good half occlusion estimates originates from the disambiguities in the matching process of disparity estimation. One issue that has repeatedly come up in this thesis is the lack of texture in images and videos. Low textures make it very difficult for most disparity estimation algorithms to perform reliable matching. Future work is needed to address this issue in stereo matching. The measure of texture should be included in the stereo matching process as a sort of weighting. This weighting can be used to influence whether the focus should be more on matching pixels across different views to estimate the disparity or instead utilize the disparity estimates of neighbouring pixels with good texture.

This thesis has focused on some of the issues related to stereo content creation for cinemas. Our aim was to provide the tools to detect such issues and generate knowledge that would allow for the creation of high quality stereo content. Doing so is vital to increasing the acceptance and the prevalence of 3D movies in cinemas. However, there are a variety of other problems related to stereo content creation that have not been addressed in this work. Some of which are known but lack proper tools

for automated detection. Future work in this area can focus on developing tools for identifying if the content contains excessive depth budget. The depth budget refers to the difference between maximum and minimum disparity in a scene defined by the furthest and closest object to the viewer. Too much depth either in front of the screen or behind the screen makes viewing uncomfortable. A big separation of objects in the depth plane due to high depth budget would make viewing uncomfortable. This is due to the big depth jump created as the viewer shift their attention between the objects. It is also important to avoid having big depth changes when there are scene transitions in a movie. Developing tools to detect and control these parameters would help the creation of good 3D movies. Future effort should also focus on ensuring each of these tools perform as close to real-time operation as possible. Real-time operations would enable the use of such tools on set. It would also encourage more cinematographers to make 3D movies especially those who were previously comfortable with and stuck to making 2D movies.

References

[1] D. Akimov, A. Shestov, A. Voronov, and D. Vatolin. Automatic left-right channel swap detection. In *3D Imaging (IC3D), 2012 International Conference on*, pages 1–6, Dec 2012.

[2] Martin S Banks, Jenny CA Read, Robert S Allison, and Simon J Watt. Stereoscopy and the human visual system. *SMPTE motion imaging journal*, 121(4):24–43, 2012.

[3] P. N. Belhumeur and D. Mumford. A bayesian treatment of the stereo correspondence problem using half-occluded regions. In *Computer Vision and Pattern Recognition, 1992. Proceedings CVPR'92., 1992 IEEE Computer Society Conference on*, pages 506–512. IEEE, 1992.

[4] J. Bouguet. Camera calibration toolbox for matlab, 2004.

[5] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(9):1124–1137, Sept 2004.

[6] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(11):1222–1239, Nov 2001.

[7] G. Bradski and A. Kaehler. *Learning OpenCV: Computer vision with the OpenCV library.* " O'Reilly Media, Inc.", 2008.

[8] D. K. Broberg. Guidance for horizontal image translation (hit) of high definition stereoscopic video production. In *IS&T/SPIE Electronic Imaging*, pages 78632F–78632F. International Society for Optics and Photonics, 2011.

[9] J. M. Brown and N. Weisstein. A spatial frequency effect on perceived depth. *Perception & Psychophysics*, 44(2):157–166, 1988.

[10] C. Chang, S. Chatterjee, and P.R. Kube. On an analysis of static occlusion in stereo vision. In *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR '91., IEEE Computer Society Conference on*, pages 722–723, Jun 1991.

[11] Eva Cheng, P Burton, Jonathan Burton, Alex Joseski, and I Burnett. Rmit3dv: Pre-announcement of a creative commons uncompressed hd 3d video database. In *Quality of Multimedia Experience (QoMEX), 2012 Fourth International Workshop on*, pages 212–217. IEEE, July 2012.

[12] D. Comaniciu and P. Meer. Mean shift: a robust approach toward feature space analysis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(5):603–619, May 2002.

[13] D. Corrigan, F. Pitie, V. Morris, A. Rankin, M. Linnane, G. Kearney, M. Gorzel, M. O'Dea, C. Lee, and A. Kokaram. A video database for the development of stereo-3d post-production algorithms. In *Visual Media Production (CVMP), 2010 Conference on*, pages 64–73, Nov 2010.

[14] B.G. Cumming and A.J. Parker. Responses of primary visual cortical neurons to binocular disparity without depth perception. *Nature*, 389(6648):280–283, 1997.

[15] G. Egnal and R. P. Wildes. Detecting binocular half-occlusions: empirical comparisons of five approaches. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(8):1127–1133, 2002.

[16] European Broadcasting Union. 3dtv public test set, June 2012.

[17] A. Fusiello, E. Trucco, and A. Verri. A compact algorithm for rectification of stereo pairs. *Machine Vision and Applications*, 12(1):16–22, 2000.

[18] B. R Gardner. The dynamic floating window: a new creative tool for 3d movies. In *IS&T/SPIE Electronic Imaging*, pages 78631A–78631A. International Society for Optics and Photonics, 2011.

[19] D. Geiger, B. Ladendorf, and A. Yuille. Occlusions and binocular stereo. *International Journal of Computer Vision*, 14(3):211–226, 1995.

[20] Lutz Goldmann, Francesca De Simone, and Touradj Ebrahimi. A comprehensive database and subjective evaluation methodology for quality of experience in stereoscopic video. In *Proc. SPIE*, volume 7526, pages 75260S–75260S–11, San

Jose, USA, 2010. International Society for Optics and Photonics. Available at `http://mmspg.epfl.ch/3dvqa`.

[21] S. Grossberg and N. P. McLoughlin. Cortical dynamics of three-dimensional surface perception: Binocular and half-occluded scenic images. *Neural Networks*, 10(9):1583 – 1605, 1997.

[22] M. J. Hannah. Computer matching of areas in stereo images. Technical report, DTIC Document, 1974.

[23] K. He, J. Sun, and X. Tang. Guided image filtering. In *Computer Vision–ECCV 2010*, pages 1–14. Springer, 2010.

[24] H. Hirschmuller and D. Scharstein. Evaluation of cost functions for stereo matching. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007.

[25] A. Hosni, M. Bleyer, C. Rhemann, M. Gelautz, and C. Rother. Real-time local stereo matching using guided image filtering. In *Multimedia and Expo (ICME), 2011 IEEE International Conference on*, pages 1–6, July 2011.

[26] S. S. Intille and A. F. Bobick. *Disparity-space images and large occlusion stereo.* Springer, 1994.

[27] T. Kanade, H. Kano, S. Kimura, A. Yoshida, and K. Oda. Development of a video-rate stereo machine. In *Intelligent Robots and Systems 95.'Human Robot Interaction and Cooperative Robots', Proceedings. 1995 IEEE/RSJ International Conference on*, volume 3, pages 95–100. IEEE, 1995.

[28] V. Kolmogorov and R. Zabin. What energy functions can be minimized via graph cuts? *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(2):147–159, Feb 2004.

[29] S. Kumar, C. Micheloni, C. Piciarelli, and G. Foresti. Stereo rectification of uncalibrated and heterogeneous images. *Pattern Recognition Letters*, 31(11):1445–1452, 2010.

[30] J. J. Little and W. E. Gillett. Direct evidence for occlusion in stereo and motion. *Image Vision Comput.*, 8(4):328–340, November 1990.

[31] G. Mather. Image blur as a pictorial depth cue. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 263(1367):169–172, 1996.

[32] B. Mendiburu. *3d TV and 3d cinema: tools and processes for creative stereoscopy*. Taylor & Francis, 2011.

[33] B. Mendiburu. *3D movie making: stereoscopic digital cinema from script to screen*. CRC Press, 2012.

[34] K. Nakayama and S. Shimojo. Da vinci stereopsis: Depth and subjective occluding contours from unpaired image points. *Vision Research*, 30(11):1811 – 1825, 1990. Optics Physiology and Vision A Festschrift Honoring Professor Gerald Westheimer on His 65th Birthday.

[35] N. D. Narvekar and L. J. Karam. A no-reference image blur metric based on the cumulative probability of blur detection (cpbd). *Image Processing, IEEE Transactions on*, 20(9):2678–2683, 2011.

[36] Y. Niu, F. Liu, W. Feng, and H. Jin. Aesthetics-based stereoscopic photo cropping for heterogeneous displays. *Multimedia, IEEE Transactions on*, 14(3):783–796, 2012.

[37] H. Ono, K. Shimono, and K. Shibuta. Occlusion as a depth cue in the wheatstone-panum limiting case. *Perception & psychophysics*, 51(1):3–13, 1992.

[38] R. P. O'Shea, S. G. Blackburn, and H. Ono. Contrast as a depth cue. *Vision research*, 34(12):1595–1604, 1994.

[39] C. Rhemann, A. Hosni, M. Bleyer, C. Rother, and M. Gelautz. Fast cost-volume filtering for visual correspondence and beyond. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 3017–3024, June 2011.

[40] T. W. Ryan, R. T. Gray, and B. R. Hunt. Prediction of correlation errors in stereo-pair images. *Optical Engineering*, 19(3):193312–193312, 1980.

[41] D. Scharstein. Matching images by comparing their gradient fields. In *Pattern Recognition, 1994. Vol. 1-Conference A: Computer Vision & Image Processing., Proceedings of the 12th IAPR International Conference on*, volume 1, pages 572–575. IEEE, 1994.

[42] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1-3):7–42, 2002.

[43] D. Scharstein and R. Szeliski. High-accuracy stereo depth maps using structured light. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 1, pages I–195–I–202 vol.1, June 2003.

[44] T. Smitley and R. Bajcsy. Stereo processing of aerial, urban images. In *Proc. Seventh Int. Conference on Pattern Recognition*, pages 433–435, 1984.

[45] A. Spoerri. The early detection of motion boundaries. Technical report, Massachusetts Institute of Technology, 1990.

[46] B. Su, S. Lu, and C. L. Tan. Blurred image region detection and classification. In *Proceedings of the 19th ACM International Conference on Multimedia*, MM '11, pages 1397–1400, New York, NY, USA, 2011. ACM.

[47] H. Su and B. He. A simple rectification method of stereo image pairs with calibrated cameras. In *Information Engineering and Computer Science (ICIECS), 2010 2nd International Conference on*, pages 1–4, 2010.

[48] N. Sugie. Neural models of brightness perception and retinal rivalry in binocular vision. *Biological Cybernetics*, 43(1):13–21, 1982.

[49] R. Szeliski. *Computer vision: algorithms and applications*. Springer Science & Business Media, 2010.

[50] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, Aseem Agarwala, M. Tappen, and C. Rother. A comparative study of energy minimization methods for markov random fields with smoothness-based priors. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(6):1068–1080, June 2008.

[51] H. Tong, M. Li, H. Zhang, and C. Zhang. Blur detection for digital images using wavelet transform. In *Multimedia and Expo, 2004. ICME '04. 2004 IEEE International Conference on*, volume 1, pages 17–20 Vol.1, June 2004.

[52] R. Trapp, S. Drüe, and G. Hartmann. Stereo matching with implicit detection of occlusions. In *Computer Vision-ECCV98*, pages 17–33. Springer, 1998.

[53] T. Troscianko, R. Montagnon, J. Le Clerc, E. Malbert, and P. Chanteau. The role of colour as a monocular depth cue. *Vision Research*, 31(11):1923–1929, 1991.

[54] E. Trucco and A. Verri. *Introductory techniques for 3-D computer vision*, volume 201. Prentice Hall Englewood Cliffs, 1998.

[55] M. Urvoy, M. Barkowsky, R. Cousseau, Y. Koudota, V. Ricorde, P. Le Callet, J. Gutierrez, and N. Garcia. Nama3ds1-cospad1: Subjective video quality assessment database on coding conditions introducing freely available high quality 3d stereoscopic sequences. In *Quality of Multimedia Experience (QoMEX), 2012 Fourth International Workshop on*, pages 109–114, July 2012.

[56] B. A. Wandell. *Foundations of vision*, volume 8. Sinauer Associates Sunderland, MA, 1995.

[57] O. Wang, M. Lang, M. Frei, A. Hornung, A. Smolic, and M. Gross. Stereobrush: Interactive 2d to 3d conversion using discontinuous warps. In *Proceedings of the Eighth Eurographics Symposium on Sketch-Based Interfaces and Modeling*, SBIM '11, pages 47–54, New York, NY, USA, 2011. ACM.

[58] J. Weng, N. Ahuja, and T. S. Huang. Two-view matching. In *ICCV*, volume 88, pages 64–73, 1988.

[59] A. J. Woods, T. Docherty, and R. Koch. Image distortions in stereoscopic video systems. In *IS&T/SPIE's Symposium on Electronic Imaging: Science and Technology*, pages 36–48. International Society for Optics and Photonics, 1993.

[60] A. L. Yuille and T. Poggio. A generalized ordering constraint for stereo correspondence. Technical report, Massachusetts Institute of Technology, 1984.

[61] L. Zhang, C. Vazquez, and S. Knorr. 3d-tv content creation: Automatic 2d-to-3d video conversion. *Broadcasting, IEEE Transactions on*, 57(2):372–383, June 2011.

[62] Y. Zhang, R. Hartley, J. Mashford, and S. Burn. Superpixels, occlusion and stereo. In *Digital Image Computing Techniques and Applications (DICTA), 2011 International Conference on*, pages 84–91, Dec 2011.