

Convergent Active Stereo: Investigating Depth Perception with a Robotic Binocular Camera System

Mingshi Chi

A Thesis Submitted to the Faculty of Graduate Studies in Partial
Fulfillment of the Requirements for the Degree of
Master of Applied Science

Graduate Program in Electrical and Computer Engineering
York University
Toronto, Ontario
August, 2025

©Mingshi Chi, 2025

Abstract

While extensive research exists on stereo disparity estimation algorithms for fixed parallel systems, there is limited exploration of active convergent stereo cameras. This thesis investigates the differences in depth perception between a robot head mimicking human eye movement with the ability to fixate and a static parallel setup. A novel convergent stereo algorithm, which is capable of searching for correspondences with diagonal epipolar lines, is proposed. To evaluate the algorithm, the first natural image convergent stereo dataset was created, which includes ground truth for parallel views, parallel images of each scene, and images of fixations across the scene. An in-depth analysis reveals the advantages and limitations of both systems, highlighting geometric benefits of active convergent stereo, especially in challenging scenes. The results from these findings can contribute to further the development of more effective vision-based depth perception system designs for humanoid robots.

Acknowledgements

Firstly, I would like to thank my supervisor, Professor John K. Tsotsos, for his guidance and expertise throughout this process. I've learned a lot about conducting research and choosing avenues to investigate while working together.

I would also like to thank my lab mates at various stages of my thesis. Frequent conversations early on about various aspects of robotics and computer vision allowed me to better conceptualize the whole picture of the problem of convergent stereo vision. Your supportive and encouraging feedback during lab meetings was very helpful in strengthening and improving my work.

To my supervisory committee, Professor Michael Jenkin and Professor Regina S. K. Lee, I thank you for your time and thorough review and feedback of my thesis. I appreciate the expertise your feedback brings.

I would finally like to thank my family, my mom and my dad, for supporting me unconditionally throughout not only my master's thesis but throughout the entirety of my life's education. Without your nurture and support, I would not have the same drive and motivation to expand my knowledge as I do today. To my dog, Cody, and my cat, Jack, I thank you for your company throughout the majority of my thesis. Being greeted with the same enthusiasm every day added that bit of joy and motivation every day. And of course, to my lovely partner, Juan, who was always a source of reason, happiness, and support throughout this process.

Table of Contents

Abstract	ii
Acknowledgements	iii
Table of Contents	iv
List of Figures	vii
List of Tables	xi
1 Introduction	1
1.1 Overview	1
1.2 Motivation	1
1.3 Human Visual System	2
2 Problem Statement	6
3 Related Work	8
3.1 Stereo Geometry	8
3.2 Stereo Matching Algorithms	12
3.3 Stereo Vision Datasets and Benchmarks	19
3.4 Convergent and Parallel Stereo Comparisons	22
3.5 Humanoid Robot Perception	24
3.5.1 Binocular Fixed Stereo Robot Heads	25
3.5.2 Robot Heads with Three Mechanical Degrees of Freedom	25

3.5.3	Robot Heads with Four Mechanical Degrees of Freedom	28
3.5.4	Robot Heads with Six Mechanical Degrees of Freedom	30
3.6	Summary	32
4	Methodology	34
4.1	Robot Head	34
4.1.1	Camera-Motor Calibration	35
4.1.2	Camera-Motor Calibration Data	38
4.1.3	Motor Consistency Tests	39
4.2	Image Data Collection	43
4.2.1	Experimental Setup	43
4.2.2	Finding the Parallel Configuration	44
4.2.3	Scene Setup	45
4.2.4	Ground Truth Using Structured Light	47
4.2.5	Dataset Fixations	52
4.3	Convergent Stereo Matching Algorithm	53
4.3.1	Determining the Fundamental matrix	53
4.3.2	Stereo matching algorithm	57
4.4	Summary	68
5	Results and Analysis	70
5.1	Overall results	71
5.1.1	Normal Objects Subset	83
5.1.2	Far Objects Subset	86
5.1.3	Close Subset	89
5.1.4	Smooth Planes Subset	92
5.1.5	TEOS Objects Subset	95
5.1.6	Repeated Patterns Subset	99
5.1.7	Dark Subset	102
5.1.8	Summary	106
5.2	Rectifying vs Not Rectifying	108

5.3	Depth Ordering	113
5.4	Random-dot Stereogram	114
5.5	Middlebury Parallel Images	117
5.6	Summary	119
6	Conclusion	121
6.1	Implications	121
6.2	Limitations	123
6.3	Future Work	125
6.4	Conclusion	126
	Appendices	128
	A Stereo Dataset	128
	B Image Size and Stereo Methods	133
	C Rectification Methods	135
	Bibliography	137

List of Figures

1.1	The muscles of the eye. Source: Encyclopaedia Britannica [1]	3
3.1	Parallel stereo configuration.	9
3.2	Parallel stereo configuration with a bird's eye view of the space that can be viewed in both cameras.	10
3.3	Convergent stereo configuration where both cameras are pointed at the center of the blue ball.	11
3.4	Convergent stereo configuration with a bird's eye view of the space that can be viewed from both cameras.	12
3.5	A representation of the fixations and head positions in the GENUA PESTO synthetic stereoscopic images dataset [2] with 3D fixations as red dots scattered across the scene.	21
3.6	Variation of depth uncertainty for stereo cameras with uniform pixel discretization in parallel (top image) and nonsymmetrical vergence angles (bottom image) taken from [3].	23
3.7	An illustration of the three degrees of freedom of the eyes that many robot heads exhibit. Each eye can independently pan but tilts as a unit.	26
3.8	An illustration of the four degrees of freedom of the eyes that many robot heads exhibit. Each eye can independently pan and tilt.	28
3.9	An illustration of the six degrees of freedom of the eyes that a few robot heads exhibit	31

4.1	Robot head with 3 degrees of freedom in the neck and 3 degrees of freedom for each camera, totaling 9 degrees of freedom.	35
4.2	Calibration method image centers visualized.	38
4.3	Calibration ChArUco boards placed around the field of view to ensure that calibration reaches all possible motor values	39
4.4	Experimental setup for data collection.	43
4.5	Parallel configuration board shown in the left and right cameras respectively.	44
4.6	The full dataset of 33 scenes showing one sample image from each, a left camera image from the parallel camera configuration.	46
4.7	Example images of projected Gray-code patterns onto a scene.	47
4.8	Example disparity outputs after decoding. Each output is the result of one set of 46 Gray-code images.	48
4.9	Example final ground truth disparity after combining disparity outputs of all decoded structured light sets.	49
4.10	Laser measurement errors taken with a LRF20M3PS laser module and a measuring tape. Human errors are expected to be included in these numbers.	50
4.11	Depth difference comparisons between depth projected from disparity values from structured light and depth measurements taken from the LRF20M3PS laser module.	51
4.12	Visualizing an example of the ground truth disparity map on the right and the fixations chosen for the scene on the left.	52
4.13	Epipolar lines plotted on an example fixation image pair.	54
4.14	Diagram of the convergent stereo matching algorithm, CAS (Convergent Active Stereo).	58
4.15	Detected SIFT keypoints (left) on a coarser level of the Gaussian image pyramid and its corresponding prior horizontal disparity map (right) for horizontal disparity after filtering matched points with the fundamental matrix.	60
4.16	The set of different empirically determined Gabor filters used on the left and right images before stereo matching.	62

4.17	An example fixation on a ChArUco board. Scene items are not scaled equally, as some items appear larger in the left image than in the right and vice versa.	64
4.18	The details of how horizontal and vertical disparities are turned into a depth map, which can be compared to the ground truth in the parallel configuration.	66
5.1	Diagram of convergent fixated images and the measurements used for analysis.	71
5.2	Diagram of the method used to aggregate errors for each pixel radius to fixa- tion category.	72
5.3	Diagram of parallel fixated images and the measurements used for analysis. .	73
5.4	Error distribution of all scenes divided into fixation areas.	74
5.5	Example outputs of disparity and depth error maps rotated to parallel camera configuration with fixations in the upper right corner.	76
5.6	Example outputs of disparity and depth error maps rotated to parallel camera configuration.	77
5.7	Example output disparity images of one of the scenes in the “Normal” subset.	83
5.8	Example output disparity images of one of the scenes in the “Far” subset. . .	86
5.9	Example output disparity images of one of the scenes in the “Close” subset.	89
5.10	Example output disparity images of one of the scenes in the “Smooth Planes” subset.	92
5.11	Example output disparity images of one of the scenes in the “TEOS Objects” subset.	95
5.12	Example output disparity images of one of the scenes in the “Repeated Pat- terns” subset.	99
5.13	A demonstration of the difficulty of determining correspondence in the “Re- peated patterns” subset.	100
5.14	Example output disparity images of one of the scenes in the “Dark” subset. .	103
5.15	Results of the rectified methods compared to CAS convergent non-rectified method.	110
5.16	The shapes used to test random-dot stereograms with their left and right images below.	115

5.17	Disparity output results on the donut random-dot stereogram	116
5.18	Example output disparity images of one of the scenes from the Middlebury stereo benchmark.	118
A.1	The full dataset of 33 scenes with their fixations visualized in red numbers on their parallel camera configuration left image.	129
A.3	The subset of scenes that has darker illumination.	129
A.2	The subset of scenes that contain objects that are close to the robot head. .	130
A.4	The subset of scenes that contain objects that are far from the robot head. Objects were placed at 1.5m or more away from the robot's cameras' baseline.	130
A.5	The subset of scenes that contain objects that had smooth planes without many features and textures.	131
A.6	The subset of scenes that contain objects that are typically found in stereo training sets and benchmarks.	131
A.7	The subset of scenes that contain objects that had repeated patterns. . . .	132
A.8	The subset of scenes which contain TEOS [4] self-occlusion blocks	132
B.1	ELAS results on different-sized images. Level 0 is the full-sized image, and every level is downsized by a factor of 2 from the previous level.	133
B.2	SGBM results on different-sized images. Level 0 is the full-sized image, and every level is downsized by a factor of 2 from the previous level.	134
C.1	Example rectification results from each method of finding the fundamental matrix.	136

List of Tables

4.1	The mean motor positional error of the left and right eye.	41
4.2	The mean motor rotational difference of the left and right eye between 3 sets of calibration.	42
5.1	Distribution of fixations in all scenes in the left image parallel view by dividing the left image into a 4×3 grid.	74
5.2	Absolute depth errors (m) and absolute disparity errors (pixels) across all scenes reported in mean, median, and STD values.	78
5.3	Evaluations of the chosen stereo methods with the percentage of bad pixels metric.	80
5.4	Evaluations of the chosen stereo methods with the percentage of bad depths metric.	82
5.5	Absolute depth errors (m) and absolute disparity errors (pixels) across the “Normal” objects subset reported in mean, median, and STD values.	84
5.6	Evaluations of the chosen stereo methods with the percentage of bad pixels metric on the “Normal” subset.	85
5.7	Evaluations of the chosen stereo methods with the percentage of bad depths metric on the “Normal” subset.	85
5.8	Absolute depth errors (m) and absolute disparity errors (pixels) across the “Far” objects subset reported in mean, median, and STD values.	87
5.9	Evaluations of the chosen stereo methods with the percentage of bad pixels metric on the “Far” subset.	88

5.10	Evaluations of the chosen stereo methods with the percentage of bad depths metric on the “Far” subset.	88
5.11	Absolute depth errors (m) and absolute disparity errors (pixels) across the “Close” objects subset reported in mean, median, and STD values.	90
5.12	Evaluations of the chosen stereo methods with the percentage of bad pixels metric on the “Close” subset.	91
5.13	Evaluations of the chosen stereo methods with the percentage of bad depths metric on the “Close” subset.	91
5.14	Absolute depth errors (m) and absolute disparity errors (pixels) across the “Smooth Planes” objects subset reported in mean, median and STD values. .	93
5.15	Evaluations of the chosen stereo methods with the percentage of bad pixels metric on the “Smooth Planes” subset.	94
5.16	Evaluations of the chosen stereo methods with the percentage of bad depths metric on the “Smooth Planes” subset.	94
5.17	Absolute depth errors (m) and absolute disparity errors (pixels) across the “TEOS Objects” objects subset reported in mean, median and STD values. .	96
5.18	Evaluations of the chosen stereo methods with the percentage of bad pixels metric on the “TEOS Objects” subset.	97
5.19	Evaluations of the chosen stereo methods with the percentage of bad depths metric on the “TEOS Objects” subset.	98
5.20	Absolute depth errors (m) and absolute disparity errors (pixels) across the “Repeated Patterns” subset reported in mean, median, and STD values. . .	101
5.21	Evaluations of the chosen stereo methods with the percentage of bad pixels metric on the “Repeated Patterns” subset.	101
5.22	Evaluations of the chosen stereo methods with the percentage of bad depths metric on the “Repeated Patterns” subset.	102
5.23	Absolute depth errors (m) and absolute disparity errors (pixels) across the “Dark” subset reported in mean, median, and STD values.	104
5.24	Evaluations of the chosen stereo methods with the percentage of bad pixels metric on the “Dark” subset.	105

5.25	Evaluations of the chosen stereo methods with the percentage of bad depths metric on the “Dark” subset.	105
5.26	Absolute depth errors (m) and absolute disparity errors (pixels) across all scenes	109
5.27	Evaluations of the chosen stereo methods with the percentage of bad pixels disparity metric across all scenes, including outputs of CAS, ELAS, and SGBM on rectified convergent images.	112
5.28	Evaluations of the chosen stereo methods with the percentage of bad depths metric across all scenes, including outputs of CAS, ELAS, and SGBM on rectified convergent images.	113
5.29	Results of the percentage of fixation pairs with the correct depth ordering in each set of scenes.	114
5.30	Stereogram disparity analysis with percentage of bad pixels metric.	116
5.31	Stereogram disparity analysis with mean disparity error, median disparity error, and disparity error standard deviation metrics.	117
5.32	Middlebury disparity analysis with mean disparity error, median disparity error metrics.	118
5.33	Middlebury image disparity analysis with percentage of bad pixels metric. . .	118

Chapter 1

Introduction

1.1 Overview

Binocular stereo vision has been an active field of research in computer vision and robotics for many years. However, human binocular depth perception, which has various mechanical degrees of freedom in the eyes, behaves differently from machine binocular depth perception, which is commonly performed using a fixed pair of cameras that do not move in relation to each other. Human depth perception involves the coordinated motion of two eyes with a large amount of fixation points, and it is an active system that performs well for the daily tasks we need it for. Although there exists a large literature in both the biological and computational aspects of binocular vision, there has been very little research that investigates the differences between human binocular depth perception and machine binocular depth perception, along with the advantages and disadvantages of each.

1.2 Motivation

The motivation for this thesis stems from the gap in research where the differences in depth perception have not been explored between a robot head with cameras that have the same degrees of freedom as human eyes and a traditional fixed parallel stereo camera pair. Human vision, often more efficient and dynamic than computer vision pipelines today, operates on 6 mechanical degrees of freedom (DoF) of the eyes, with three for each eye, and

various motions which support visual perception. Investigating depth perception in robots with similar active abilities of the cameras can reveal geometric advantages in either setup and can inform us of algorithmic and mechanical design decisions for vision-based depth perception in robotics in the future.

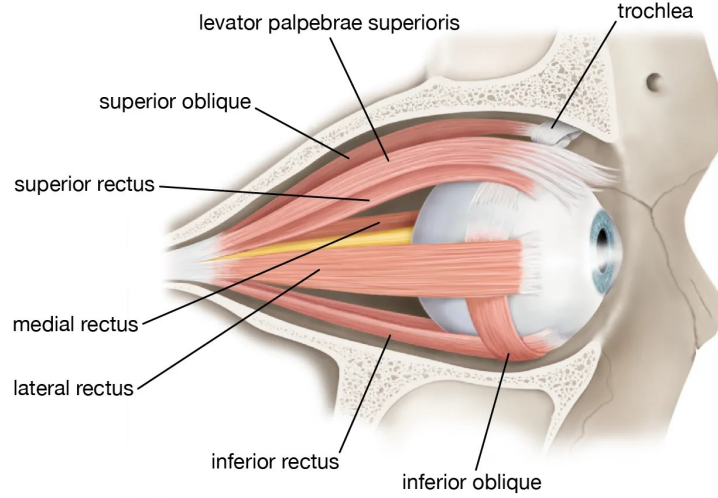
Current robotic systems often rely on fixed parallel stereo cameras for vision-based depth perception. Existing research in stereo disparity estimation algorithms predominantly focuses on fixed parallel stereo cameras for 3D reconstruction. There is a gap in exploring the potential benefits of active convergent stereo camera configurations, particularly in mobile robot heads and humanoid robots. Humanoid robots have been a popular topic of research for roboticists, computer vision scientists, and psychologists. They are often designed to mimic human features and capabilities to allow the robot to perform human tasks or to deploy cognitive science models of information processing or decision-making for verification purposes. While many robot heads are only equipped with fixed parallel stereo cameras very few enable depth map computations with active convergent cameras, and to my knowledge, none evaluate the outputs with ground truth measurements nor analyze the advantages of an active convergent stereo camera system with respect to the 3D points recovered from depth map estimates. Comparing the results of parallel stereo and convergent stereo would be important to reveal any flaws and strengths each system has.

1.3 Human Visual System

The human visual system includes two eyes that have three degrees of freedom due to 6 muscles. The human eyes have six extraocular muscles that control the movements of each eye. These muscles, shown in Figure 1.1, involve four rectus muscles: the superior rectus, which primarily elevates the eye, the inferior rectus, which primarily depresses the eye, the medial rectus, which controls adduction, and the lateral rectus, which controls abduction. Two oblique muscles control the torsional movements of the eye: the superior oblique, which rolls the eyes inward, and the inferior oblique, which rolls the eyes outwards. These muscles work together to provide the various movements of the eyes [1, 5].

There are a variety of ways that eye motions support depth perception, including control-

Muscles of the right eye



© Encyclopædia Britannica, Inc.

Figure 1.1: The muscles of the eye. Source: Encyclopaedia Britannica [1]

ling vergence to align the retinas for optimal binocular stereopsis [6]. Binocular stereopsis refers to the process responsible for depth perception based on small disparities between the images viewed from each eye. Disparity is the difference in image location of an object seen by the left and right eyes. Additionally, when the body or head is in motion, particularly when the head is tilted, the eyes naturally roll in the opposite direction to compensate for the head movement and helps maintain a stable visual field [7].

Stereopsis was studied by many nineteenth-century scientists, such as Helmholtz and Hering. Helmholtz's work included studying the horopter and demonstrating that vertical disparities are used to calibrate depth perception from horizontal disparity [8]. When the horizontal disparity exceeds a certain value, binocular fusion is broken and two images are seen, a sensation referred to as diplopia [9]. Panum's fusional area describes the region around the horopters (regions in space that project to corresponding retinal locations and appear at the same location when viewed monocularly in each eye) within which binocular fusion can still occur, allowing for slight disparities to be fused without causing diplopia [10, 11]. While computer vision commonly only considers horizontal disparity, vertical disparity combined with a variety of cues is also important for human depth perception. Backus et al.[12] found that human observers cannot determine a slant on a surface by horizontal disparity alone.

In addition to horizontal disparity, they also rely on both the vertical size ratio (VSR, a common measure of vertical disparity is a ratio between the vertical angles subtended by a surface patch at the left and right eyes) as well as the sensed eye position to interpret the measured horizontal disparity, giving more weight to VSR when the two are placed in conflict.

[13] compares psychophysical methods to determine the perceived egocentric distance. In general, it is agreed that the psychophysical power function relating perceived distance D' , and physical distance, D is in its simplest form: $D' = KD^n$ where K is a constant defining the scale unit, and the exponent n is a parameter that affects the acceleration of the function relating the distance estimates to physical distance. n also indicates the degree of distance constancy, where if $n > 1.0$, there is a positive acceleration and indicates overestimation of distance, if $n < 1.0$, there is a negative acceleration and indicates underestimation of distance, and if $n = 1$, then there is perfect distance constancy. Although the reported n exponent depends on the psychophysical method used and the environment, as well as many other factors, it was generally agreed upon that distance estimates became worse as physical distance to the object increased. For the majority of methods surveyed, the exponent was less than 1.

Tresilian et al. [14] investigate the contribution of eye vergence to visual distance perception in near space up to 2-3m. Using ophthalmic prisms to perturb the vergence cue while leaving the other cues unaffected, the study found that the prisms reliably influenced distance estimates. There was a strong contraction bias (specific distance tendency) that was observed when cues like vergence or size were available in isolation, but the bias progressively disappeared as more retinal information was added. The contribution of vergence cues was found to decrease as target distance and other retinal information increased. The authors note that the relationship with target distance is likely due to decreasing reliability and larger uncertainty of vergence angle distance indication at larger distances. They concluded that vergence cues play a significant role in distance perception in near-space perception.

The multiple degrees of freedom utilized by the human eyes are well studied and are crucial for achieving accurate depth perception and visual stability while mobile. Each eye's ability to move allows for proper alignment for binocular fusion and stereopsis. It would

be interesting to investigate the value of the human ability for mobile eyes and the depth estimations from convergent images in the context of computer vision. A robotic system will be used in this thesis to investigate these differences. The aspects of human vision's connection to the robotic system consist of the ability to fixate on different parts of the scene and the importance of vergence. Neurons, brain areas, or other psychophysical relationships will not be investigated in this thesis.

Chapter 2

Problem Statement

In human vision, it is evident that the ability to move the eyes play an essential role in how humans perceive the world. However, there is a lack of research around the performance differences and possible advantages of active convergent stereo systems with the ability to fixate on different parts of the scene compared to fixed parallel stereo systems. Current stereo cameras are often fixed and parallel, and most popular stereo matching methods are based on the assumption that the cameras are fixed and parallel. Additionally, there are no stereo benchmarks with natural images that provide both parallel and convergent image pairs as well as ground truth disparity to compare with. Furthermore, of the existing stereo benchmarks, most images are filled with many objects and textures that heavily aid stereo matching algorithms without specific sets to evaluate difficult scenarios that stereo algorithms commonly fail on.

The work conducted in this thesis involves the development and evaluation of depth perception using a binocular stereo head designed to provide kinematics and dynamics similar to those found in biological systems. This thesis aims to address the following research questions:

- RQ1.** How do the resulting depth maps of fixed parallel stereo compare to convergent stereo with multiple fixation points in terms of accuracy and map fidelity in various regions?
- RQ2.** Is there an advantage to having the ability to fixate on different areas in the scene for depth perception? If so, in which scenarios are the added degrees of freedom

particularly useful?

RQ3. What are the consequences of rectifying convergent images to be parallel, and is it better for depth accuracy to perform disparity calculations using original convergent images?

To answer these questions, a novel stereo matching algorithm that can be used with any convergent stereo head is proposed. The algorithm is named Convergent Active Stereo (CAS). Although the use-cases of cyclovergence will not be explored, the algorithm is applicable in scenarios where cameras exhibit cyclovergence. Its performance is evaluated against state-of-the-art deep learning and traditional stereo methods. To evaluate the proposed stereo matching algorithm, a novel dataset is created of various objects and scenes that not only test stereo matching in the presence of an abundance of reliable image information, but also under common challenging conditions.

The remainder of this thesis is organized as follows. Chapter 3 provides an overview of the related works, including stereo geometry, stereo algorithms, previously constructed robot heads and their perception system, and convergent and parallel stereo analyses. Chapter 4 details the process of data collection and the proposed stereo algorithm. Chapter 5 gives an overview on experiments and results. Finally, Chapter 6 concludes the thesis.

Chapter 3

Related Work

3.1 Stereo Geometry

Figure 3.2 illustrates the traditional parallel stereo configuration. The diagrams show two pinhole cameras with their optical axes parallel to each other and their image planes coplanar, the defining characteristic of a parallel stereo configuration. Conventionally, the global frame of reference is the left camera’s frame of reference, and disparity and depth are computed in this frame. In psychophysics and in human studies, depth is often reported in terms of cyclopean depth or cyclopean disparities [15]. This information can be obtained if desired but for the remainder of this document, depth and disparity are reported in the left camera’s reference frame. Figure 3.1 illustrates a 3D scenario of viewing blocks and shows an example of the views from each camera. Notice that the red dashed epipolar lines in both the left and right images fall at the same vertical height as each other and slice through the same corresponding points. The epipolar constraint dictates that for any point in one image, its corresponding point in the other image must lie on its corresponding epipolar line. In the case of parallel stereo, with the left camera as the reference frame, all points viewed by both cameras will have a negative disparity, with the points in the right image appearing to the left of the points in the left image. This is shown in Figure 3.2. Because of this parallel configuration, the search space for corresponding pixels is simplified immensely. The search space is only in the negative direction along one line. Some traditional stereo methods typically will use a target pixel in the left image’s surrounding patch as a template

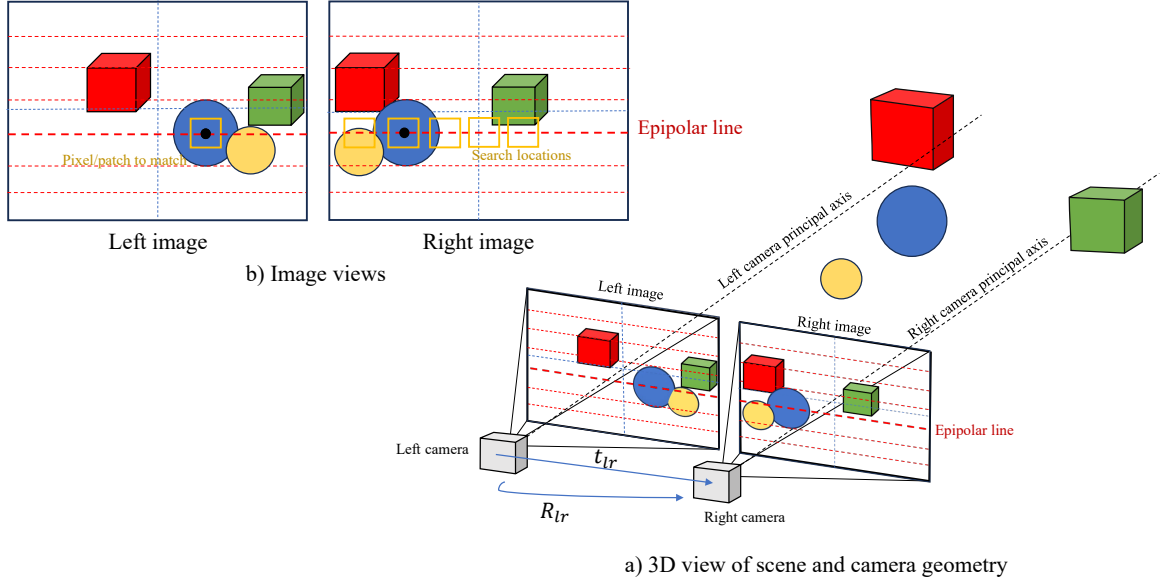


Figure 3.1: Parallel stereo configuration. a) shows the 3D view of the scene and the parallel camera geometry. R_{lr} is the rotation that aligns the left camera’s frame of reference to the right camera’s frame of reference orientation, and t_{lr} is the translation that moves the left camera’s frame of reference to the right camera’s frame of reference. The principal axes of both cameras are shown with a dotted line extending from the camera center, through the image center, and outwards. The principal axes are parallel. b) shows the images obtained from both cameras while viewing the scene displayed in a). The epipolar lines are horizontal and on the same vertical level, and the search locations in the right image for a target pixel in the left image are directly to the left of the left pixel’s location in the horizontal direction. If using a patch matching method, the patch sizes are the same for every search location.

to match along the epipolar line in the right image. The search patches in the right image, as a result of parallel epipolar lines, are the same shape and size as the patch extracted from the left image.

In a convergent stereo configuration the epipolar lines are not parallel, rather, they meet at epipoles that are not placed at infinity. Unlike the parallel configuration, if the cameras both fixate on one common 3D point, the optical axes of the two cameras will intersect at a point in 3D space, called the fixation point. Convergent configurations are shown in Figure 3.3 and Figure 3.4. Notice that the red dashed epipolar lines are no longer horizontal and parallel to the image planes. This means that for a point in the left image, its corresponding epipolar line in the right image is a line that passes through the right camera’s

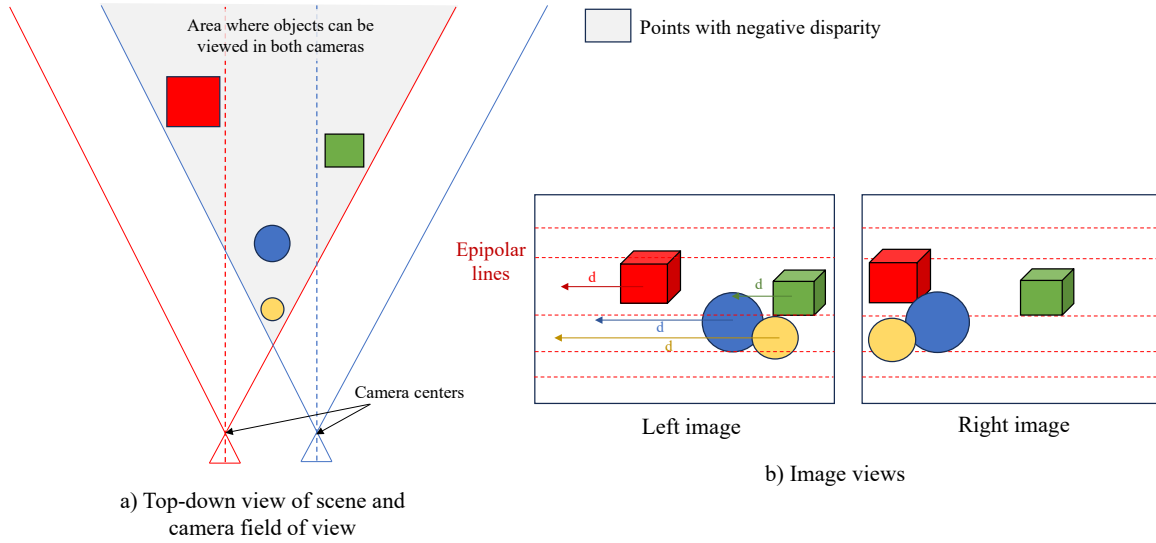


Figure 3.2: Parallel stereo configuration with a bird's eye view of the space that can be viewed in both cameras. a) shows the top-down view of the scene with the solid lines marking the limits of the cameras' fields of view and the dotted lines marks the principal axes. b) shows the images obtained from both cameras when viewing the scene. All objects seen in both cameras will exhibit a negative disparity when matched between images, as corresponding points will appear more to the left in the right image. 'd' arrows indicate the direction of the object's disparity between the left and right images.

epipole, which is not placed at infinity, unlike in parallel configurations. A consequence of this is that corresponding objects may not have the same shape or size when viewed between the left and right images. Additionally, as can be seen in Figure 3.3, the patch matching method would not work well in this configuration, as each patch in a new search location would have a different shape and size. To accurately use a patch matching method, one would need to compare patches of different sizes and shapes, which would introduce a large amount of computational complexity. In Figure 3.4, the cameras are fixated on the blue ball. The region in front of this ball, which the yellow ball resides within, is an area that guarantees negative horizontal disparity, like in parallel images. The points that fall beyond the fixation point will exhibit positive horizontal disparity, unlike typically negative horizontal disparities in the parallel configuration. As can be seen in Figure 3.4, only the yellow ball, which resides in the negative horizontal disparity area, will have its corresponding points in

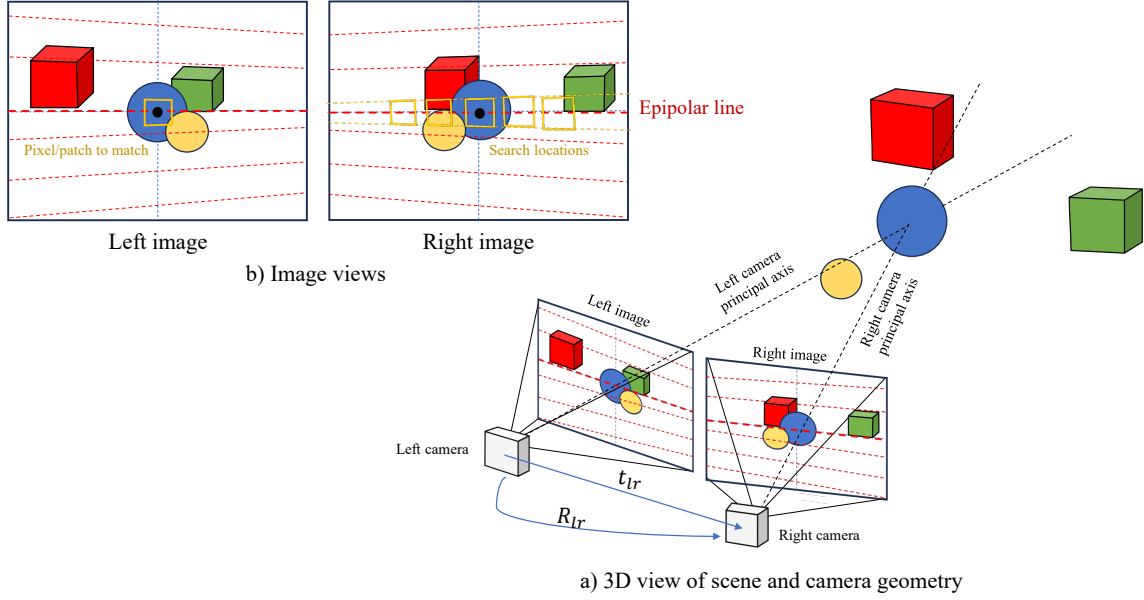


Figure 3.3: Convergent stereo configuration where both cameras are pointed at the center of the blue ball. a) shows the 3D view of the scene and the parallel camera geometry. R_{lr} is the rotation that aligns the left camera's frame of reference to the right camera's frame of reference orientation, and t_{lr} is the translation that moves the left camera's frame of reference to the right camera's frame of reference. The principal axes of both cameras are shown with a dotted line extending from the camera center, through the image center, and outwards. The principal axes meet at the fixation point. b) shows the images obtained from both cameras while viewing the scene displayed in a). Epipolar lines are not horizontal, and the search locations can be in either direction. If patch matching is desired, the patch sizes and shapes would be different for each search location.

the right image to the left of its location in the left image. The other objects have their corresponding points in the right image to the right of their location in the left image. In practice, there is a threshold where positive horizontal disparity turns into negative horizontal disparity with a zero disparity crossing. This zero horizontal disparity crossing area is centered about the fixation point and appears similar to the horizontal horopter in humans, an area which also exhibits zero horizontal binocular disparity. The horizontal and vertical disparity at the fixation point will be zero. The area and shape of the region with negative horizontal disparity will change depending on the fixation point, and so will the area and shape of the region with positive horizontal disparity.

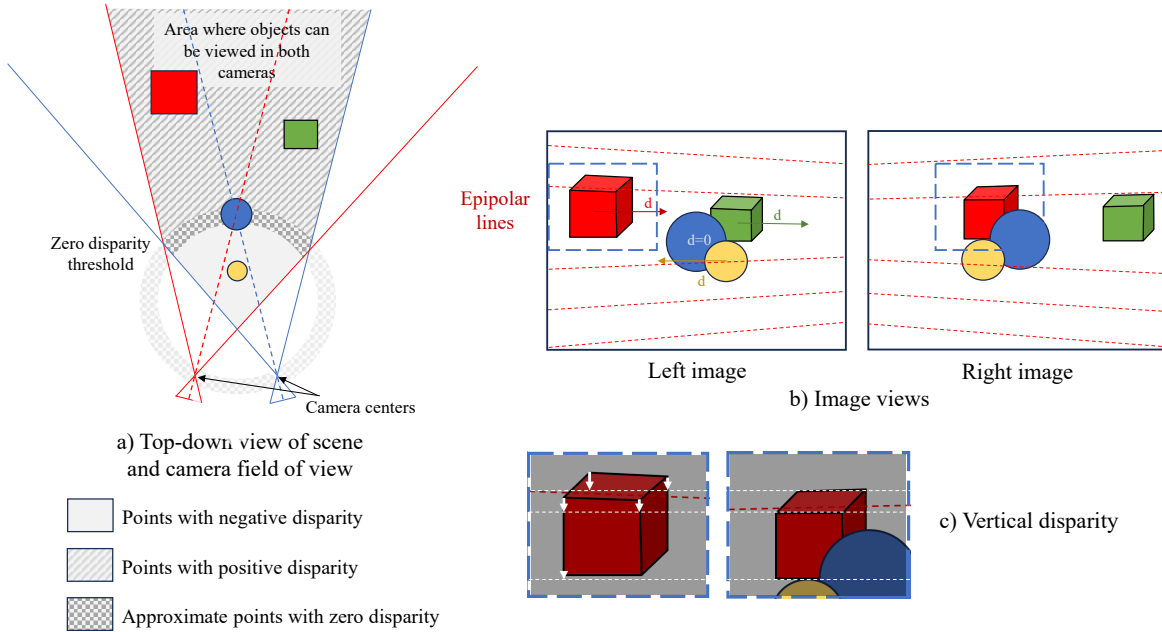


Figure 3.4: Convergent stereo configuration with a bird's eye view of the space that can be viewed from both cameras. a) shows the top-down view of the scene with the solid lines marking the limits of the cameras' fields of view, and the dotted lines marking the principal axes. The area with positive horizontal disparity are marked with a striped gray pattern. The area with negative disparity are marked with light gray. In practice, there will be an threshold centered about the fixation point where the positive disparity crosses over to negative disparity. Disparity at this threshold will be zero. In this diagram, the threshold is approximated. b) shows the images obtained from both cameras when viewing the scene. The cameras are fixated on the blue sphere. A fixation is achieved when both cameras' principal axes intersect at the same 3D point. 'd' arrows indicate the direction of the object's horizontal disparity. Note that in a convergent stereo configuration, there will be vertical disparity as well. Disparity at the fixation point is zero. c) shows a case where vertical disparity is evident. It shows a zoomed-in version of the blue dotted windows in both images seen in b). The dotted windows are on the same vertical height as each other, with a difference in horizontal translation. The white arrows show the visible vertical disparity that occurs in non-parallel camera configurations.

3.2 Stereo Matching Algorithms

Most stereo vision algorithms in the past have been developed for fixed parallel stereo camera configurations as can be seen in surveys and reviews such as in [16], [17], [18], [19], [20], [21], and [22]. Stereo vision algorithms seek to solve the correspondence problem such that for every pixel in the left view within the overlapping scene area, which can be viewed in both cameras, there is a unique corresponding pixel found in the right view to create a dense

disparity map. The choice of an adequate stereo-matching algorithm can also depend on the specific application requirements and constraints, such as time and resource limitations.

The history of human-inspired stereo matching algorithms is rooted in attempting to understand and replicate how the human visual system processes binocular disparity to perceive depth. Early ideas believed that in human vision, object recognition preceded the establishment of binocular correspondence [23]. However, in 1960, Julesz invented the random-dot stereogram, which demonstrated that stereopsis can occur solely based on disparity alone and in the absence of identifiable objects [24]. With this, Marr and Poggio developed computational theories of human stereo vision and proposed a 3D network of neurons that iteratively refined disparity estimates based on uniqueness and continuity constraints [25]. The same authors later proposed a different algorithm with psychophysics inspiration, which used sparse matching zero-crossing primitives at different spatial resolutions, implementing a coarse-to-fine refinement strategy [26]. Bandpass filters were also introduced as a first step before attempting to match correspondences. The biological inspiration for these bandpass methods comes from experiments that showed that the visual cortex encodes information using bandpass spatial frequency filters [27, 28, 29]. Some methods that use bandpass filters rely on the phase information extracted from bandpass-filtered versions of stereo image pairs. Instead of comparing pixel intensities or matching explicit features, these methods focus on analyzing the local spatial frequency content of images and the phase relationships between the corresponding points [30, 31, 32, 33].

Langley et al. [34] provided a method for interpreting disparity using phase differences as an extension of Marr’s work, and can obtain large disparity estimates. They introduced strategies of smoothing disparity estimates and employing a simulated mechanism for eye vergence to reduce phase error, resolve aliasing, and bring features into better correspondence. The effect of this simulated vergence is to bring features nearer in correspondence, implying shifting the image at parts to bring correspondence estimates closer to avoid phase errors at large disparities. The technique was also extended to 2D using orientationally selective filters, but it introduced a stereoscopic aperture problem where ambiguities exist at single edges. The aperture problem means that there is ambiguity in determining the disparity in image regions that contain only singly directional features, such as single edges, and

in these regions, disparity can only be reliably measured in the direction perpendicular to the plane. The aperture problem is best resolved in image regions with corner-like features. Thus, their algorithm does not evaluate vertical disparity at edges. Furthermore, only one example disparity image of their 2D algorithm was shown, and accuracy was not reported.

Aside from human-inspired stereo matching methods, many methods have been created to tackle the correspondence matching problem. Traditional stereo vision algorithms are often categorized into local and global methods. Local methods are also known as area-based or window-based approaches and determine the disparity of a pixel based on the intensity values within the window surrounding the pixel in the left image and the candidate window in the right image. The process for local methods generally involves 4 stages: matching cost computation, cost aggregation, disparity optimization, and disparity refinement [16]. Local methods have lower computational complexity but can be less accurate in texture-less regions or near depth discontinuities. They often suffer from noise sensitivity and variations in intensity that occur in challenging regions [18].

Global methods, on the other hand, take into consideration the entire image when estimating disparity. They are often more computationally heavy but result in more accurate disparity maps. Global methods treat disparity estimation as an energy minimization problem and aim to assign disparity values that minimize the global energy function, often utilizing consistency and smoothness constraints. Some global methods will take advantage of feature extraction and solving the correspondence problem between features. Global methods often perform better in challenging areas such as textureless regions, occlusions, and depth discontinuities [18].

Traditional methods often build upon matching constraints, some of which occur due to the parallel configuration of the cameras. For intensity-based approaches, the matching pixels are assumed to have similar intensity values or the matching windows must be highly correlated. The uniqueness constraint states that almost always, given a pixel or feature from one image, it can match no more than one pixel or feature from the other image unless transparent objects are present. Another constraint is the continuity constraint where disparity of matches should vary smoothly unless at a depth discontinuity. The epipolar constraint [35] states that given a pixel m in the left image, its corresponding pixel m' in the

right image must lie on the epipolar line which is conveniently at the same height as m and is strictly horizontal in parallel stereo configurations. The ordering constraint states that if m and m' are corresponding pixels and n and n' are another pair of corresponding pixels, and if m is to the left of n , then m' should also be to the left of n' . This constraint holds true in most places other than areas known as the forbidden zone [35] where the constraint fails. The forbidden zone is where if in the left image m is to the left of n , but in the right image, m is to the right of n . This occurs when a 3D point n is in the area, which is constrained by the line connecting the left image center to the 3D point m and the line connecting the right image center to the 3D point m . In Figure 3.2, if the blue ball is m and the yellow ball is n , then n would be inside the forbidden zone. If the red square is m and the green square is n , then this constraint holds true. With these assumptions, these methods cannot compute convergent images even without the horizontal epipolar line constraint, as it is not guaranteed that corresponding pixels occur ordered in both images, since points in 3D past the fixation point will have disparity in the opposite direction.

One widely used traditional method is Semi-Global Matching (SGM) [36]. This method starts with a pixel-wise cost calculation based on mutual information using a hierarchical approach. The cost is aggregated, and continuity constraints are enforced, leading to a disparity measure at the minimum aggregated cost. The disparity is then refined with a consistency check to detect occlusions and false matches. The method can achieve comparable accuracy to other global methods at the time and is robust to illumination changes; however, it is sensitive to parameter tuning. The modified version of SGM has been integrated into the OpenCV library [37] as Semi-Global Block Matching (SGBM). The method differs from the original SGM method by only considering 5 directions instead of 8 when searching for a match. The modified method also matches blocks instead of individual pixels, and the mutual information cost function has been substituted with a less expensive sub-pixel metric [38]. If applied to non-parallel epipolar lines, SGBM’s patch matching technique would hold an incorrect assumption that patches of corresponding pixels would contain the same or very similar information and structure. In a convergent camera configuration, the epipolar lines are not parallel, and there are perspective differences, which may make corresponding pixel patches to be different sizes and shapes.

Another commonly used traditional method is ELAS [39]. With its fast processing, it is often used in robotic applications [40]. The method builds a prior 2D mesh of disparities by first identifying a set of sparse matched ‘support points’ detected using horizontal and vertical Sobel filters. A 2D mesh is created over these points using Delaunay triangulation to estimate the disparities of the prior. The prior helps to reduce the search space for matching the remaining pixels. The image likelihood is modeled as a Laplace distribution based on the L1 distance between feature vectors of the left and right images. Then, a maximum a posteriori estimation is used to compute the disparity map, and this can be done in parallel, reducing computation time. However, this method is sensitive to parameters and relies on an accurate and distributed set of initial support points.

Deep learning approaches were first introduced in [41] and today deep learning methods have offered significant improvements over traditional approaches on benchmarks. Categories include non-end-to-end learning, using CNNs to improve specific components of the pipeline, end-to-end, replacing the entire stereo pipeline, and unsupervised learning, alleviating reliance on ground truth disparity data for training [20]. Although deep methods obtain a higher accuracy on benchmarks than traditional methods, they may not be as reliable, as it is difficult to understand the process these models use to assign disparity. With traditional methods, the failure points and the process of assigning disparity values are well understood and predictable. Deep learning methods often fine-tune on a particular dataset before submitting for benchmark evaluation, which may raise concerns over their ability to generalize to other data. Additionally, they are often extremely computationally and memory-intensive [18] and require a large amount of data to train. Often, these networks are also trained on synthetically generated data with known ground truth, such as in Blender [42].

The results of deep learning methods on convergent setups have not been investigated. Common training datasets are derived from fixed parallel stereo camera configurations. In practice, if a stereo setup is not strictly parallel, the cameras are calibrated, and their images are rectified so that the epipolar lines are parallel. Since training datasets are typically constructed using parallel camera systems, data-driven supervised learning methods would not be exposed to these types of rectified images. Additionally, the parallel configuration

shown in Figure 3.2 indicates that there is only negative horizontal disparity between the left and right images. In a convergent stereo system, shown in Figure 3.4, there would be both negative and positive horizontal disparity as well as vertical disparity. Since deep learning methods are only exposed to parallel image pairs, they are not able to assign disparities in the opposite direction nor account for vertical disparity.

Teed and Deng [43] formulated the deep network architecture RAFT: Recurrent All-pairs Field Transforms for optical flow. It consists of 3 main components, which involve a feature encoder that extracts per-pixel features from both input images and a context encoder that extracts features only from the first image, a 4D multiscale correlation volume built with the feature vectors, and an update operator that iteratively updates a flow field. This method was then modified for specific stereo vision applications in [44], and at the time, it outperformed the next best method on the percentage of bad pixels over 1 pixel error metric by 29%, which inspired many other deep learning methods after it to use similar architectures [45, 46]. The main modifications in RAFT-Stereo are the correlation volume being 3D, which is due to the epipolar horizontal constraint and the update operator. While RAFT uses a recurrent GRU-based update operator, RAFT-Stereo introduces multi-level convolutional GRUs that maintain hidden states at multiple resolutions simultaneously for better information propagation across different scales.

A recent advancement in 3D reconstruction has been the creation of NeRF: Neural Radiance Fields [47]. This method synthesizes novel views of scenes by optimizing an underlying continuous volumetric scene function using a sparse set of input views. The goal is to map a static scene as a continuous 5D function with dimensions representing 3D location and 2D viewing distance to a volume density and a view-dependent color at a spatial location. The 5D function is approximated by a fully-connected deep neural network without any convolutional layers. This method has been used in [48] to generate new scenes to train other deep stereo models, specifically PSMN-Net, RAFT-Stereo, and CFNet. Here, they collected their own data and for each scene, trained an independent NeRF and defined multiple virtual stereo cameras for each model to render multiple binocular stereo pairs per scene as well as an additional third image to create a rectified stereo triplet. Using these NeRF-generated images, they supervised training with triplet photometric loss using structural similarity in-

dex measure (SSIM) and absolute pixel difference. They also used an additional loss between the predictions and rendered disparities by NeRF. This allowed them to train models with scenes without the need of any ground truth, treating aspects of the result of the NeRF as the ground truth. At the time of publication, they obtained a 30-40% improvement over existing self-supervised methods on the Middlebury stereo benchmark [16, 49, 50, 51, 52].

Iterative Geometry Encoding Volume (IGEV) [53] aimed to improve on the limitations of both cost filtering-based methods and iterative optimization-based methods like RAFT-Stereo. IGEV-Stereo constructs a Combined Geometry Encoding Volume (CGEV) that encodes geometry and context information as well as local matching details. It then uses a ConvGRU-based update operator to iteratively refine the disparity map by indexing the CGEV. Finally, a full-resolution disparity map is obtained by a weighted combination of the predicted disparity at $1/4$ resolution using higher resolution context features to generate these weights. Multi-range Geometry Encodings (MGEV) were introduced in IGEV++ [46] to improve this method. Separate Geometry Encoding Volumes for different disparity ranges, adaptive patch matching, and selective geometry feature fusion (SGFF) are aspects of MGEV that improved the performance of IGEV++.

MoCha-Stereo [45] was built upon the foundations of IGEV and RAFT-Stereo. It was designed to address the issue of geometric structure loss during feature channel generation in learning-based stereo matching, which often results in edge detail errors. MoCha restores lost detailed features in feature channels by using motif channels, repeatedly occurring geometric contours, with normal channels. To determine more accurate edge match costs, the authors introduce Motif Channel Correlation Volume (MCCV). The motif channels are extracted with adaptable sliding windows to mine repetitive patterns in the frequency domain and then are projected onto feature maps. The disparity estimation is then further refined by optimizing for both high and low frequency errors.

Considerable efforts and progress have been made to address the problem of stereo correspondence matching. However, most of these methods assume a fixed parallel stereo camera configuration. They can take advantage of the lack of degrees of freedom and parallel configuration by simplifying the search space and imposing constraints that further refine disparity accuracy. These methods are not suited for a convergent stereo head, as many assumptions

made are incorrect in the non-parallel configuration. And while Langley et al.[34] reported a method for determining vertical and horizontal disparity, they did not evaluate on a convergent stereo system, as well as ignored vertical disparity in parts of the image with singular edges. Overall, very little to no effort has been put into the development and evaluation of an active convergent stereo system.

3.3 Stereo Vision Datasets and Benchmarks

There are many available datasets and benchmarks to evaluate stereo matching. The goal of stereo matching is to match every pixel in the left image, excluding occlusions due to camera motion parallax and the field of view, to the correct unique pixel in the right image. In doing so, one can obtain the depth estimates of these pixels given camera intrinsic and extrinsic parameters, such as focal length, principal point, and camera orientations relative to each other. There has been considerable effort to solve this matching problem and to evaluate the performance of a stereo vision algorithm; standardized benchmarking datasets with well-defined error metrics are necessary. Reliability of these benchmarks depends on the accuracy of their ground-truth disparity. All of the most prominent stereo benchmarks focus on fixed parallel stereo camera configurations and ensure that their epipolar lines are parallel.

Standard benchmarking datasets for parallel stereo vision algorithm evaluation include Middlebury [16, 49, 50, 51, 52], ETH3d [54], SUN-RGBD [55, 56, 57], KITTY [58, 59], and Scene Flow [42]. Many of these stereo benchmarks are predominantly composed of scenes with rich texture and diverse object arrangements, which stereo algorithms often handle well due to the abundance of reliable image information. However, stereo methods commonly struggle with scenarios such as occlusions, repeated patterns, and textureless smooth surfaces. Despite the knowledge of these failure cases, current benchmarks do not include scenes that explicitly evaluate the performance of stereo algorithms under these challenging conditions. Common error metrics include the percentage of bad pixels, average error, RMS error, and percent error quantile. The range of depth of the objects in the scenes of the Middlebury dataset is around 0.6 m- 3 m for object scenes and tabletop scenes, such

as ‘skates’ and ‘backpack’ but can be as far as 20m for larger room scenes such as ‘classroom’ and ‘podium’ [52, 60].

Obtaining reliable ground-truth measurements for stereo datasets is an essential part of any stereo benchmark. The goal is to create pairs of real-world images of complex scenes where each pixel is labeled with its correspondence in the other image. This allows for testing the accuracy of stereo algorithms against known ground-truth correspondences. ETH3d used laserscans for precise depth measurements and proposed an alignment method to align each scan with the image pixels [54]. The SUN-RGBD dataset obtains its ground truth through depth maps obtained by their RGB-D cameras and improving each frame’s depth map by filling in missing values with nearby time-frame maps [55, 56, 57]. Scene flow’s ground-truth is known since the dataset is simulated through Blender.

Middlebury does not struggle with finding the correspondences between its ground truth measurements and the pixels they are associated with, as its ground truth data comes directly from the same cameras and camera positions of the dataset’s parallel images. Their method is closest to the method chosen for collecting the ground truth dataset in this thesis. In [49], the authors use structured light projected onto a scene, a method which has been used in the past for 3D reconstruction [61, 62]. Random light patterns have sometimes been used to provide artificial texture to scenes for better stereo matching [63]. In structured light for 3D reconstruction, typically one projector is used and one or a pair of cameras are used to capture images. For graycode patterns, a series of black and white pixel value patterns and their inverses are projected onto the scene. The captured images under structured light are processed to decode each pixel’s unique label, which is then used to establish correspondences between two stereo image pairs.

For the Middlebury sets in [49] and [52], the authors take many sets of the same scene and camera positions with variations in exposure and projector locations. A variation of exposure was also used in [49] to handle surfaces with varying albedos. To combine disparity estimates, [49] takes an average at pixels with multiple disparity estimates if most disparity values are within a close range. In [52], they use maximum min-stripe-width Gray codes [64, 65], and the results of the left and right cameras individually are combined with the results of using information from both cameras and merged into a single disparity estimate. The

resulting disparity map is then filtered with a bilinear kernel in both x and y directions to fill in and smooth inconsistencies and holes. For each scene, 3 different exposure values are used to collect structured light data with 4-18 projector locations.

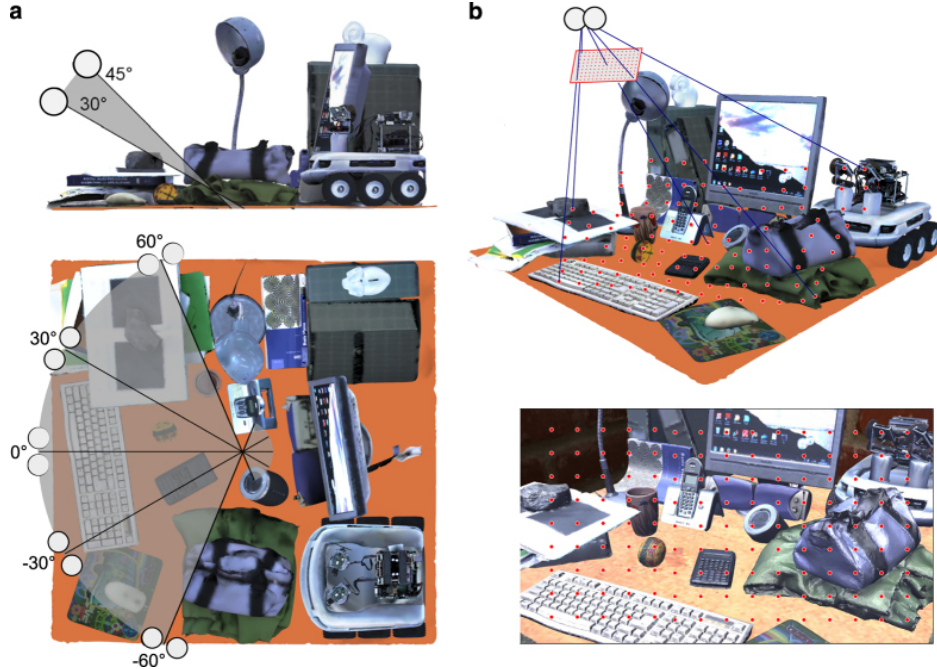


Figure 3.5: A representation of the fixations and head positions in the GENUA PESTO synthetic stereoscopic images dataset [2] with 3D fixations as red dots scattered across the scene. a) Side and top views of the position of 10 vantage points used for the head positions within the 3D scene. The solid thick lines in the images represent the nose direction of each head position. The top image indicates the different elevation and angling of the head b) fixations acquired by the cyclopean camera from 10 vantage points (top), and the fixation points of the camera system (bottom).

GENUA PESTO [2] is the only stereo benchmark that is not catered for parallel stereo. GENUA PESTO is a dataset of stereoscopic images with 6 degrees of freedom for their camera pair. Their dataset is meant to mimic human eye fixations within peripersonal space. The authors develop this set in simulation with 3D virtual models of everyday scenes obtained from a 3D laser scan of a real-world scene and a stereo vision simulator that accounts for vergent eye geometry and cyclotorsion. For each scene, they choose fixations in a 9×15 grid for each head position, Figure 3.5. Each image is 640×480 pixels. Although this dataset is a valuable addition to the research into human vision, its images are taken in simulation and may not fully capture the complexities and nuances of real-world stereoscopic images. Additionally, each scene is constructed of laser-scanned scenes as well as inserted singular

laser-scanned objects. With this method, objects that were scanned with a different lighting source or lighting positioning than the rest of the scene would not appear coherent in the scene and would appear differently if viewed in the real world. Furthermore, their dataset only consists of 2 scenes. This dataset unfortunately does not fully cover the intricacies and varieties of real world textures and scene structures.

3.4 Convergent and Parallel Stereo Comparisons

There has been very little work done to compare the differences between a parallel and convergent camera setup. Geometrically, in parallel stereo, the cameras' optical axes are parallel to each other, and this simplifies the disparity estimation and correspondence problem such that the corresponding points in the left and right images lie on horizontal epipolar lines. It allows for computationally efficient one-dimensional search algorithms. In convergent stereo, the cameras' optical axes converge towards a point, and the correspondence problem is now a 2D search along diagonal epipolar lines. Unfortunately, the analysis on geometrical differences in these camera setups have been limited to pixel quantization error analysis in previous works.

Chang et al. [66] examine the impact of image quantization error on depth estimation in stereo vision systems in both convergent and parallel configurations. They focus on the impact of convergence angle, assuming symmetric fixation, meaning both cameras point inward at the same angle, and its influence on depth estimation accuracy. Establishing a mathematical relationship between depth error, generalized depth resolvability (a function of baseline, focal length, and image resolution), and convergence angle, it was revealed that increasing the convergence angle can reduce expected error. This work was conducted in simulation and in theory and was not verified through a mechanical system with real cameras. Asymmetric convergence, where cameras have different vergence angles, a property of human vision that will be explored in this thesis, was not addressed in this work.

Sahabi and Basu [3] investigate how vergence affects depth estimation error in stereo vision systems in both symmetric and asymmetric fixation scenarios. The authors focus on the discretization error caused by discrete pixels which can be visualized in Figure 3.6.

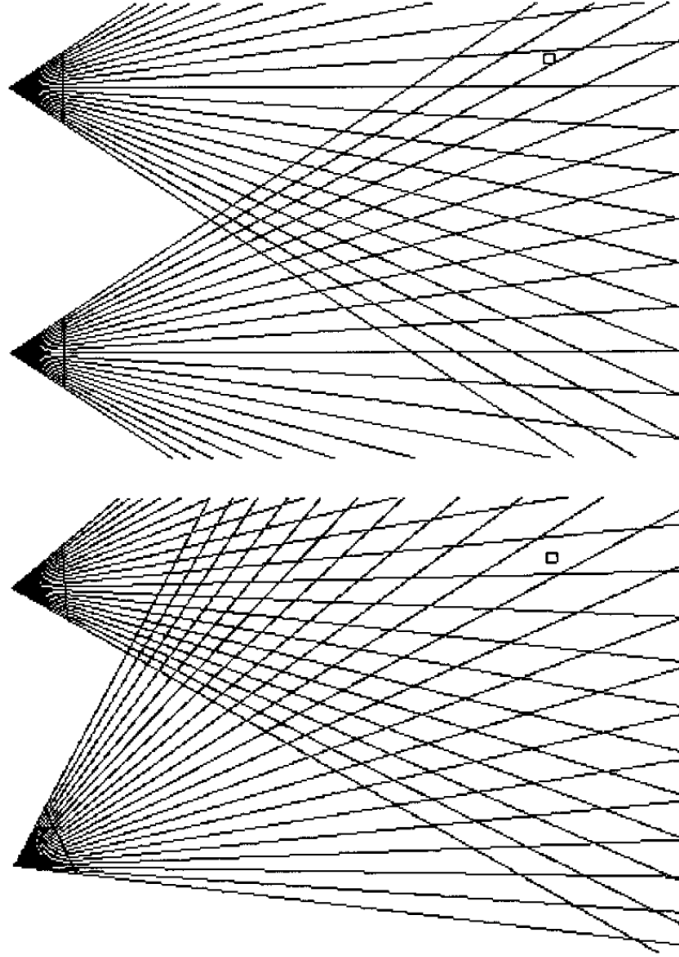


Figure 3.6: Variation of depth uncertainty for stereo cameras with uniform pixel discretization in parallel (top image) and nonsymmetrical vergence angles (bottom image) taken from [3]. The images visualize the differences in the 2D space represented by the cross-section of the corresponding pixels in the left and right images. For a matched corresponding pixel, the volume of space the pixel represents depends on the location the 3D point is in relation to the image planes of both images.

The error increases with distance and is influenced by pixel separation and focal length. The closer the object, the lower the error. This finding is also backed by Blostein and Huang [67], who analyze the error of a parallel stereo setup assuming an ideal pinhole camera model. Sahabi and Basu found that the expected error of a detected 3D point is highest at the center of images due to the pixel quantization effect in a uniform-resolution image. This holds true for both parallel and convergent stereo setups. For a more human-like resolution, they proposed an exponentially decreasing resolution from the center of the image out such that the quantization error of the 3D point would be lowest when the cameras fixate on

the 3D point at the center of each camera’s image. This approach attempts to mimic the sampling strategy of the human eye’s fovea. Their proposal was verified using two cameras with changeable vergence angles. However, the focal length and lens distortion of the images were not considered. The effect of lens distortion typically results in worse image quality at the edges of the image. The quality at the center of the image is more preserved in the context of lens distortion. However, this contrasts with the quantization error being largest at the center of the image, and the two effects have not been studied in conjunction. It is difficult to conclude expected error in practice based on single-source error theories alone, with real systems having many sources of error.

Overall, an exact error analysis and comparison can be difficult when accounting for all sources of error. Other sources of error aside from quantization in a stereo system can come from foreshortening errors, misalignment of camera errors, camera calibration errors, lens quality and distortion errors, and matching errors [68]. While the works mentioned above compare convergent and parallel stereo setups in terms of pixel quantization error, they do not account for other types of error that may arise in practice. Notably, the camera lens distortion effect for many cameras results in the edges of the image having a lower resolution after distortion rectification. This effect, which preserves the center of the image as the area with the least error, has not been studied in conjunction with the quantization effect, which supports that the center of the image has the highest amount of error.

Comparing the resulting depth maps from fixed parallel stereo and a convergent stereo camera system with the ability to fixate on various 3D points in the scene has not been studied. To fully understand the differences in performance between an active convergent camera stereo system (such as those on humanoid robot heads) and a fixed parallel stereo system, one should compare the resulting disparity maps and their projections into 3D world points from each configuration, which encompass all sources of error.

3.5 Humanoid Robot Perception

Many designs of humanoid robots have been created in the past with varying purposes, from search and rescue, general-purpose worker, to psychological studies of cognitive devel-

opment, human perception, and human-robot-interaction [69, 70, 71]. However, very few have implemented active binocular disparity calculations.

3.5.1 Binocular Fixed Stereo Robot Heads

Humanoid robots meant for general tasks and search-and-rescue predominantly have fixed cameras for vision perception. Boston Dynamics Atlas (pre 2024), one of the more famous humanoid robots, utilized a fixed parallel Multisense SL stereo camera [72]. The DARPA Robotics Challenge aimed to conduct humanitarian, disaster relief, and other related operations. Many robots that competed were humanoid. Other bipeds that have competed in the DARPA Robotics Challenge, including THORMANG v3 [73], ESCHER [74], and WALK-MAN [75], are equipped with either a fixed parallel stereo camera system or a single camera along with other sensors for perception.

Other humanoid robots and robot heads using fixed stereo cameras include M-Hubo (a wheeled variation of HUBO from KAIST)[76], ASIMO [77], ARMAR 1 and 2 [78], Valkyrie [79], Robotnaut [80], TWENDY-ONE [81], NAO [82], Reem-B [83], and Phoenix [84], to name a few. These robots were primarily tasked with conducting object manipulation tasks and environment interaction tasks such as active 3D object localization, and were built with this purpose in mind [85]. While these robots model many of the degrees of freedom that human bodies have, they do not include the degrees of freedom of human eyes, which contrasts with the specific goals of this research. For the purposes of studying the value of perception with human-like degrees of freedom, having similar mechanical degrees of freedom as human eyes is important.

3.5.2 Robot Heads with Three Mechanical Degrees of Freedom

Other designs of robot heads, as a standalone head or as a part of a humanoid robot, have included cameras that can converge or pan individually but tilt together as one unit, which results in a total of 3 mechanical degrees of freedom. Figure 3.7 demonstrates these degrees of freedom. Some of these robots include: iCub [86], KOBAIN [87], Armar 3-6 [78] which implements the Karlsruhe head [88], Cog Project [89], Kismet [90], BARTHOC [91],

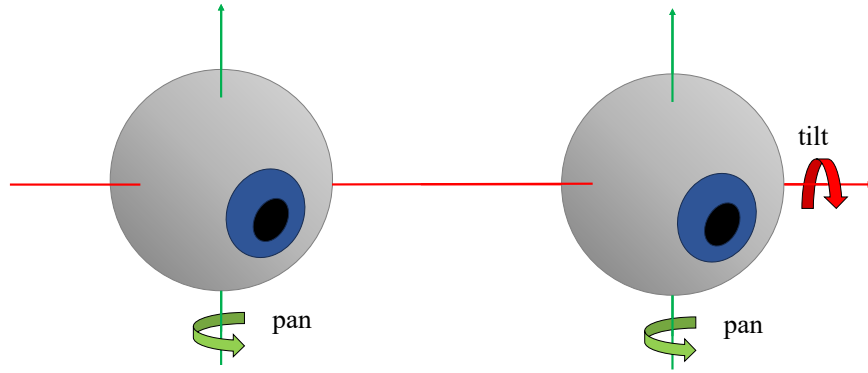


Figure 3.7: An illustration of the three degrees of freedom of the eyes that many robot heads exhibit. Each eye can independently pan but tilts as a unit.

Flobi [92], Balthazar [93], Manzotti et al head [94], and Wang and Jonker head [95].

Some of these robots, including KOBAIN, Kismet, and Flobi, were designed to study human-robot-interaction and hence implemented more degrees of freedom in the cameras to have a persuasive human-like appearance. However, they did not incorporate depth map estimations while utilizing all degrees of freedom.

The Armar 3-6 robots, which use the Karlsruhe head, do not utilize all degrees of freedom to position their cameras and keep the motions of the two eyes parallel to avoid online calibration of the stereo system [96].

Although Cog was designed to study robot cognition based on the hypothesis that human-like intelligence requires learning through experiences and interacting with humans, Cog focused on mechanically mimicking human abilities such as saccade speed instead of developing an accurate or dense method for depth estimation [89].

Notably, the Wang and Jonker head included a set of controllers that mimic neck and eye movements but also included stereo matching and disparity map computations for moving cameras [95]. The images were first rectified to have parallel epipolar lines before using a parallel stereo algorithm, SGBM, to find corresponding pixel matches. Their method, however, performs poorly at high convergence angles, and they did not compare depth maps with ground truth measurements or to a parallel stereo setup.

Pasquale et al. [97] enabled depth perception with the Efficient Large-scale Stereo (ELAS) stereo matching algorithm [39] for iCub’s perception. Here, they estimate the relationship between the cameras of the robot and find the fundamental matrix, which relates the left and right images to each other by defining the epipolar line in the other image given a pixel location in the target image, through robot kinematics. Due to the robot cameras being tendon-driven, the authors note that using purely kinematic calibrations produces unreliable estimates of relative camera orientations, which results in low-quality depth maps. Instead, they use images with detected corresponding features every few frames to correct the estimate of the two cameras’ relationship with the essential matrix. The images were first rectified so that they were parallel, and then the ELAS algorithm was used to calculate the disparity. The disparity map is then filtered with a Gaussian filter, the closest blob is selected, and the centroid and the region of interest for this blob are computed. iCub then directs its gaze toward the closest object using a feedback loop. They report on the tracking abilities from the closest blob selection to compare with a monocular color-based object tracker. The authors do not report on the accuracy of the resulting depth maps with ground truth measurements.

Fanello et al. [40] utilized the previously described iCub depth perception method to more accurately calibrate the hand position of the robot in hand-eye coordination tasks. The effectiveness of this method was evaluated through tool grabbing and placement tasks, and demonstrated 3D scene reconstruction abilities using the estimated depth maps of the working environment. However, this scene was not evaluated against ground truth measurements, nor were the configuration differences between parallel and convergent stereo evaluated.

Manzotti et al. [94] present a method for global disparity estimation using log-polar images for vergence control in a robotic head. They argue that log-polar images, inspired by primate vision, can have advantages over traditional Cartesian images, namely higher resolution in the fovea and a wide field of view. Their technique involves simulating horizontal shifts between left and right images and calculating the disparity that maximizes normalized correlation between images. Unfortunately, they do not report on the accuracy of their method.

Leitner et al. [98] employ a convergent disparity estimation using the Balthazar head [93]

based on a Bayesian framework. First, a finite set of possible disparities is defined with prior probabilities. Then, the likelihood of each pixel in the stereo pair is computed using a generative probabilistic model. Bayes rule is used to compute the posterior probability of each disparity value at each pixel, given the image, and the likelihood of similar disparities at close spatial locations is reinforced to deal with uniform image regions. The disparity of the highest posterior probability is assigned. Although this method produces a disparity map, the computational cost is very high for larger images, so the authors limit the image size to 360×480 . This is due to the inefficient search space method used. To account for both horizontal and vertical disparities, the authors simply define a rectangular region around the pixel of the dimensions (maximum horizontal pixel range $\times$ maximum vertical pixel range). The authors chose 30 pixels in both directions for the horizontal range and 6 pixels in both directions for the vertical range, resulting in a 61×13 rectangular search space. The authors did not evaluate their results with any ground truth nor quantitatively assess the method other than with run times. Additionally, constraining the search space for vertical disparity to be small can be incorrect, as higher convergence angles and pixels farther from the center of the image will result in a higher vertical disparity value.

3.5.3 Robot Heads with Four Mechanical Degrees of Freedom

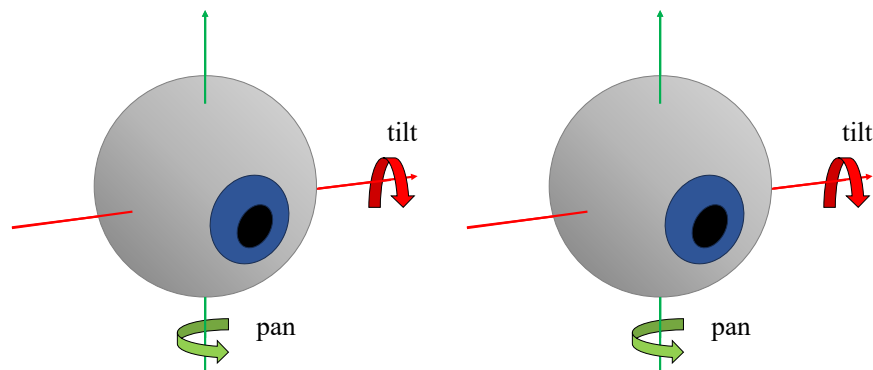


Figure 3.8: An illustration of the four degrees of freedom of the eyes that many robot heads exhibit. Each eye can independently pan and tilt.

Robot heads with 4 total mechanical degrees of freedom are often built to study human-robot-interaction, robot cognition, and modeling human perception in a robot. These robots have cameras that can independently pan and tilt without the ability of cyclotorsion, shown in Figure 3.8. These robots include, but are not limited to: HRP-4C [99], DB [100], CB [101], WABAIN-RII [102], KTH [103], HUBO KHR-3 [104], WE-3 [105], Hadaly2 [106], Romeo [107], Qingbin Wang head [108], Lily [109], FOVEA [110], Xiaolin Zhang head [111], NJU [112], OREO [113], and ESCHeR ETL [114]. While these robots can move their cameras independently for a total of 4 degrees of freedom, many do not utilize these capabilities in practice. The WE-3 robot and the Qingbin Wang head estimate one point of depth at fixation, and Romeo uses pre-calibrated markers to determine the depth and position of its hand while performing tasks. OREO focused on matching the speed of each type of eye movement to human speeds. Xiaolin Zhang’s head focused on eye movements and particularly tracking, but only the horizontal degree of freedom in both cameras was used. Most do not specifically mention the strategies used to compute depth estimations, if any, as the research focuses on human-robot-interaction, providing services, or bipedal motion control.

Rougeaux and Kuniyoshi implemented binocular tracking on the ESCHeR robot [115]. The system uses optical flow and disparity as independent visual cues to track moving targets in complex environments without prior knowledge of object shape or texture. For disparity estimation, the system used a phase-based technique with Fourier analysis for vergence control and target position estimation. The points that lie on the horopter were identified by finding near-zero disparity points in the 400×400 pixel image. Using the phase difference, the target position and its disparity can be determined. This work primarily focuses on ensuring that both cameras are fixated on the same object during binocular tracking.

Klarquist and Bovik propose FOVEA (Foveated Vergent Active Stereo System) [110], which computes local depth maps at multiple resolutions over a series of fixation points and combines local depth maps into a multi-resolution map of the entire scene. Areas with rich image information are prioritized for fixation. This method simplifies the correspondence problem by using a foveated approach with a non-uniform resolution image where the center “foveal” region has a higher resolution than the periphery. This approach confines the impact of vertical disparity to sub-pixel levels as the area of high resolution is in the center of the

image, where vertical disparity due to convergence is not as apparent. Because of this, they ignore vertical disparities. They add an additional degree of freedom to the system by allowing the baseline length to be changed based on the characteristics of the scene being viewed. The maximum resolution used was 480×512 pixels, and the method was tested on surfaces instead of natural scenes with varying objects. They did not report on the accuracy of their method, nor test on ordinary objects and scenes.

Maki et al. implemented a phase-based disparity estimation for tracking on the KTH head [116]. Similar to ESCHeR’s binocular tracking, the paper focuses on a phase-based algorithm but instead uses phase to compute disparity, a stereo method first introduced by Jenkin and Jepson in 1988 [117]. To address the constrained range of the filter’s wavelength, a hierarchical method with pixel-shift using a coarse-to-fine strategy on a Gaussian pyramid of the images was used to estimate disparities incrementally at different scales. The disparity was then used to maintain focus on an object as its distance changes, thereby tracking the desired object. This approach only considers disparities in the horizontal direction. There was no mention of vertical disparity due to convergent images, despite the images not being rectified to be parallel to satisfy the horizontal disparity simplification. The authors report a 0.5-pixel precision on 256×256 pixel images but do not report on the accuracy of the depth maps. They do not report on the effects of vertical disparities due to large convergence angles but mention that hierarchical methods for vertical disparity detection may be a direction for future efforts.

Although more works with 4 degrees of freedom cameras have implemented depth computation, none of these methods compare the resulting depth estimate accuracy and other performance metrics against fixed parallel stereo cameras. Additionally, they do not report on the accuracy of their methods against ground truth measurements when viewing natural scenes. These methods also do not analyze any advantages or disadvantages that having more degrees of freedom than traditional stereo cameras provide.

3.5.4 Robot Heads with Six Mechanical Degrees of Freedom

There are a few robot heads that have been built with 6 degrees of freedom, including: ETL-Humanoid [118], TRISH [119], MAC-EYE [120, 121], and Robot Bionic Eye [122]. The

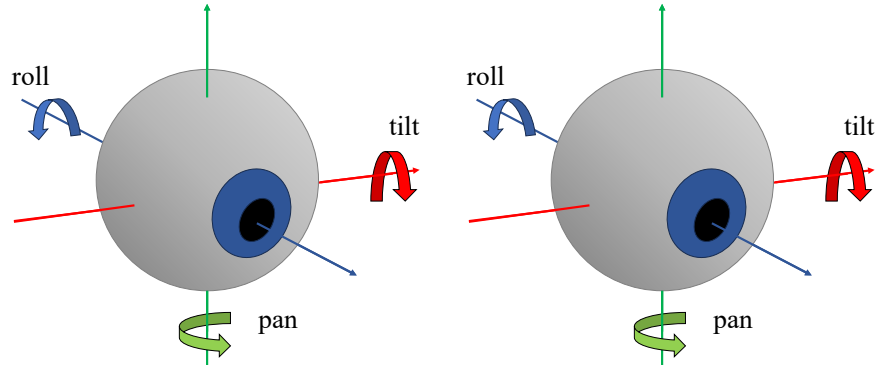


Figure 3.9: An illustration of the six degrees of freedom of the eyes that a few robot heads exhibit

6 degrees of freedom are shown in Figure 3.9 and they include the ability for each camera to pan, tilt, and roll independently.

MAC-EYE primarily focused on implementing the motion abilities of human eyes with a tendon-driven design. Robot Bionic Eye focused on image stabilization. None of these works, however, implemented a stereo disparity algorithm for depth perception. There was no mention from ETL-Humanoid of its perception system.

Chen et al. developed a method for hybrid image stabilization with the Robot Bionic Eye [122]. The robot has a total of 9 degrees of freedom, with 6 for the cameras and 3 for the neck. In [123], they proposed an active exploration algorithm to take advantage of the cameras' mobility and select viewpoints that were better suited for their localization and mapping algorithms. Binocular disparity calculations while using various degrees of freedom were not mentioned.

Jenkin and Tsotsos [124] present techniques for manipulating the horopter by rotating cameras about their optical axes such that the camera orientations were suited for different tasks. They argue that cyclotorsion (controlled camera roll) can be taken advantage of in three ways: mapping structure in the world to the detection regions of the disparity detector operators, making a slanted surface appear fronto-parallel in disparity space which suits binocular processing in stereopsis algorithms, and using explicit target knowledge to replace the bottom-up search process in vision with a top-down search. A small range of disparities

near the horopter was computed using the TRISH head to prove these concepts.

3.6 Summary

The field of stereo vision has predominantly focused on developing algorithms designed for fixed parallel camera setups. While many traditional local and global methods as well as more recent deep learning approaches, have achieved great progress on benchmarks, these methods often take advantage of the simplified geometry and constraints inherent to parallel camera configurations. Consequently, these methods are not suited for convergent stereo systems which have non-parallel epipolar lines, perspective differences, and both positive and negative disparities. In convergent images, the disparity patterns are considerably different from those in parallel images, and the search for stereo correspondence becomes a 2D problem. The previously exploited assumptions of parallel stereo algorithms are incorrect in these configurations.

Benchmarking and standardized datasets are important for any developing field of study. Although GENUA PESTO offers unique configurations with 6 degrees of freedom in their camera pair, they lack a large variety of scenes with only 2 scenes available and they may lack real-world complexities and image features with simulated images. Most stereo matching benchmarks, such as Middlebury [16, 49, 50, 51, 52], ETH3d [54], SUN-RGBD [55, 56, 57], KITTY [58, 59], and Scene Flow [42], focus on fixed parallel camera setups. In general, there is a lack of data available in convergent real-world scenes.

Existing work comparing parallel stereo with convergent stereo vision has primarily examined theoretical aspects like image quantization error and the impact of convergence angle on depth estimation accuracy, often in simulation. There is a largely unexplored area of comprehensive comparisons of real-world parallel stereo vision with an active convergent stereo system that accounts for all error sources like lens distortion, camera misalignment, and resulting depth estimate analysis.

Previous works on humanoid robots and their vision systems reveal gaps in evaluating performance differences between convergent and parallel stereo systems. Robots with fixed cameras or fewer degrees of freedom, such as Atlas and other DARPA robots, focus more

on robustness and task performance rather than discovering differences between human and computer vision. Robots with 3 or 4 degrees of freedom may have implemented depth map calculations or depth point calculations at the point of fixation, but do not evaluate their depth accuracy and scene reconstruction. Although robots such as iCub and the Wang and Jonker head compute a disparity map using convergent images, they rectify images to be parallel while using an off-the-shelf parallel stereo algorithm, ignoring the possible image warping consequences that come with this. While the KTH head implemented a phase-based disparity estimation in their 4 degrees of freedom eyes, they do not report on the accuracy of their methods against ground truth measurements when viewing natural scenes. They also do not analyze any advantages or disadvantages that having more degrees of freedom than traditional stereo cameras provide. Additionally, these previous methods perform stereo matching with low-resolution images and do not evaluate performance in terms of accuracy against ground truth. Finally, previous works have not evaluated their robot against a fixed parallel stereo setup using various vergence configurations. This results in a general absence of detailed comparative analysis on depth estimation performance across different camera configurations compared to the common fixed parallel stereo configuration.

Overall, despite the importance of benchmarking in computer vision, current standardized stereo benchmarks largely only consider fixed parallel cameras with only one dataset of convergent images in simulation. While some theoretical work has compared the possible advantages and disadvantages between convergent and parallel stereo, a real-world analysis of the two systems that accounts for all error sources is lacking. A few systems enable depth perception with convergent eyes in humanoid robots and robot heads, but often the stereo disparity problem is simplified by rectifying images to have parallel epipolar lines, which ignores possible negative side effects of rectification. There is very little effort in understanding performance differences between active convergent stereo systems and traditional fixed parallel stereo systems.

Chapter 4

Methodology

The following sections detail the methods and apparatus used to collect data for experiments as well as the algorithm developed to address convergent stereo depth estimation. These sections provide an overview of the physical robot and its abilities, data collection and ground truth measurements, determining parallel configurations, and the stereo matching algorithm developed. It was important to use a real robot and natural images such that the developed method can be used in real robotic applications. The scenes include a variety of objects and scene geometry meant to evaluate the compared stereo methods on a multitude of situations, including commonly known difficult scenarios for stereo methods, as well as typical scene constructions that would be found in popular benchmarks.

4.1 Robot Head

The research conducted throughout this thesis uses a robotic system, CODY, with two cameras with 6 degrees of freedom, pan, tilt, and roll in both eyes. However, only 4 degrees of freedom was explored, pan and tilt in both eyes. The cameras used are the U3-3881LE AF Rev.1.2 autofocus camera with the Corning® Varioptic® C-S-25H0-026-06 liquid lens. The motors used to control the roll, pitch, and yaw motions of each camera are the MG90S metal gear servos with ball bearings for smooth rotations. The support and frames for the cameras are from 3D-printed material. The cameras have a baseline of 11.5cm. This robot head is a combination of efforts from several people who have contributed to the design

and software, including Bikram Bir Dey, Professor John Tsotsos, Professor Michael Jenkin, Mostafa Kamali, Mingshi Chi, Yuqing Yuan, Zhongzhe Chi, Yiqian Liu, and Markus Solbach. An image of this robot head is shown in Figure 4.1. This work utilizes the top half of the robot, which involves the cameras and motors that move the cameras, excluding the neck. This robot head, along with the work presented in this thesis, is intended to be used on a mobile robotic platform to investigate active observer tasks.

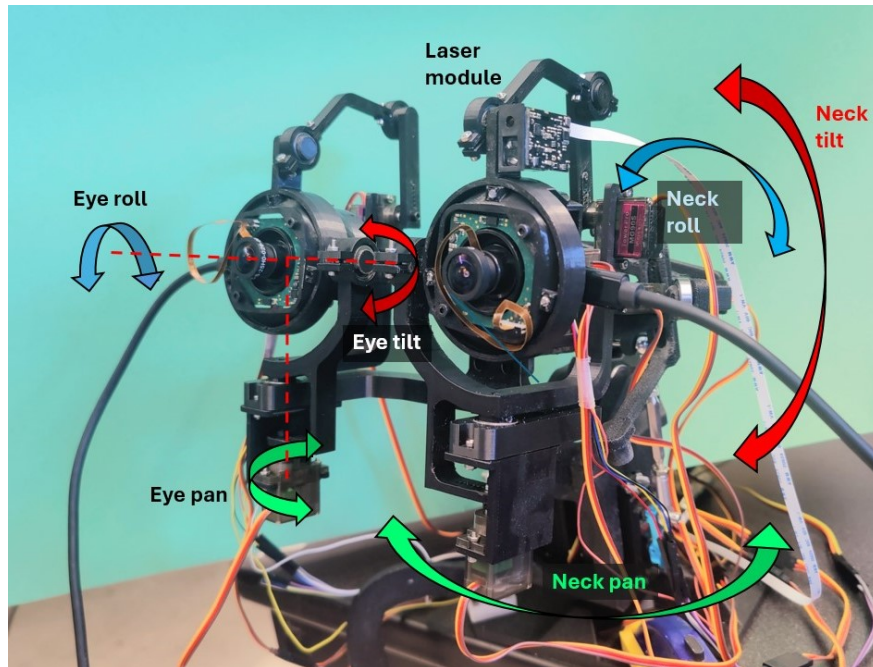


Figure 4.1: Robot head with 3 degrees of freedom in the neck and 3 degrees of freedom for each camera, totaling 9 degrees of freedom.

4.1.1 Camera-Motor Calibration

To ensure that there is a good estimation of camera orientation with respect to motor values, a calibration method for each set of camera motors developed by Kamali et al. [125] was used. This method does not rely on the reported motor angles, as some rotational axes of the robot are not directly aligned with their respective motors' rotational axes. In this case, and in the case of inaccurate motors, a calibration method is needed to ensure trustworthy camera-pose estimations.

The calibration method establishes a correspondence mapping between the motor values

and the camera position. A ChArUco board [126], a derivative of ArUco tags [37], is placed stationary in front of the camera. Then, each of the motors is moved from its minimum to maximum value in steps of 5 or 10 degrees. For each of the steps, the motor values and the image taken from the camera are recorded. Assuming a pinhole camera model, a planar homography matrix relates two images when the captured scene is a planar surface in 3D space and the camera undergoes a rigid body rotation between the images [127]. The homography matrix between two images is estimated using corresponding points detected from the ChArUco board, and in this case, it is the homography for a 3D rotation. It is assumed that there is a pure rotation about the camera center although in practice there may be small translations in the system as well.

A pinhole camera is modeled by $\bar{x} = P\bar{X}$ where $\bar{X}$ is the homogeneous coordinate of a 3D point $X = [X, Y, Z]^T$, $\bar{x} = [u, v, 1]^T$ is the homogeneous coordinate of the image point $x = [u, v]^T$, $P = K[R|t]$ is the camera projection matrix that maps a 3D point to its corresponding image point which involves the camera rotation R and translation t and the camera intrinsic matrix K :

$$K = \begin{bmatrix} f & 0 & p_x \\ 0 & f & p_y \\ 0 & 0 & 1 \end{bmatrix}$$

where f is the focal length, and p_x and p_y are coordinates of the principal point of the camera. The relationship between the 3D point X and its pixel projection x_1 of the first image can be expressed as the following:

$$\bar{x}_1 = K[R|t] \begin{bmatrix} X \\ 1 \end{bmatrix}$$

where K is the intrinsic camera matrix, R is the rotation between the frame of reference X is viewed in and the frame of reference of the image plane, and t is the translation between the frame of reference X is viewed in and the frame of reference of the image plane. The reference frame is assumed to be the first camera frame, and the second camera has a rotation with respect to it with no translation. $[R|t]$ is the 3×4 extrinsic matrix. Since the frame

of reference in this case is the first image's camera frame of reference, there is no relative rotation nor translation. Thus, the relationship becomes:

$$\bar{x}_1 = K[I|0] \begin{bmatrix} X \\ 1 \end{bmatrix} = K(X + 0) = KX \Rightarrow \bar{x}_1 = KX \Rightarrow X = K^{-1}\bar{x}_1$$

I indicates no rotation, and 0 is indicating 0 translation. However, the second image is taken after a rotation is applied to the camera. To describe the same 3D point viewed in the second image, there needs to be a rotation applied to the points to align the frame of reference to the first camera orientation. If the camera is rotated by a rotation R , the following relationship can be observed:

$$\bar{x}_2 = K[R|0] \begin{bmatrix} X \\ 1 \end{bmatrix} = K(RX + 0) = KRX = KRK^{-1}\bar{x}_1$$

$$\bar{x}_2 = KRK^{-1}\bar{x}_1 = H\bar{x}_1$$

$\bar{x}_1$ and $\bar{x}_2$ are the homogeneous coordinates of the projection of X in the first and second image respectively. K is the camera calibration matrix. Here, KRK^{-1} is an invertible 3x3 homography matrix.

This mathematical model of the relationship between two images before and after camera rotation is used for the calibration of motor shaft angle changes to the rotational change in the camera. An image of a stationary ChArUco board is taken at five-degree intervals for each possible motor value. The rotation between two adjacent intervals is calculated using the ChArUco board points and the above mathematical relationship for homography. The motor values and rotations between them are interpolated at each five-degree interval. Bilinear interpolation is used between the closest 4 camera locations. The motor values that have images that do not show enough of the calibration board are excluded from the calibration results. An example of the result of one calibration pass is shown in Figure 4.2. The dots represent the center pixel of the image at each possible motor value.

The result of this calibration is the ability to estimate the rotation between any two sets of motor values and to determine which motor values are needed to obtain a desired rotation,

given the starting motor values from a lookup table.

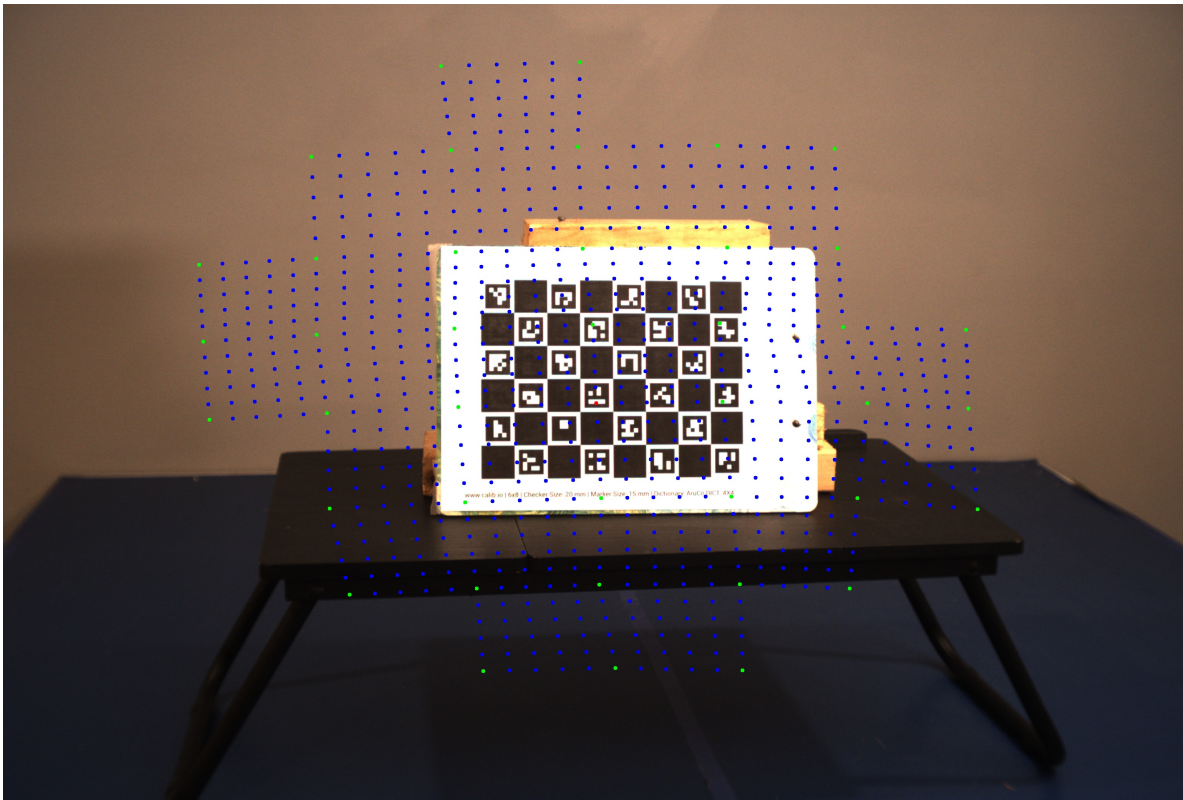


Figure 4.2: Calibration method image centers visualized. The red dot in the center of the image is coincident with the current pair of motor values. The other dots are possible center pixel fixations of the camera and possible motor values. The blue dots are interpolated motor values. The green dots are the 5-degree intervals during calibration.

4.1.2 Camera-Motor Calibration Data

In theory, a single set of calibration images taken in the central field of view of the stereo head would be sufficient. In practice, the wide range of available motions of the cameras in the stereo head complicates this. To see this, consider Figure 4.2 where only placing the calibration board in the center of the field of view does not include motor values for the edges of the image. To address this, many different calibration sets were taken to span the entire camera’s field of view in its parallel position. In total, 12 calibration sets were taken, with 3 repeated middle sets to increase robustness in camera orientation estimation.

With this method, there will be areas of overlap, and a pair of start and end motor values may give multiple rotation estimates depending on the overlapping sets. To account

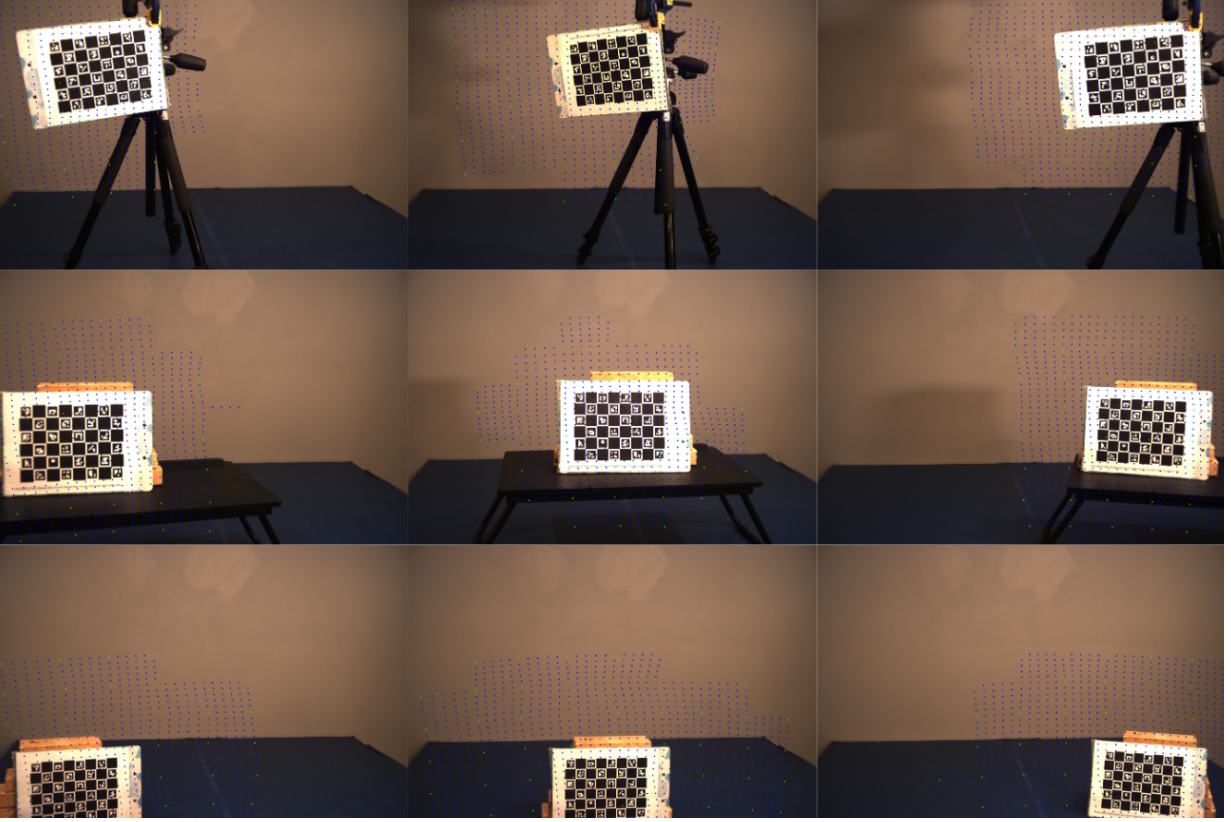


Figure 4.3: Calibration ChArUco boards placed around the field of view to ensure that calibration reaches all possible motor values

for this, the estimated rotations were first filtered by their positional standard deviation. Let $[x_1, y_1, z_1] \dots [x_n, y_n, z_n]$ be the set of all rotation estimates where n is the number of estimates obtained and x, y, z are Euler angles that represent the rotation. Then, $[\bar{x}, \bar{y}, \bar{z}]$ are the mean positional values of the estimates and $[\sigma_x, \sigma_y, \sigma_z]$ are the positional standard deviations. Then, a lower bound and an upper bound range are defined for each value $[\bar{x} \pm \sigma_x, \bar{y} \pm \sigma_y, \bar{z} \pm \sigma_z]$ which is used to determine if a rotation estimate is in or out of distribution. If any positional value of a singular estimate $[x, y, z]$ is beyond the upper or lower bounds, this estimate is discarded. The positional mean of the remaining rotation estimates is used as the final rotation.

4.1.3 Motor Consistency Tests

The calibration of the motors is performed from the minimum value of the motors to the maximum. However, to use this calibration in an active vision scenario where fixations do not

occur in a controlled sequence like in calibration, tests need to be conducted to ensure that the motor values are consistent and that the robot’s camera arrives at the same orientation, regardless of the sequence of motions beforehand.

Unlike stepper motors and DC motors, which operate with an open-loop control system, servo motors, which are used in CODY, utilize a closed-loop control system. The shaft of a servo motor is connected to a positional sensor, here a potentiometer, whose signal is used as feedback to the servo control circuit to continuously correct its position. This means that if there is a constant pulse width, the motor shaft will move to that position, and if anything tries to move the motor from that position, the feedback loop will apply torque to the motor to restore its position. Although small errors may accumulate throughout a very long period of use due to component wear and tear, limitations of the feedback sensor, and control loop imperfections, a servo motor’s positional error generally does not accumulate to noticeable amounts throughout a short period of time, as its closed-loop system corrects its error continuously. A servo motor’s positional error can mostly be considered independent for every new position it arrives at in a short period of time. Recalibrating the motors may be useful after long periods of use.

In an ideal scenario, no matter which orientation (motor values) the camera starts in, if the ending motor values are the same, the camera would result in the same orientation every time. However, in practice, many things can result in rotational errors, such as mechanical errors that propagate through possible slack in the joints in the robot and motor gear backlash. To test the average error of one eye, an ArUco tag was placed in the center of the view of the camera. Then, the pan and tilt motors were iterated through each of their values from minimum to maximum, including the maximum value, in 5-degree intervals going back to the center value with the ArUco tag in the center of its image between each position, and taking an image. If the system is perfectly accurate and precise, for all images, the ArUco tag should be in the center of the image. After the images were collected, the ArUco tag was detected in each image, and the pixel location of the center of the tag was saved. Then, the mean and standard deviation of the absolute horizontal and vertical pixel error were calculated from the image distance between the ArUco tag’s location to the center of the image. From this, the mean of the error in degrees was calculated by determining the

axis-angle needed to align the principal axis of the camera to the 3D ray that goes from the camera center through the pixel of the center of the ArUco tag from using the calibrated camera intrinsic parameters.

To obtain the mean absolute degree error, the L2 norm of each axis angle was calculated and the mean was calculated of these norms. For the left eye, the mean absolute degree error is 0.533 degrees; for the right eye, the mean absolute degree error is 0.614 degrees. To separate this into horizontal and vertical components, the mean pixel error off of the center of the image is taken in the x and y directions of the image plane. Using only the horizontal component, the axis-angle is calculated and the norm is taken to obtain the average absolute degree error in the pan direction. The same is done in the vertical/tilt direction. The results are summarized in Table 4.1. Since the rotation error is fairly small, the calibration of the system can be used to estimate, within a small margin of error, the rotation between any two motor values.

Table 4.1: The mean motor positional error of the left and right eye.

	Left	eye	Right	eye
	Pan	Tilt	Pan	Tilt
Mean	0.532	0.042	0.605	0.106

Motor gear backlash, observed through interacting with the servos, can be a significant source of error. Motor backlash is a clearance or lost motion in the gear mechanisms of the motor caused by gaps between the gears. Its effects are most apparent when a motor changes the direction it is moving. To test the influence of motor backlash in these errors, the calibration process described in Chapter 4.1.1 was performed 3 times for each camera with the same sequence of motions, from the minimum motor value to the maximum motor value, and the same ChArUco board position. This process ensures that, for each calibration set, the motors are rotated in the same direction with minimal changes to the rotational direction. Taking all possible starting and ending sets of motor values, over 1400 pairs of motor values were obtained for the left and right cameras. For each calibration set, the resulting calibration files were used to save the estimated axis-angle rotation between the pairs of motor values in degrees. Then, the absolute degree differences between all 3 of the sets were compared.

The mean degree difference in the left eye with respect to the camera image frame is reported in Table 4.2. The small amount of calibration degree difference indicates that the motors arrive at the same positions with a smaller amount of error, given the same sequence of motion, compared to arriving at the same position from various start locations in the first experiment. Minimizing the effects of motor backlash can reduce camera 3D rotational error. Thus, to reduce errors due to gear backlash, during data collection, the motors are moved from their minimum position to their final position to mirror the sequence of motions during calibration.

Table 4.2: The mean motor rotational difference of the left and right eye between 3 sets of calibration.

	Left Pan	eye Tilt	Right Pan	eye Tilt
Mean	0.201	0.156	0.096	0.113

However, in practice, one does not need to move the eyes in such controlled sequences and can view the scene from fixation to fixation while being aware of the error margin described in Table 4.1. An error in rotation estimates affects farther fixations significantly more than closer fixations. For example, an error of 0.15 degrees at 0.6m in the pan direction in a symmetrical fixation in both cameras will result in an error of 0.016m, whereas the same error at 2.0m would result in a 0.17m error. Knowing the error in the system can influence the confidence of output depth estimates and can trigger movement closer to the target to view it at a more reliable and confident distance. Additionally, through long-term use, the mechanical parts of the robot head CODY may shift and render the calibration ineffective. However, the effects of any mechanical shifts were not observed throughout the collection of 100+ fixations in the dataset by returning motor values to the same values which is expected to place known landmarks in the center of the image, which did not move between each scene. Though if CODY is mounted on an active mobile robot, this may change due to increased vibrations and movements, and it would be recommended to recalibrate before any experiments are performed.

4.2 Image Data Collection

All images collected are in their original form before camera lens distortion correction. Before experiments, they are rectified to remove camera lens distortion effects from a camera intrinsics matrix obtained through camera calibration with 30 images of the ChArUco target calibration board moved to various locations of the camera’s field of view.

4.2.1 Experimental Setup

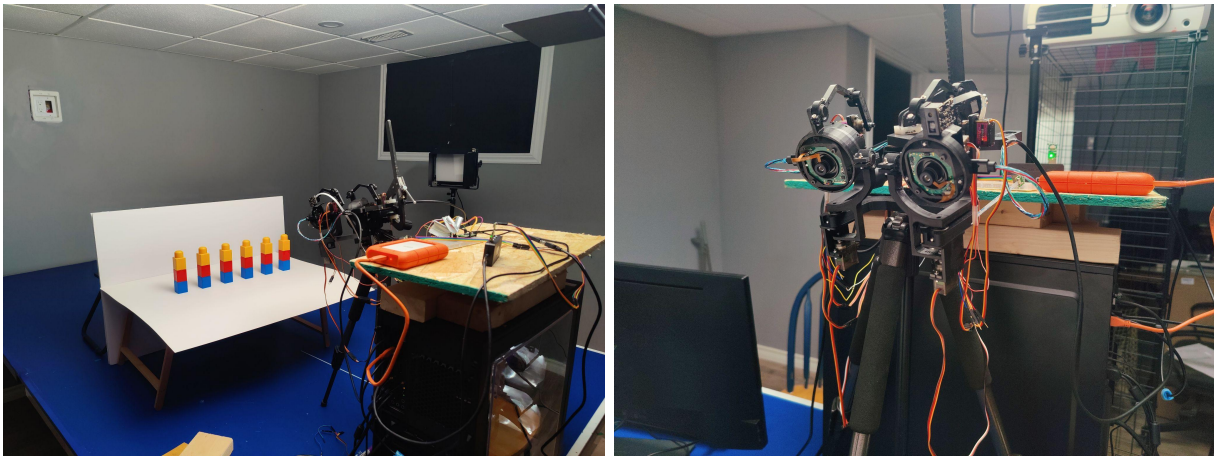


Figure 4.4: Experimental setup for data collection. The room is a contained lighting environment with no external light sources. The robot head is stationary and angled downwards to view the tabletop scenes.

To conduct experiments and collect data, an enclosed lighting environment with no external light sources was used. The closest distance between the robot to the wall is approximately 2.5m.

The light sources used were 2 Neewer CRI 96+ bi-color LED lights. The temperature range is 3200-5600K and includes diffusers to minimize harsh reflections. When possible, the lights were pointed upwards so that the light could be reflected off the ceiling to further diffuse the light. The yellow LEDs were kept at the same or lower brightness level as the white LEDs. When both LED panels were used, their illumination and temperature levels were kept the same.

The cameras’ gain, white balance, and shutter speed were kept constant throughout all

data collection, varying only the 2 LED panel light sources for effective exposure value. Only the cameras' focus varied from scene to scene, to focus on the main part of each scene. The focus was constant for each scene's various fixation images.

The robotic stereo camera pair CODY was mounted on a platform attached to a camera tripod. Only the cameras were moved for each scene, and the mount and tripod were never moved throughout the duration of data collection.

4.2.2 Finding the Parallel Configuration

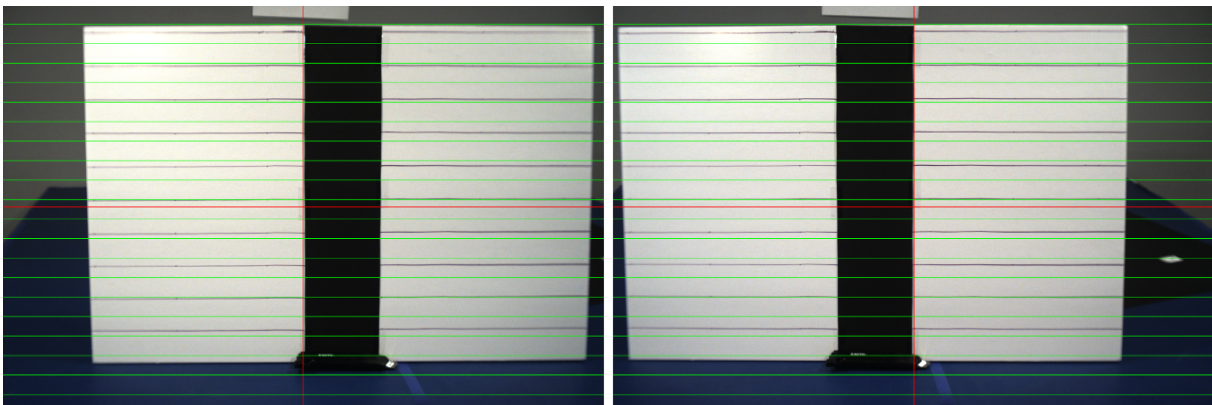


Figure 4.5: Parallel configuration board shown in the left and right cameras respectively. The red lines show the center of the image in the horizontal and vertical directions. The green lines are guidelines for visual adjustment of the placement of cameras.

Before each set of data collected, a process was used to confirm that the expected parallel motor values indeed resulted in parallel images with parallel epipolar lines. The goal is to have the baseline be parallel with the image planes and the horizontal axis of the camera with no relative rotation between camera frames. The only transformation between the two cameras' frame of reference should be a translation in the horizontal direction. To do this, a calibration board, shown in Figure 4.5, with a black strip with an 11.5cm width in the middle, the length of the baseline, was used. The goal is to have the vertical red line, which is drawn in the center of the image, in the left image be on the left edge of the black strip, and the vertical red line in the right image to be on the right edge of the black strip. Once this requirement is satisfied, the camera pan values are close to correct. Then, using the drawn green guidelines in each image, the tilt and roll of each camera were adjusted so that the

scene features would lie on the same line in each image, satisfying the parallel epipolar line constraint. For each motor adjustment, the motors start from the minimum position to their final position to minimize the effects of motor backlash and to create a similar movement to the motor calibration process. Finally, the calibration board was replaced with a ChArUco board to confirm that the rotation between the images is close to the identity matrix.

Let R_{lc} and t_{lc} be the rotation matrix and translation vector of the ChArUco board relative to the left camera. Let R_{rc} and t_{rc} be the rotation matrix and translation vector of the ChArUco board relative to the right camera. These values are obtained from the OpenCV function `estimatePoseCharucoBoard()`. The world points of the ChArUco board P_w can be obtained by detecting the points of the board in each camera's image. Using known information about the ChArUco board's parameters, the exact world locations of the detected ChArUco board points can be extracted. The transformation from the ChArUco board to the left camera is $P_l = R_{lc}P_w + t_{lc}$. The transformation from the ChArUco board to the right camera is $P_r = R_{rc}P_w + t_{rc}$. To find the transformation from the left camera to the right camera, this series of equations is used:

$$P_w = R_{rc}^{-1}(P_r - t_{rc})$$

$$P_l = R_{lc}R_{rc}^{-1}(P_r - t_{rc}) + t_{lc}$$

$$P_l = R_{lc}R_{rc}^{-1}P_r - R_{lc}R_{rc}^{-1}t_{rc} + t_{lc}$$

Therefore, the rotation matrix of the right camera relative to the left camera is $R_{lr} = R_{lc}R_{rc}^{-1}$ and the translation is $t_{lr} = t_{lc} - R_{lr}t_{rc}$. The rotation R_{lr} and the translation t_{lr} are visualized in Figure 3.3 a.

4.2.3 Scene Setup

The set of scenes created for the convergent stereo dataset spans a variety of challenges that are common to stereo matching methods. Several “normal” scenes were included with common objects that provided many features, colors, and textures. Various other scenes were constructed with the intent to test challenging scenarios such as repeated patterns,

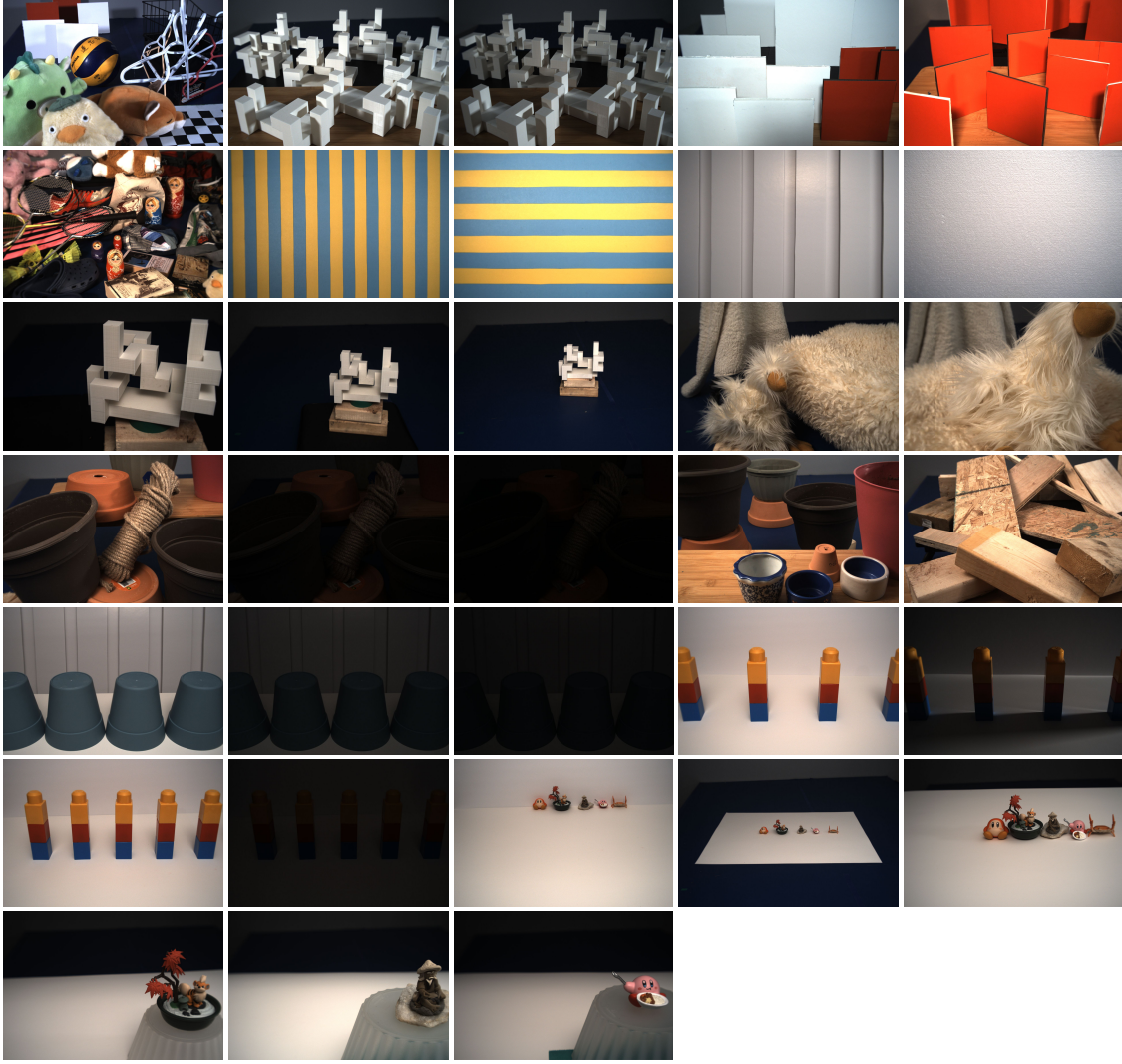


Figure 4.6: The full dataset of 33 scenes showing one sample image from each, a left camera image from the parallel camera configuration.

smooth and textureless planes, high occlusions, dark images, and out-of-domain objects. In total, 33 sets were collected of parallel image pairs shown in Figure 4.6 with 5-10 fixations for each which can be seen in Appendix A.1. This full set has been divided into 7 subsets for further analysis with the subcategory names of “Close”, “Dark”, “Far”, “Smooth Planes”, “Normal”, “Repeated Patterns”, and “TEOS Objects”. The left images of these subsets can be found in Appendix A.

4.2.4 Ground Truth Using Structured Light

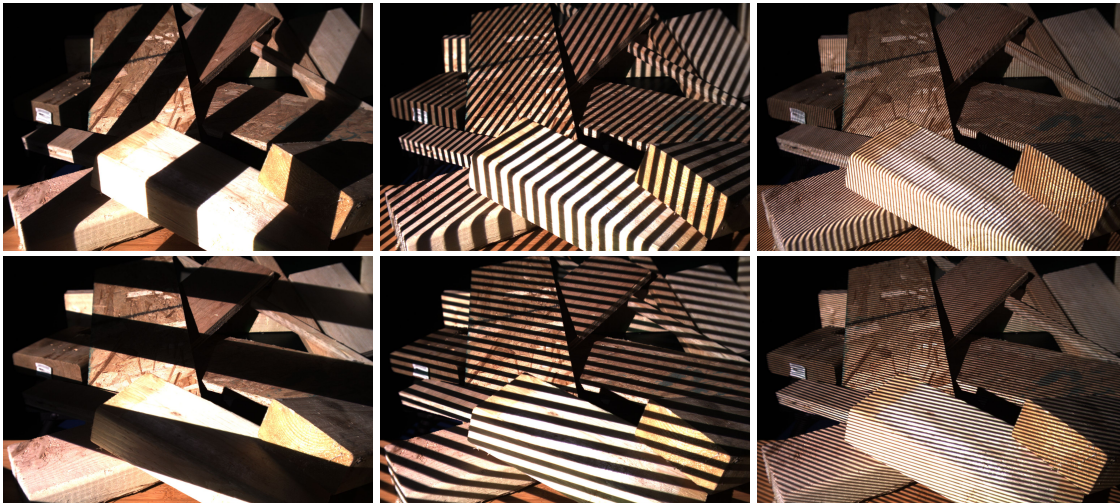


Figure 4.7: Example images of projected Gray-code patterns onto a scene. Gray-codes are patterns of ‘on’ and ‘off’ pixels, which are represented as a black and white striped image. The top row shows projected vertical patterns and the bottom row shows projected horizontal patterns.

The method that is closest to the ground truth collection performed in this thesis is the method used to collect the Middlebury datasets. In [49], the authors use structured light projected onto a scene, a method which has been used reliably in the past for 3D reconstruction [61, 62]. Although random light patterns have sometimes been used to provide artificial texture to scenes for better stereo matching [63], this approach did not provide good results when tested. Structured light is particularly useful in featureless or smooth textured objects as it does not rely on features to determine disparity. Another very important advantage is that they do not need to rely on finding a correspondence between the image pixels and the ground truth depth measurements if the ground truth measurement device is separate from the camera. Aligning measurements can introduce a variety of noise and errors, but with structured light, this alignment process is unnecessary, as the ground truth measurement device is the camera itself. Structured light often has challenges with shiny or transparent surfaces, which are largely avoided in this dataset to ensure proper ground truth data.

After setting up each scene and ensuring that the cameras are parallel, a 1080×1920 resolution projector was used to project 46 total images, which include binary Gray-code

patterns and their inverse onto the scene. Gray-code patterns are patterns of ‘on’ and ‘off’ pixels, which are represented as a black and white striped image. Figure 4.7 shows a few examples of images taken by the left camera of projected Gray-code patterns. Using these Gray-codes, each image is decoded at each pixel to give it a unique label. Gray-code patterns only contain black and white stripes. When using two cameras with a known spatial relationship, the ground truth disparity values of each pixel can be calculated after decoding using functions provided by OpenCV, which follows the work of Herakleous and Poullis [128].



Figure 4.8: Example disparity outputs after decoding. Each output is the result of one set of 46 Gray-code images.

For each scene, 4-12 sets of images of Gray-codes are taken by varying the projector location, and for scenes that spanned a larger depth range, the focus of the cameras as well. This would ensure that projector image occlusions are minimized and that all ranges of objects are captured. Figure 4.8 demonstrates 4 disparity outputs of separate structured light sets from different projector locations (although more were taken, 4 locations were shown to show the variations in disparity maps). Projector light occlusions can be seen near plank edges. It can also be seen that while some projector positions have small occlusions, the other projector positions cover these areas.

Let N be the total number of disparity map outputs. To combine estimates, all N

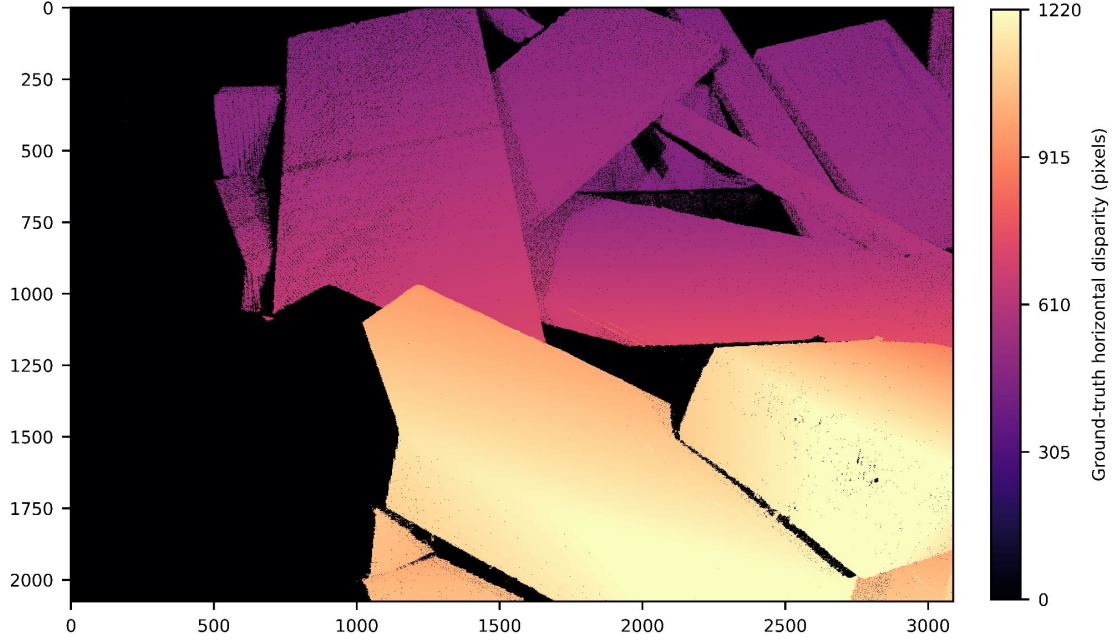


Figure 4.9: Example final ground truth disparity after combining disparity outputs of all decoded structured light sets.

disparity outputs were iterated through per pixel. If there are 2 or fewer disparity maps with values for the pixel, the pixel remains unassigned. Otherwise, the values of the pixel from the disparity maps that have a value assigned to this pixel are collected, resulting in n disparities for this pixel, where $2 \leq n \leq N$. The mean and the standard deviation of the n disparity values are calculated. The disparity values that fall outside of 2 standard deviations of the mean are discarded. Then the mean of the remaining disparity values is assigned as the ground truth value for the pixel. An example of the final output is shown in Figure 4.9. As can be seen in this example, the disparity map is smoother with fewer discontinuities. Unassigned values can be a result of camera field of view and camera motion parallax, where a pixel in one image cannot be seen in the other image.

The structured light images are taken in a parallel camera configuration with both images rectified to account for camera lens distortion. To test the viability of this structured light method, the disparity outputs are transformed to depth measurements through the following equation:

$$Z = \frac{bf}{d}$$

Where Z is the depth of the pixel, b is the baseline of the cameras, f is the focal length of the camera and d is the disparity at the pixel, assuming a parallel stereo camera setup where the focal lengths are the same and there is no difference in yaw between the two cameras.

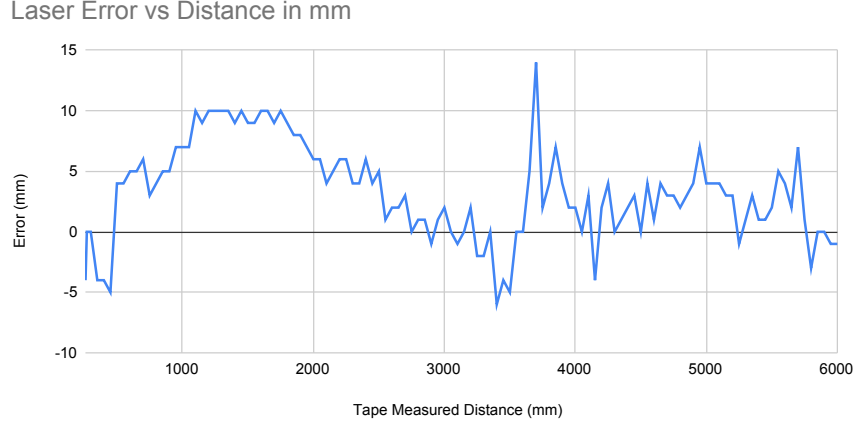


Figure 4.10: Laser measurement errors taken with a LRF20M3PS laser module and a measuring tape. Human errors are expected to be included in these numbers.

These estimates are compared to depth measurements taken with a LRF20M3PS laser module rigidly attached to the top of the left camera frame at 3cm above the camera sensor. The measurement range of the laser is reported to be 0.045m - 20m with a 2mm error range within 10m. The laser was tested separately with a tape measure and a white paper target, varying distances every 5cm between 0.30m to 6.0m. The error in mm is plotted in Figure 4.10, although there may be human error involved in these measurements as well. These measurements serve mainly to check for consistency between real performance and reported performance from the laser’s manufacturer. The precision was also evaluated by taking 5 separate laser measurements for each target location. The location of the target was varied between 0.5m and 6.0m with 0.5m steps. The average difference between the 5 sets of measurements was 0.32mm. These measurements provide confidence in the laser’s accuracy and precision.

The laser is mounted with a slight translation; this was accounted for by transforming frames of reference. The camera’s frame of reference is with the z axis outwards, x axis to the right, and y axis pointing down. Let R_{lc} be the rotation of the laser’s coordinate system around its center to align with the camera’s coordinate system and $t_{lc} = [0, 30, 0]$ be

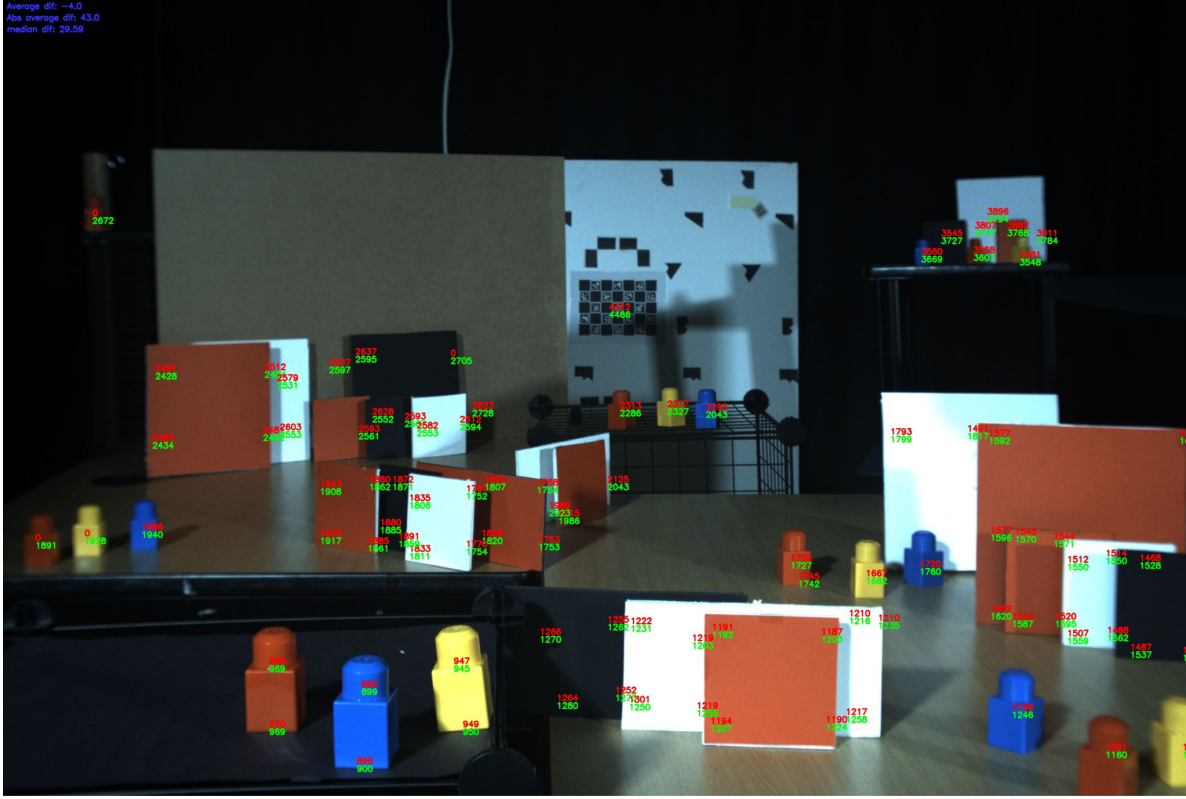


Figure 4.11: Depth difference comparisons between depth projected from disparity values from structured light and depth measurements (mm) taken from the LRF20M3PS laser module. Red depth values are from the structured light ground truth method described in Chapter 4.2.4. Green depth values are the values taken from the laser module.

the translation between the laser center and camera center in the laser’s frame of reference which has the z axis outwards, x axis to the right and y axis pointing down. R_{lc} was estimated by the angle of the mount designed in CAD. The laser’s ray was designed to intersect with the camera’s principal axes at 3m. This was confirmed with a target placed at 3m from the camera centered in the center of the image. The laser point landed at the center of the target. The 3D point that the laser ray hits in the laser’s frame of reference is $r_l = [0, 0, d]$ where d is the depth measure. The ray in the camera’s frame of reference is then $R_{lc}^{-1}(r_l - t_{lc})$.

For the scene shown in Figure 4.11, manual measurements were taken at many corners of the square planes and at least one measurement for each large building block. The red text shows the depth estimate obtained from the disparity map of structured light in mm, and the green text shows the depth estimate obtained from laser measurements. Comparing the depth measurements between the laser and the output of structured light, an absolute

average difference of 43mm and a median difference of 30mm is calculated. This provides strong confidence in the accuracy of the structured light method.

4.2.5 Dataset Fixations



Figure 4.12: Visualizing an example of the ground truth disparity map on the right and the fixations chosen for the scene on the left. Fixations are labeled with red numbers. Black pixels in the ground truth disparity map are unassigned values. These pixels cannot be viewed in both images due to camera-motion parallax.

To have pairs of convergent images that are fixated on fixation points, the cameras of the robot head needed to be moved to fixate on these points. For each scene, a set of 5-12 fixations was chosen depending on the complexity and number of objects in the scene. For scenes with many objects, the fixations were chosen so that they are dispersed throughout the whole image and range of depths. For scenes with fewer than 10 objects, fixations were chosen so that every object is fixated on at a central part of the object. For objects that had a wider range of depth, more fixations were required to capture the whole object. Fixations were also chosen in areas where there is a known ground truth value. Points that cannot be seen in both cameras were not chosen as they are not areas where correspondence matching can be evaluated. In Figure 4.12, the left image shows the set of fixations that covers all the objects in the scene, and the right image shows the collected ground truth of this scene, where the black pixels are areas with no ground truth disparity values. Refer to Appendix A.1 for the visualization of the full dataset and the fixations chosen.

To fixate on each fixation point, the left camera’s motors are first moved from their minimum position to the position of fixation using a red cross drawn in the center of the

image as a guide. This results in the left image having the target point in the center of its image. Once the left eye is fixated, the right camera's motors are moved using the same procedure to have the chosen point in the center of the image. Cyclotorsion was not used in this dataset. The images and motor values at the fixation are saved. The point that was chosen for the fixation was also saved as a pixel value in the parallel left and right images. These values are all available in the dataset.

4.3 Convergent Stereo Matching Algorithm

The following sections detail the methods used to find stereo correspondences. First, an important step of an active convergent stereo method is, for every fixation, determining the fundamental matrix that relates the relationship and orientation between two images. The fundamental matrix allows us to find the epipolar line in the right image for any pixel in the left image. Once the epipolar search line is found for a pixel, this line can be searched to find the best corresponding pixel in the right image. A stereo matching method, named CAS (Convergent Active Stereo), computes the disparity map, and the final disparity is filtered with a Weighted Least Squares filter.

4.3.1 Determining the Fundamental matrix

Calculating the fundamental matrix is an important first step to any stereo matching method. The fundamental matrix relates corresponding homogeneous pixel points $\bar{x}_l$ in the left image and $\bar{x}_r$ in the right image with the relationship $\bar{x}_r^T F \bar{x}_l = 0$

In parallel stereo, cameras are parallel, and the epipolar lines between images lie on the same level as each other. Parallel stereo has motion of the cameras as pure translation with no rotation and no change in internal parameters. In the reference frame of the left camera, the camera matrices are $P_l = K_l[I|0]$ and $P_r = K_r[I|t]$ for the left and right cameras, respectively. Let $K_r = K_l = K$, and let $e' = [1, 0, 0]^T$ represent the vector for the epipole in the right image, then the fundamental matrix always takes on the form:

$$F = [e']_{\times} K R K^{-1} = [e']_{\times} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}$$

With the relationship $\bar{x}_r^T F \bar{x}_l = 0$ and the above fundamental matrix, the relationship reduces to $\bar{x}_{l2} = \bar{x}_{r2}$ with $\bar{x}_{l2}$ and $\bar{x}_{r2}$ being the second element of $\bar{x}_r^T$ and $\bar{x}_l$, indicating that epipolar lines correspond to the same height in the image. The search for a corresponding point in the right image becomes a 1D problem along the same horizontal line as the target point that the algorithm is attempting to match in the left image. This situation is sought out by the process of image rectification, where a transformation is applied to non-parallel images to align corresponding epipolar lines to be parallel. In practice, many out-of-the-box stereo systems use image rectification to better align their images to account for possible mechanical errors.

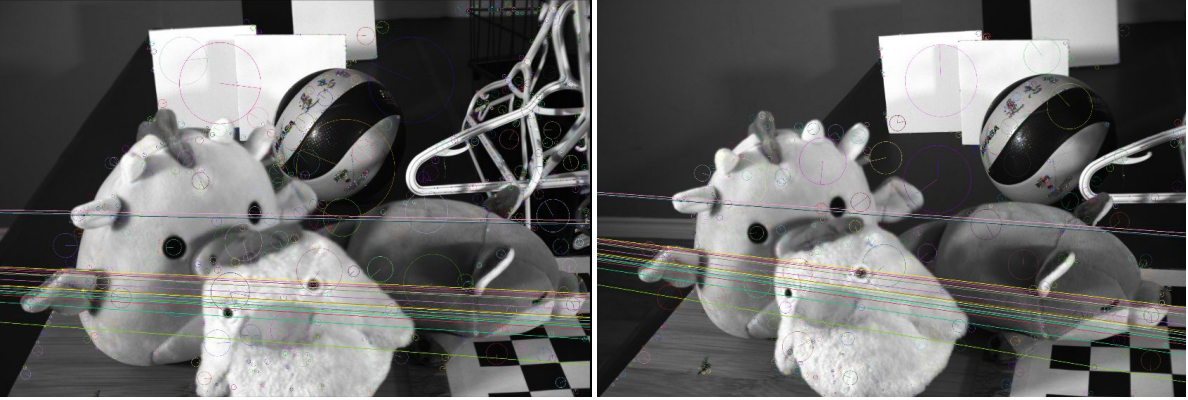


Figure 4.13: Epipolar lines plotted on an example fixation image pair. A random number of epipolar lines are chosen and plotted in both images. Corresponding epipolar lines are shown with the same color. It is expected that any pixel in the left image on a chosen epipolar line can be found on the corresponding epipolar line in the right image, given that there is no occlusion obstructing the view. Detected SIFT features and their orientations are shown with circles and a line indicating the orientation.

However, in a convergent system, the epipoles are not at infinity, and the fundamental matrix is not as simple. Instead, for every fixation of the convergent system, a new fundamental matrix needs to be estimated to obtain the correct epipolar lines to search for pixel correspondences. Figure 4.13 shows an example of a pair of images when fixated on the same point in space. Despite not being parallel, the corresponding epipolar lines are plotted in

the same colors, going through corresponding points and satisfying the epipolar constraint. Obtaining an accurate fundamental matrix estimation is important in being able to search in the correct locations for the corresponding pixel. Additionally, the camera projection matrices, which include information about camera geometry and orientations, are critical in projecting disparity information into depth information.

One common way to find the fundamental matrix of the image pair is the 8-point algorithm from a set of corresponding image points. It was first introduced by Longuet-Higgins in 1981 [129], but in practice, the normalized 8-point algorithm described by Hartley in 1997 [130] gives better results. The algorithm estimates these matrices from a set of 8 or more corresponding image points. To obtain these corresponding points, feature extraction and matching are used. The algorithm is sensitive to outlier points, and often a point filtering strategy such as RANSAC is used before feeding the remaining points to the algorithm. However, the algorithm may produce an unsuitable fundamental matrix from rank deficiencies if the matched points do not cover the entirety of the scene or if the points come from the same plane. For example, if the points are sampled from one plane, the correspondences will satisfy a homography rather than a general fundamental matrix and will result in the algorithm being rank deficient with multiple possible solutions for F . Scenes with very few features or with an uneven distribution of features may pose an issue with this method.

Another way to estimate the fundamental matrix is when the right camera's rotation and translation relative to the left camera is known. The essential matrix relates back projected rays of corresponding pixels in the left and right image with $\bar{y}_r^T E \bar{y}_l = 0$ where $\bar{y}_l$ and $\bar{y}_r$ are back projected rays of corresponding pixels in the left and right images, respectively, and E is the essential matrix. One can use the known translation t and rotation R between the cameras to determine the essential matrix as it is defined as $E = [t]_{\times} R$. Here, the transformation $[R^T | t]$ will transform points from the right image to the left image. From the essential matrix, the fundamental matrix can be calculated if the camera intrinsic matrices are known, with the relationships $E = K_r^T F K_l$ and $F = K_r^{-T} E K_l^{-1}$.

To address the issue of possible rank deficiencies and sparsely detected features, the two methods of estimating the fundamental matrix are combined. The following describes the combined method used to find the fundamental matrix for each unique fixation.

First, the expected fundamental matrix is calculated using only the robot head's kinematic calibration to determine the camera geometry. From the calibration method mentioned in Chapter 4.1.2, for each camera, the calibrated rotation between the parallel motor values, R_l and R_r , and the fixation motor values is known. The expected translation needed to be performed on the left camera to obtain the right camera's pose in the parallel configuration, assuming rotation about the optical axis is $t_{lr-p} = [0.115, 0, 0]^T$ from the design of the robot. The fixated translation vector is then $t_{lr-m} = R_l^T t_{lr-p}$. The subscript m is used to indicate that this rotation matrix is derived from motor values. The rotation of the left camera to the right camera's orientation is then $R_{lr-m} = R_r R_l^T$. The essential matrix can be constructed with $E = [t_{lr}]_{\times} R_{lr}$. The fundamental matrix can then be derived by using each camera's calibration matrix and the relationship $E = K_r^T F K_l$. This fundamental matrix F_m from motor calibrations is expected to be close to the true fundamental matrix.

Next, SIFT features [131] are detected in both images and match detected features using brute force KNN matching using the L2 norm and Lowe's ratio test described in [131]. SIFT features were chosen because they are highly distinctive and are likely to be unique to a specific feature point, and can result in more accurate matching in scenes with many similar-looking features. SIFT is often robust to scale changes, rotation, illumination changes, and viewpoint changes to a degree, but relies on texture information. It is well established and widely used in research and has often been proven to have good performance in complex scenarios. The robustness to scale changes, rotation, and viewpoint changes is critical in a convergent image pair. Due to the convergence, the same object may occupy a different number of pixels between the left and right images. Additionally, perspective differences and rotations are bound to occur due to the convergent nature of the images. Although other alternatives have been created to address some disadvantages of SIFT such as SURF [132], ORB [133], BRISK [134] and BRIEF [135], SIFT and SIFT-based features are the most accurate out of traditional feature descriptors for scale, rotation, and affine variations [15]. However, if real-time speed is necessary, another feature descriptor may be considered to increase efficiency.

The SIFT feature matches are then filtered with the epipolar constraint using the fundamental matrix obtained from motor values. If the matched right pixel is 100 pixels or

farther away from the expected epipolar line of the left pixel, then the matched pair is discarded. The remaining pairs are then used with RANSAC as an additional outlier detection method to estimate the fundamental matrix using the 8-point algorithm F_{8pt} . Once this fundamental matrix is estimated, the R_{lr-8pt} matrix can be extracted using the essential matrix (calculated with the fundamental matrix and camera intrinsic matrices), the matched feature points, and the left camera's intrinsic matrix K_l . The candidate rotation matrices and translation vector are refined to one rotation matrix and translation vector R_{lr-8pt} and t_{lr-8pt} using the Cheirality condition [136], which checks that the triangulated 3D points should have positive depth, and matched points.

The two rotation matrices R_{lr-8pt} and R_{lr-m} are compared, and the difference in angle between them is obtained by calculating the angle of rotation between the two rotation matrices. Let $R_{diff} = R_{lr-m}R_{lr-8pt}^T$ be the rotation between the two rotations then $\theta_{diff} = \arccos(\frac{\text{trace}(R_{diff})-1}{2})$ is the shortest angle of rotation. If the angle between them is higher than 5 degrees, then F_{8pt} is discarded and F_m is used for all future purposes. If the angle is less than 5 degrees, then they are sufficiently close to each other, and F_{8pt} is used for all future purposes.

4.3.2 Stereo matching algorithm

In this section, the algorithm and methods used to obtain vertical and horizontal disparities with a pair of convergent images are detailed. A flow of the algorithm is shown in Figure 4.14. The general steps are as follows:

1. Choose a 3D point that lies on a surface or object.
2. Fixate on the 3D point in the scene. A fixation is achieved when both cameras show the same point in the center of their images. This is determined visually.
3. For the fixated left and right images, construct a 5-level Gaussian pyramid
4. Taking a pair of images from the coarsest level of the Gaussian pyramid, level 4, do:
 - (a) Determine the fundamental matrix for this fixation pair using the method described in Chapter 4.3.1.
 - (b) Detect and match SIFT features that satisfy the epipolar constraint.

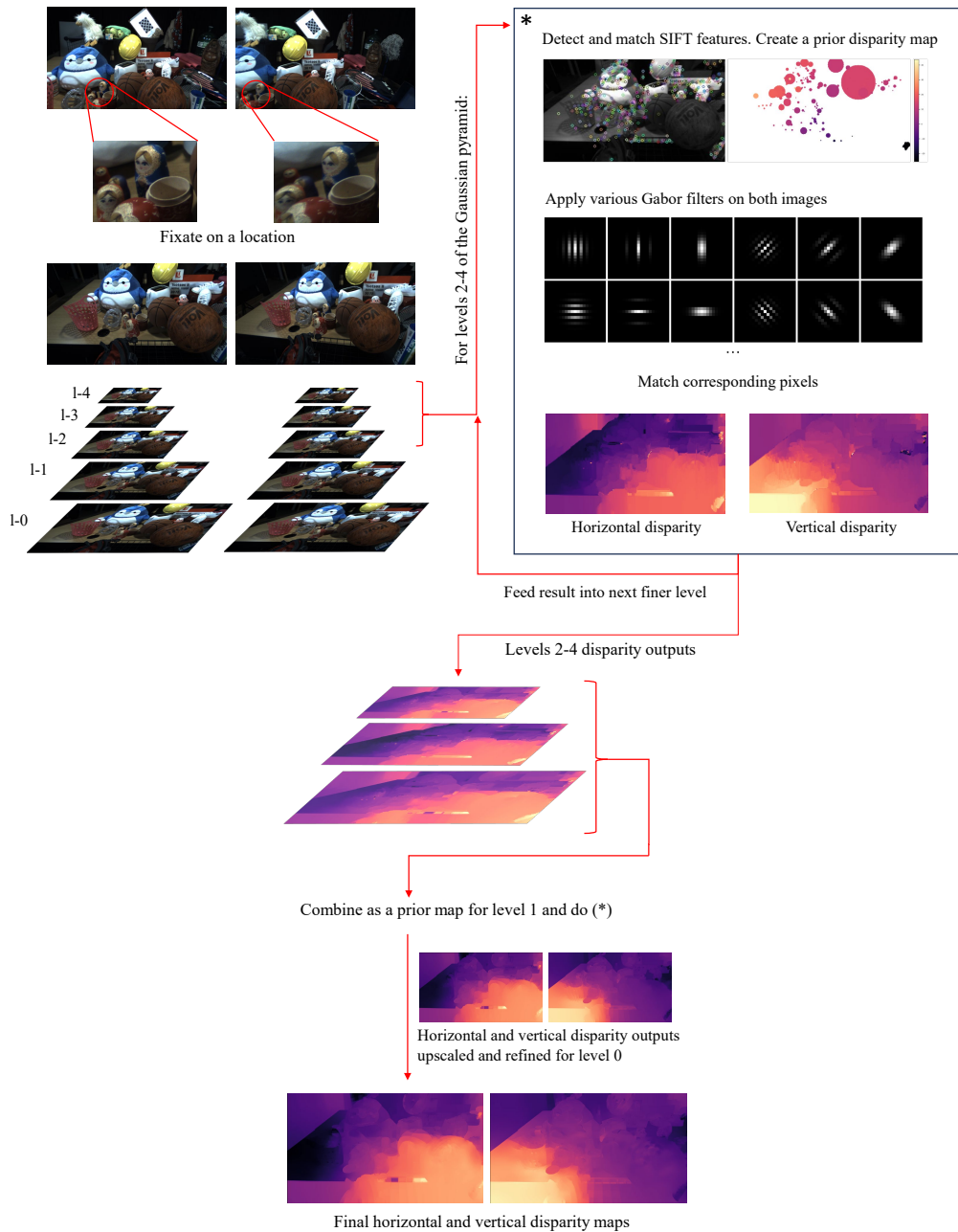


Figure 4.14: Diagram of the convergent stereo matching algorithm, CAS (Convergent Active Stereo). First, a point in the scene is fixated by both cameras and the corresponding left and right images are obtained. Then, a Gaussian pyramid is applied to both the left and the right fixated images. To match corresponding points, first determine the fundamental matrix of the fixated images and then detect and match SIFT features that satisfy the epipolar constraint and use these features as a prior for matching. Then, apply a set of predefined Gabor filters to the left and right images and find the best match using the vector of Gabor filter responses as a feature vector for comparison.

- (c) Create a prior disparity map based on the matched features' size and location
 - (d) Apply a set of predefined Gabor filters to the left and right images
 - (e) Match corresponding pixels using the method described in Chapter 4.3.2.1.
5. Repeat number 4 and collect disparity maps for levels 3 and 2.
 6. Combine disparity estimates from levels 2-4 and use this disparity map estimate as a prior map for level 1. Do number 4.
 7. Using level 1's disparity maps, upscale and refine disparities for level 0.

Given a pair of fixated images, a Gaussian pyramid of 5 layers is created for each image. The method for creating the Gaussian pyramid is described in [137] and is implemented in OpenCV. A Gaussian pyramid can be useful in stereo vision algorithms by enabling a coarse-to-fine matching strategy. Disparity estimates obtained at a coarser level provide a good estimate for the finer levels and can reduce the search space and be computationally more efficient. It allows the algorithm to handle larger disparities that would be computationally expensive and prone to noise at the original image's resolution. Say level 0 is the original image, then each level afterwards, from 1 to L is at a lower resolution than the previous. For each level of the pyramid, starting from the lowest resolution to the 2nd finest level, level 1, a disparity map is computed using the algorithm described in Chapter 4.3.2.1. For level 2 and onward, the previous level's disparity map is used as a starting location for the search space of the disparity computation algorithm.

Each level's disparity maps are saved, and their information is used as weighted priors for the search space of the 2nd finest level. The weights of the disparity maps correspond to their coarseness; the disparity maps obtained from finer levels have more weight than those obtained from coarser levels. For example, if the coarsest level is 4, then the disparity map at this level will have a weight of 1, level 3 will have a weight of 2, and so on. The equation for this is:

$$D_{1-\text{prior}}(x, y) = \sum_{l=2}^L \frac{(L+1) - l}{\sum_{n=1}^{L-1} n} * D_l(x, y) \quad (3)$$

$D_{1-\text{prior}}(x, y)$ is the disparity map prior that will be used in the 2nd finest level to search for correspondences. It is a function of both horizontal and vertical disparity (x, y) . L is the

total amount of levels, and $D_l(x, y)$ is the disparity map obtained at coarser levels. Then, for the image pairs that are of the original image size $D_0(x, y)$, the disparities are refined by searching through a small range of surrounding pixels.

4.3.2.1 Disparity map computation

To compute a disparity map between a pair of convergent images, a prior map of estimates of disparities is first created. SIFT features are first extracted in both images and matched using the epipolar constraint as a filtering method. The locations of the matches in the left and right images are turned into horizontal and vertical disparity values. A map that is the same size as the left and right images is created to save these disparity values. For each SIFT matched pair, a circular area with the same radius as the size of the matched SIFT feature is assigned the disparity values associated with this match. Pixels with multiple disparity estimates assigned are averaged to maintain one estimate. Additionally, assuming the system is convergent on the same point in space, the middle of the image would have 0 disparity, and the middle of the map is assigned to have 0 disparity. This map of vertical and horizontal disparities is passed into the matching algorithm as a prior to constrain the search space. An example of detected SIFT points plotted in the left image along with the resulting prior map is shown in Figure 4.15, where only horizontal disparities are visualized.

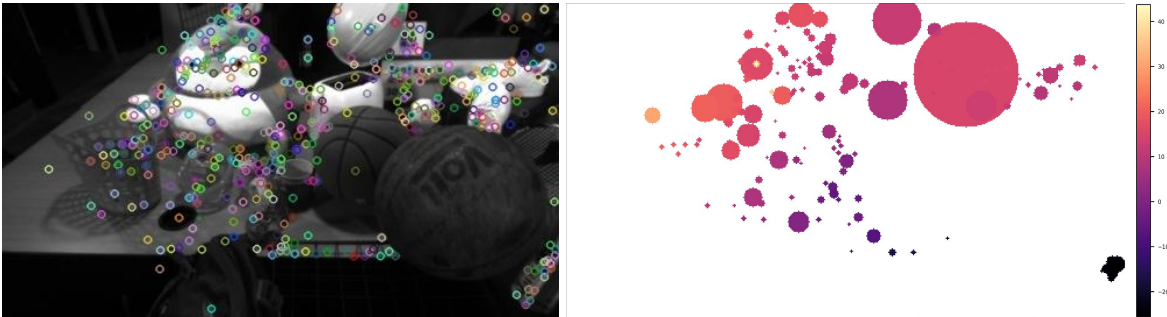


Figure 4.15: Detected SIFT keypoints (left) on a coarser level of the Gaussian image pyramid and its corresponding prior horizontal disparity map (right) for horizontal disparity after filtering matched points with the fundamental matrix. The prior map is composed of matched SIFT points between the left and right image. The horizontal and vertical disparity values of each SIFT prior circle is assigned based on the displacement between matches on the image plane. The size of the SIFT prior circle is determined based on the size of the SIFT feature. Although only horizontal disparity is visualized, a vertical disparity prior is also computed simultaneously.

Then, Gabor filters are used to extract texture and edge information from both images. Corresponding pixels of the image pair's Gabor filter responses are matched while considering both horizontal and vertical disparities. The choice of using Gabor filters as a first preprocessing step comes from its similarities to V1 cells in the human visual cortex [138].

A 2D Gabor filter in the spatial domain is a Gaussian kernel function modulated by a sinusoidal plane wave. Gabor filters are commonly used in image processing for feature extraction and texture analysis. A Gabor filter's impulse response is the convolution of the Fourier transform of the sinusoidal function and the Fourier transform of the Gaussian function. It has a real and an imaginary component and can be formed into a complex number or used individually. The real form is the following:

$$g(x, y; \lambda, \theta, \psi, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \cos\left(2\pi \frac{x'}{\lambda} + \psi\right)$$

Where (x, y) are the spatial coordinates, $x' = x\cos\theta + y\sin\theta$ and $y' = -x\sin\theta + y\cos\theta$. λ is the wavelength of the sinusoidal factor and determines the frequency of the features the filter will be sensitive to. θ is the orientation of the normal to the parallel stripes of the Gabor function and determines the direction of the edges and textures the filter will detect. ψ is the phase offset, which affects the asymmetry of the filter. σ is the standard deviation of the Gaussian envelope, and it determines the effective spatial extent of the Gabor kernel. γ is the spatial aspect ratio, which affects the ellipticity of the Gaussian envelope. When $\gamma = 1$, the envelope is circular.

For each level of the Gaussian pyramid stereo pair, a set of Gabor filters is applied to each image with different parameter variations. Having different parameters for each filter allows us to capture a diverse set of features at different orientations and scales. A single Gabor filter can only detect features of a specific orientation and frequency. Real-world images, however, may contain a large variety of textures and edges at different scales. The use of multiple filters guarantees that at least one will have a sufficient response to give an accurate disparity estimate. Different θ values make the system sensitive to edges and lines oriented in that specific direction. Varying λ allows the filters to respond to different scales or frequencies of textures. Shorter wavelengths detect finer details while longer wavelengths capture coarser

structures. 4 different orientations of θ are used from 0 to π $[0, \pi/4, \pi/2, 3\pi/4]$, 3 values of λ were used $[2, 4, 8]$, 2 values of γ were used $[0.5, 1.0]$, and 2 values of σ were used $[1, 2]$. ψ was kept at 0. Each unique combination of these parameters is shown in Figure 4.16. The filters were applied to both the left and right images, and the outputs are saved in a stack of filtered images.

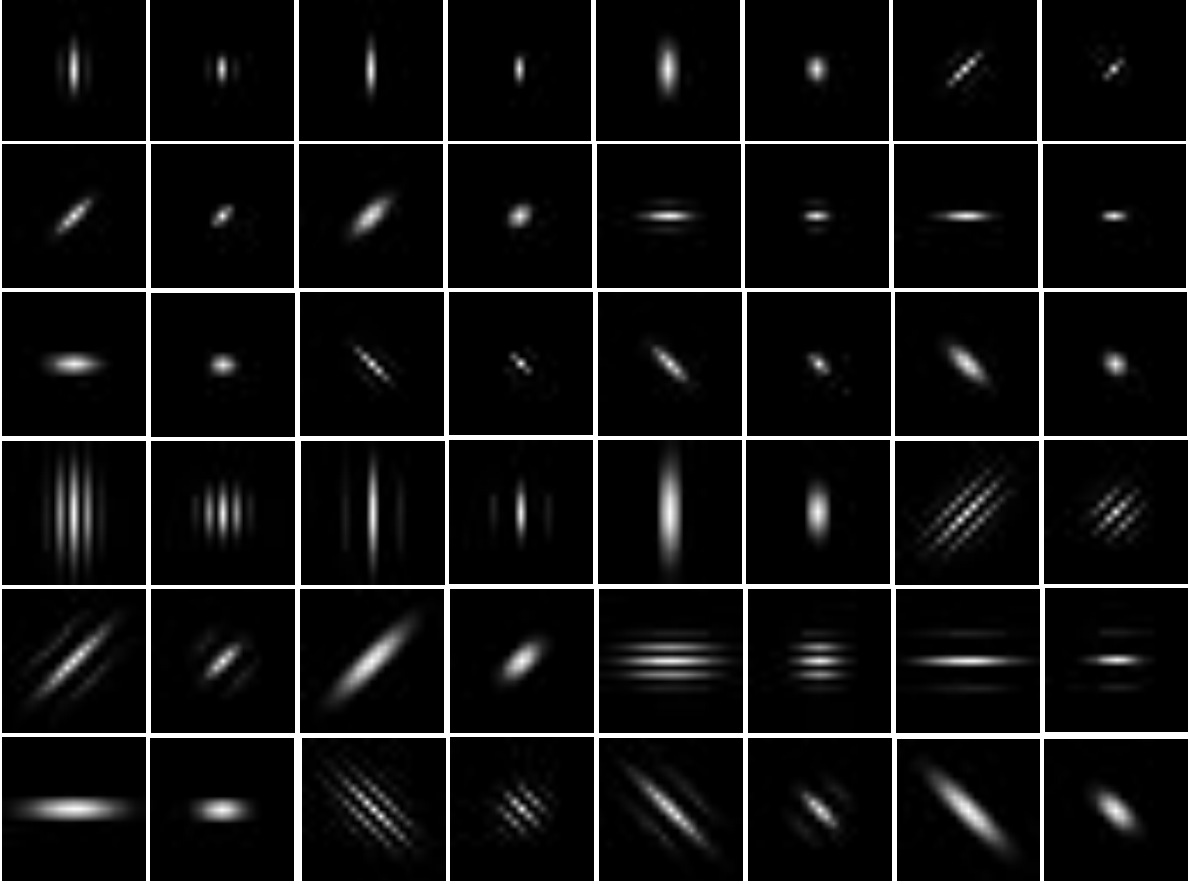


Figure 4.16: The set of different empirically determined Gabor filters used on the left and right images before stereo matching. 4 different orientations of θ are used from 0 to 2π , 3 values of λ were used, 2 values of γ were used, and 2 values of σ were used. ψ was kept at 0.

For each homogeneous pixel $\bar{p}_l = [x_l, y_l, 1]^T$ in the left image, the algorithm needs to find a corresponding homogeneous pixel in the right image $\bar{p}_r = [x_r, y_r, 1]^T$ within a defined distance from the left pixel. In the case of the left pixel having a prior disparity value from the matched SIFT points, the start of the search range begins at the estimated corresponding pixel in the right image. Suppose there is a prior map with an initial estimate of the disparity for the particular pixel. In that case, the search space for this pixel will begin

at the prior disparity and expand until a smaller threshold radius from there, reducing the search space considerably. The range varies depending on the current Gaussian pyramid level and is smaller if the left pixel has a value from the prior SIFT feature matching map. The line to search for a correspondence is found through the fundamental matrix property $l_r = F\bar{p}_l = [a, b, c]$. Since the lower levels of the Gaussian pyramid are at a reduced resolution, the line needs to be scaled appropriately as well as the pixel with $\bar{p}_{\text{scaled}} = \text{factor} * \bar{p}_l$, the line obtained with this pixel location is the line in the original scale. Thus, the line needs to be shifted such that it is in the current scale by dividing the last element of the line by the scale factor $l_r = [a, b, c/\text{factor}]$. There is a radius surrounding the line where those pixels are also considered in the correspondence search. This creates a set of N valid search pixel locations in the right image $S = \{[u_1, v_1], [u_2, v_2], \dots, [u_N, v_N]\}$ and correspondingly, a set of possible disparity values $D = \{[dx_1, dy_1], [dx_2, dy_2], \dots, [dx_N, dy_N]\}$ representing horizontal and vertical disparity obtained by the shift in the image plane if S_i $i \in 1, 2, \dots, N$ should be the corresponding pixel.

Some stereo matching approaches at this step, given the filtered versions of the left and right input signals, would match specific features such as zero-crossings or compute a correlation between patches [139]. The proposed algorithm focuses on the filtered signal amplitude, which represents the local strength of the filtered signal at a particular spatial location and scale. It indicates how much the filter has responded to the local frequency and orientation of the input image. For each stereo image pair, there are K Gabor filtered images $G_l = \{G_{l1}, G_{l2}, \dots, G_{lK}\}$ and $G_r = \{G_{r1}, G_{r2}, \dots, G_{rK}\}$ for the left and right images respectively. For each pixel location in the left image $p_l = [x, y]$, pixels of that same location from the Gabor filter responses $G_l(p_l) = \{G_{l1}(p_l), G_{l2}(p_l), \dots, G_{lK}(p_l)\}$ can be extracted creating a vector of Gabor filter responses as the feature descriptor for each pixel. Then, this pixel is compared against every candidate right image pixel $S_i = [u_i, v_i]$ and its corresponding Gabor filter responses $G_r(S_i) = \{G_{r1}(S_i), G_{r2}(S_i), \dots, G_{rK}(S_i)\}$ using a sum of squared differences (SSD) metric:

$$SSD_i = \sum_{k=1}^K (G_l^k(p_l) - G_r^k(S_i))^2$$

Where SSD_i is the SSD value at the i^{th} search location $i \in 1, 2, \dots, N$, $G_l^k(p_l)$ is the k^{th} element of the vector of Gabor filter responses of the left image at pixel p_l , $G_r^k(S_i)$ is the k^{th} element of the vector of Gabor filter responses of the right image at the i^{th} search pixel $S_i = [u_i, v_i]$. Then, the index $\hat{i}$ is found which corresponds to the smallest SSD_i value, and the disparity value $\hat{d}$ is selected according to the index. Then, the estimated disparity map is refined with a Weighted Least Squares filter using the original left image.

$$\hat{i} = \arg \min_{i \in 1, 2, \dots, N} (SSD_1, SSD_2, \dots, SSD_N)$$

$$\hat{d} = d_{\hat{i}} = (dx_{\hat{i}}, dy_{\hat{i}})$$

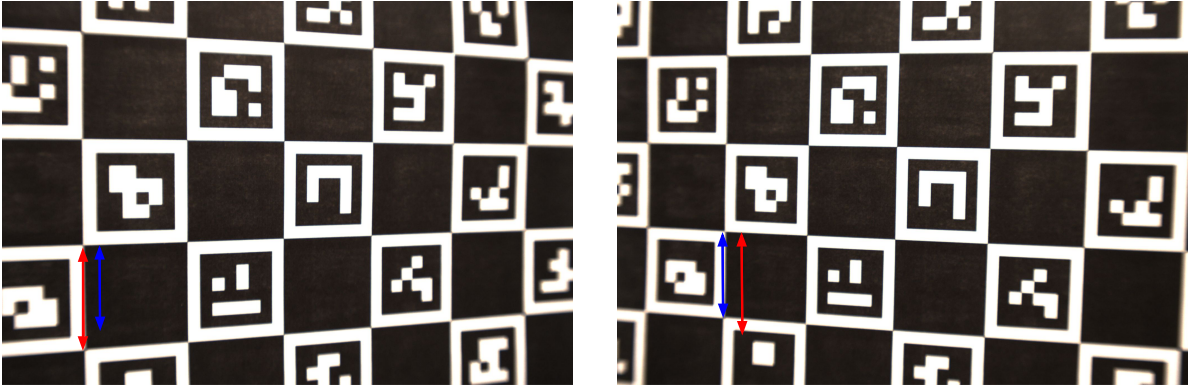


Figure 4.17: An example fixation on a ChArUco board. Scene items are not scaled equally, as some items appear larger in the left image than in the right and vice versa. The red arrow lines the side of an ArUco tag in the left image. The same tag edge in the right image is lined with a blue arrow. Placing these arrows beside each other demonstrates that scale is not preserved in a convergent camera configuration.

Some traditional stereo matching methods use patch matching as a way to use the surrounding pixels as more information for matching, instead of a singular pixel value. These methods use the same-sized patches in both images. In parallel stereo configurations, this method works well as there are no perspective differences and the scene scaling remains constant between both images. However, in convergent stereo, this is not the case. In Figure 4.17, the cameras are fixated on a close ChArUco board on a central ArUco tag. As can be seen from this pair of images, the same items in the scene appear to be different sizes between the left and right images. The same edge appears to be of different sizes in the images

due to the cameras' perspective differences. Consequently, corresponding patches will be of different shapes and sizes. Computing the patch for each pixel in the search space would increase computation load significantly, not to mention the subsequent issue of comparing patches of different shapes and sizes. The perspective effects increase as the rotational difference between the cameras increases. Other convergent methods will rectify their images so that this effect is not present, but this ignores any consequences that can come from image rectification.

4.3.2.2 Depth map from disparity

To project the horizontal and vertical disparities into depth estimates, the saved projection matrices P_l and P_r are used to triangulate the points in 3D space, resulting in a point cloud. Triangulation uses the Direct Linear Transforms (DLT) method described in [127], which calculates a matrix equation of the form $Ax = 0$. Suppose there is a 3D homogeneous point $\bar{X} = [x, y, z, w]$ viewed in both cameras which have pixel coordinates $\bar{x}_l = [u_l, v_l, 1]$ and $\bar{x}_r = [u_r, v_r, 1]$ for the left and right images respectively. Using the left camera's projection matrix, $\bar{x}_l$ can be written as $\bar{x}_l = P_l \bar{X}$. In this triangulation problem, the values of $\bar{X}$ are unknown, but can be determined by creating a form of $Ax = 0$ to solve. Since $\bar{x}_l$ and $P_l \bar{X}$ are parallel vectors, the cross product should be 0. This results in the following relationship:

$$\begin{bmatrix} u_l \\ v_l \\ 1 \end{bmatrix} \times \begin{bmatrix} p_{l1} \bar{X} \\ p_{l2} \bar{X} \\ p_{l3} \bar{X} \end{bmatrix} = \begin{bmatrix} v_l p_{l3} \bar{X} - p_{l2} \bar{X} \\ p_{l1} \bar{X} - u_l p_{l3} \bar{X} \\ u_l p_{l2} \bar{X} - v_l p_{l1} \bar{X} \end{bmatrix} = \begin{bmatrix} v_l p_{l3} - p_{l2} \\ p_{l1} - u_l p_{l3} \\ u_l p_{l2} - v_l p_{l1} \end{bmatrix} \bar{X} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Where the row vectors of P_l are written as p_{li} . Since the third row is a linear combination of the first two rows, the matrix needs to be extended to have more rows by simply finding the same equations for the right camera. This can be done as many times as there is a need to solve for multiple camera views, but in a binocular case, the goal is to solve for 2 views.

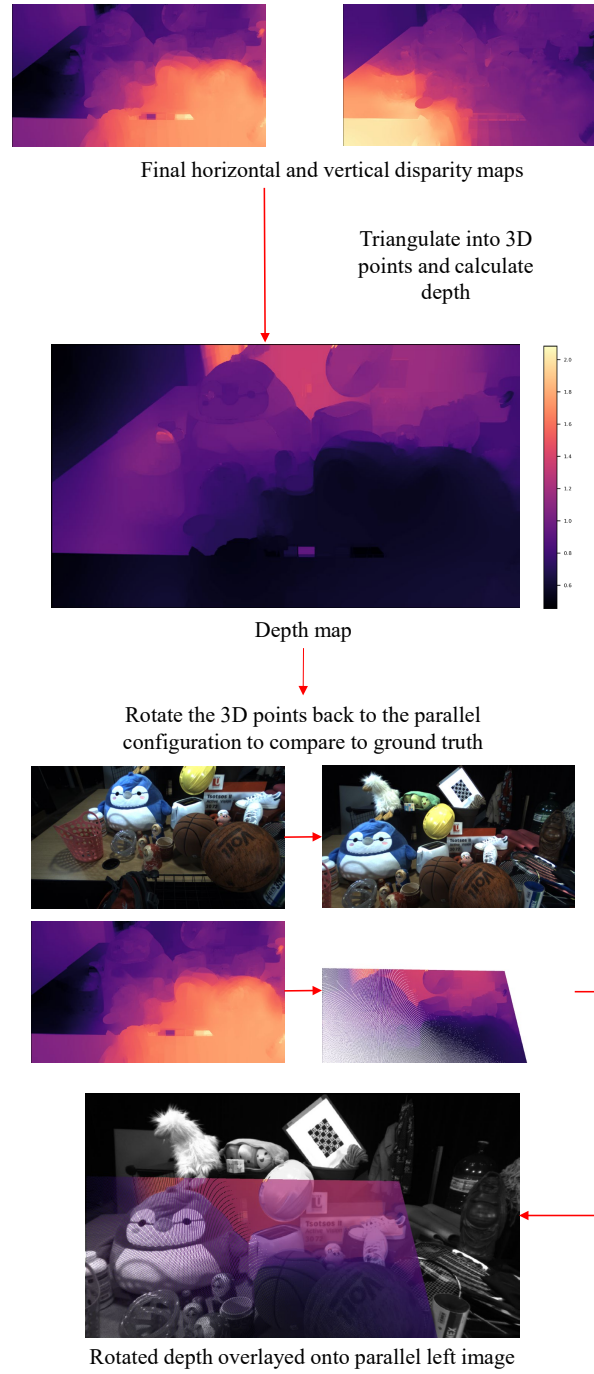


Figure 4.18: The details of how horizontal and vertical disparities are turned into a depth map, then in to a horizontal disparity map in the parallel configuration, which can be compared to the ground truth disparities in the parallel configuration. First, the convergent horizontal and vertical disparities are triangulated to 3D world points, and the distance to the left camera is taken as the depth map. Then, the 3D points are rotated so that these points are seen in the left camera's parallel configuration, and the rotated depth map is then transformed into parallel horizontal disparities which can be compared to the ground truth measurements.

$$A\bar{X} = \begin{bmatrix} v_l p_{l3} - p_{l2} \\ p_{l1} - u_l p_{l3} \\ v_r p_{r3} - p_{r2} \\ p_{r1} - u_r p_{r3} \\ \vdots \end{bmatrix} X = 0$$

With this system of equations, Singular Value Decomposition (SVD) can be used to minimize $\|A\bar{X}\|$ with $\|\bar{X}\| = 1$ since in real-world scenarios, there can be some noise that results in $A\bar{X} = w$. The solution in which w is minimized is found.

Once the 3D points have been calculated, the absolute distance of each point with respect to the left camera is found to get a depth map. This depth map is in the frame of reference of the fixated left eye. To compare this depth map to the ground-truth or to transform it back to a central frame of reference, the parallel left camera configuration, the point cloud is rotated by the inverse of the left camera's rotation between its parallel configuration and its fixated configuration. To get a depth map in the parallel configuration, take the absolute distance of the parallel point cloud. Now, the ground-truth data is in terms of disparity in the parallel configuration for simplicity and also to be able to evaluate parallel stereo methods, and so transforming the convergent disparity map with both horizontal and vertical disparity into the parallel configuration depth into parallel configuration disparity is needed for a proper comparison. This process is described in Figure 4.18

First, an array is created of homogeneous world points $\bar{X}_w = [\bar{x}_{w1}, \bar{x}_{w2}, \dots, \bar{x}_{wN}]$ that contain all the points in the pointcloud. Then, these points are projected to the homogeneous points in the left image plane with $\bar{X}_l = P_l \bar{X}_w$. Divide the homogeneous points by their last element to obtain image points. The same is done with the right image $\bar{X}_r = P_r \bar{X}_w$. To calculate the horizontal disparity of any point, take the first element of the point projected in the left image plane, which represents its horizontal position, and subtract the first element of the same point projected in the right image. For vertical disparity, take the second element of the point projected in the left image plane, which represents its vertical position, and subtract the second element of the same point projected in the right image. However, since the 3D points are projected to a parallel configuration of the cameras, one would expect

close to 0 vertical disparity. Then, the projected left image points are confirmed they be within the bounds of the actual image, and the points that fall outside of the image are discarded. The disparity values are assigned on the disparity map, and this can then be used to compare with ground-truth disparity values.

Since there are no stereo methods that compute vertical disparity, the proposed method’s output must be transformed into a form that can be compared to existing parallel methods. Thus, the comparison between the convergent stereo method and parallel methods is conducted in the parallel configuration with horizontal disparities only.

4.4 Summary

The robot used in this work is a robotic head with many degrees of freedom. Four degrees of freedom of the cameras were explored in this work, including the pan and tilt motions for both cameras. To obtain a kinematic estimation of the robot’s cameras, camera-motor calibration was conducted, and further experiments determined the expected error of the motors to understand the error behaviour of the system.

A novel convergent dataset was created through a process of data collection in a controlled environment. The parallel configuration of the cameras was determined. Ground truth data was measured using the structured light technique, which involves using gray-code patterns projected onto the scene from a projector to create accurate disparity maps. A laser module was used to confirm that the structured light method produced accurate disparities. Then, for each scene, the parallel images were taken, the ground truth disparity was measured, 5-12 fixations of the scene were taken, and all values for motor positions were recorded.

A new algorithm designed for convergent binocular images, called CAS (Convergent Active Stereo), is proposed. The first step is to determine the fundamental matrix for each fixated pair. The algorithm uses Gabor filter responses as feature vectors to match pixel pairs and does not rely on simplifications of the stereo matching problem, often leveraged by parallel stereo methods. To compare to ground truth values and state-of-the-art parallel algorithm outputs, which are in the parallel camera configuration, CAS’s disparity map output, which has both horizontal and vertical disparities in both directions, is triangulated

into 3D world points. The world points are then transformed into the parallel configuration and converted into negative horizontal disparity.

Chapter 5

Results and Analysis

This Chapter describes the analysis and evaluation conducted on the performance of CAS, the convergent stereo algorithm, to answer the three research questions proposed at the beginning of this thesis:

- RQ1.** How do the resulting depth maps of fixed parallel stereo compare to convergent stereo with multiple fixation points in terms of accuracy and map fidelity in various regions?
- RQ2.** Is there an advantage to having the ability to fixate on different areas in the scene for depth perception? If so, in which scenarios are the added degrees of freedom particularly useful?
- RQ3.** What are the consequences of rectifying convergent images to be parallel, and is it better for depth accuracy to perform disparity calculations using original convergent images?

The proposed convergent stereo algorithm, CAS, is compared with 2 traditional methods, SGBM [37] and ELAS [39], as well as 4 deep learning methods which represent some of the current most popular architectures in deep stereo vision: IGEV++ [46], MoCha [45], RAFT-Stereo [44] which will be referred to as just RAFT, and RAFT-Stereo which was trained on NeRF outputs [48] which is simplified as just NeRF-RAFT in text. Deep learning models were given images at (1544×1038) , which is close to the image size in their training sets. SGBM and ELAS were given images at 259×386 , which gave the best results with the parameters recommended. See Appendix B.2 and Appendix B.1 for examples of outputs of

the SGBM and ELAS methods at different image sizes, respectively. Since the ground truth is taken at full image resolution (3088×2076), the disparity output images are compared at this scale.

5.1 Overall results

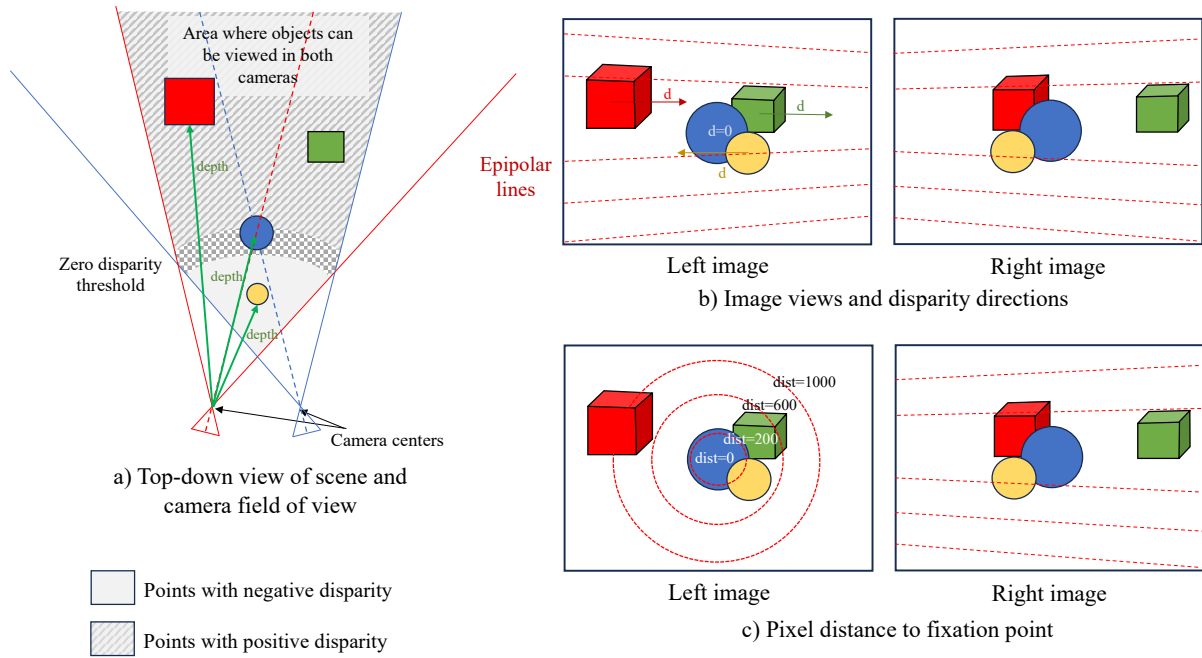


Figure 5.1: Diagram of convergent fixated images and the measurements used for analysis. a) shows the top-down view of the scene with depth measurements as the depth from the left camera to the object in 3D space. b) shows the image views from both cameras and their disparity direction. c) shows the measure of the pixel distance to the fixation point, the blue ball. Each dotted red ring represents a different pixel radius distance to the fixation point. In terms of error evaluation, at each distance, all pixels that fall within the radius will be included.

For each scene, several fixations and their disparity map outputs are compared with parallel stereo algorithms, which have one disparity map resulting from the parallel images of that scene. To find the error in disparity and depth at each fixation, the fixated disparity is converted to a disparity map that can be evaluated against the parallel ground-truth measurements using the method described in Chapter 4.3.2.2. Since there are no comparable methods which compute vertical disparity, the horizontal and vertical disparity output from

the proposed convergent stereo method is converted to depth and then projected back into only horizontal disparity in the parallel configuration to compare to parallel stereo methods.

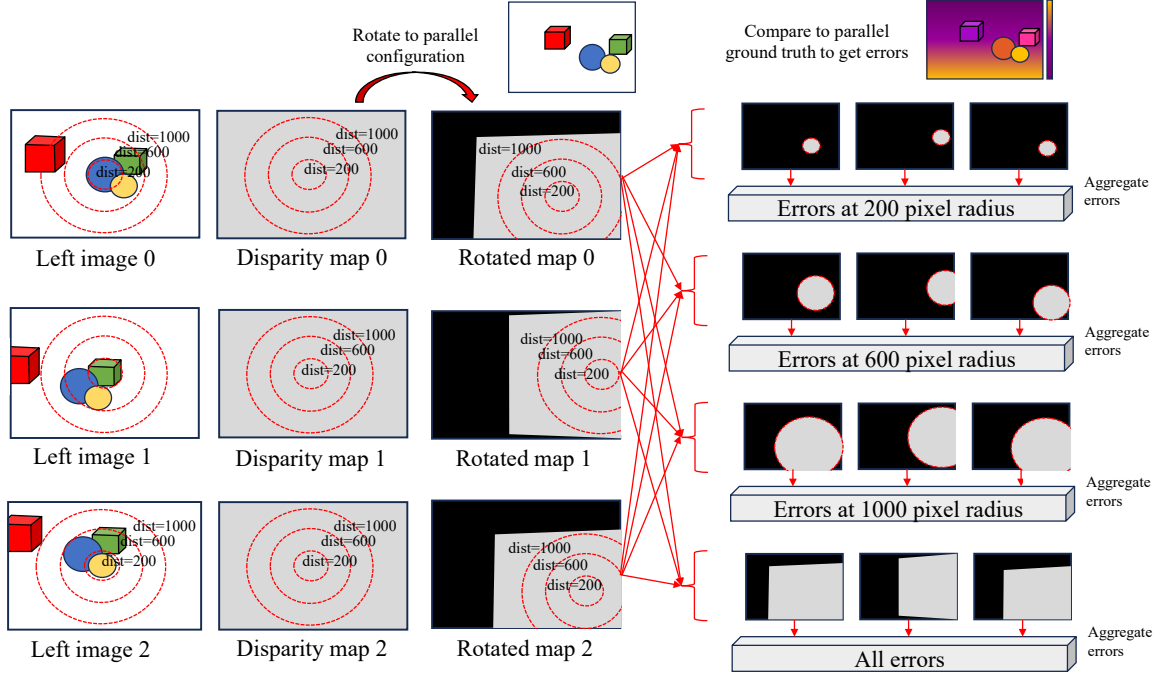


Figure 5.2: Diagram of the method used to aggregate errors for each pixel radius to fixation category. The categories include 200 pixels to fixation, 600 pixels to fixation, 1000 pixels to fixation, and the full image. For each fixation, shown in Left image 0, Left image 1, and Left image 2, there will be a disparity map output: Disparity map 0, Disparity map 1, and Disparity map 2. The disparity maps at this stage have both vertical and horizontal disparity and these disparities must be projected into 3D world points and rotated back into the parallel configuration to compare to the ground truth. This results in Rotated map 0, Rotated map 1, and Rotated map 2. Notice that the distance to fixation travels with the rotation. Then, the errors within each pixel radius to fixation category are aggregated together to form each error category. For a single pixel radius category, all the fixation errors within the category’s distance to fixation are aggregated. This process is done for all scenes. For sets of scenes, the resulting errors for each scene are then aggregated together for the entire set.

For each scene, there are multiple fixations, for each fixation, the disparity in pixels is collected into categories for the proposed convergent stereo method by Euclidean pixel distance to the fixation point. This measurement is shown in Figure 5.1 for a fixated pair of images. The analysis separates the convergent method into 4 categories based on the maximum radial distance to fixation, which would be evaluated: 200 pixels, 600 pixels, 1000 pixels, and the full image (infinity). The pixels evaluated in each category fall within the specified radius, each category evaluating more of the image than the previous. For a

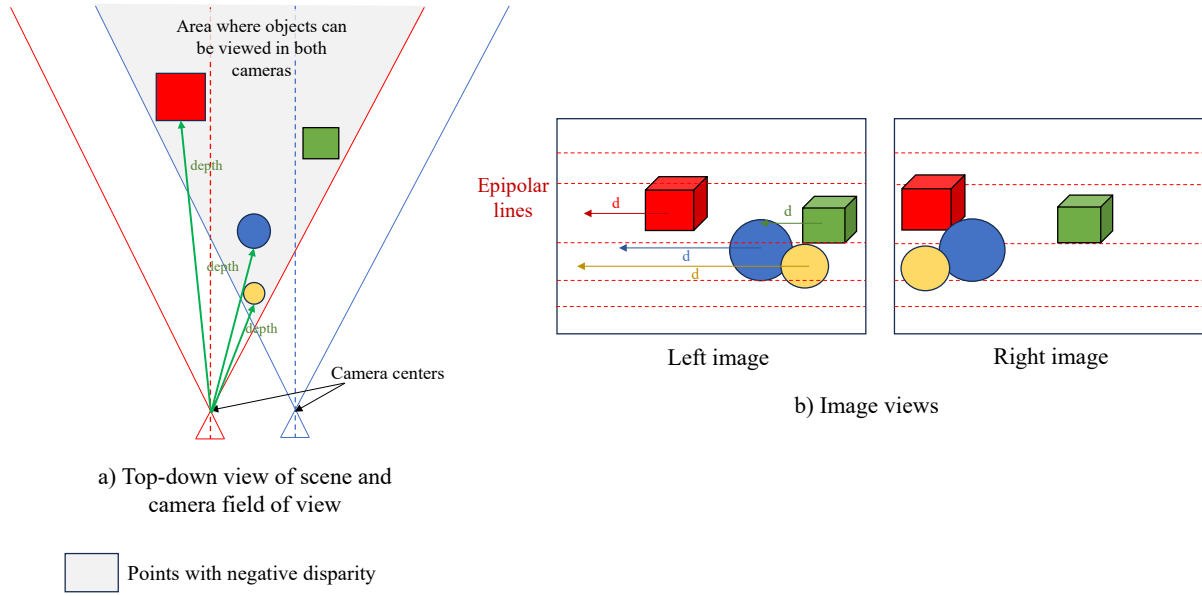


Figure 5.3: Diagram of parallel fixated images and the measurements used for analysis. a) shows the top-down view of the scene with depth measurements as the depth from the left camera to the object in 3D space. b) shows the image views from both cameras and their disparity direction.

single scene, all fixations and their errors are calculated and aggregated into their respective categories. This process is shown in Figure 5.2. For each scene, all fixations have their errors computed separately, and then each fixation's error is combined into its respective pixel distance to fixation point bins. Disparity and depth for the convergent method is compared with the ground truth in the parallel configuration. Thus, there is a need to transform the results from a convergent view to the parallel view which only uses horizontal disparities. Because of this, after the transformation, parts of the original fixated disparity map may be occluded, as can be seen in Figure 5.2 in "Rotated map 0", "Rotated map 1", and "Rotated map 2". Fixations at corners and edges of the image will have fewer pixels that are able to be evaluated. Likewise, there would not be any parallel ground truth disparity values at parts of the scene that cannot be viewed in the parallel configuration. In fixations on the edges and corners of the parallel view, the fixated images would be able to view more of the scene in that direction than in the parallel configuration.

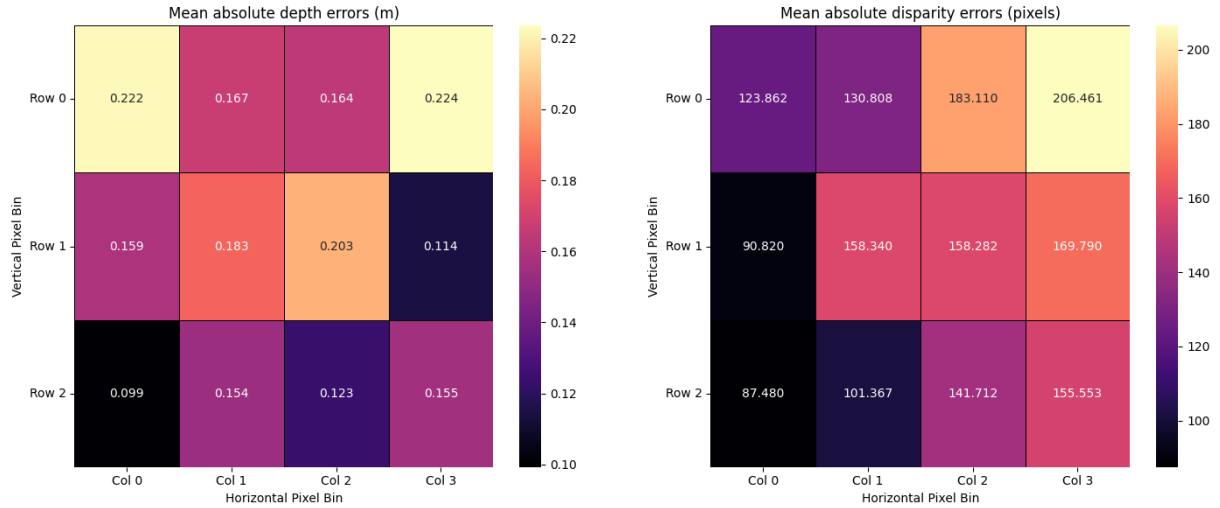
The convergent stereo method is expected to have its best performance at or near the

fixation point, while decreasing in performance as the pixel distance to fixation in the image plane increases. This is in part due to possible errors in estimating the fundamental matrix. Although it is known that the centers of the images correspond due to the images' fixated nature, any error in the fundamental matrix can lead to a large effect on the placement of epipolar lines, especially near the edges of the image. Additionally, in close convergence scenarios, large parts of the image will not contain corresponding points due to the smaller overlapping areas in the field of view. In these cases, only pixels that are close to the center of the image will have good correspondence estimates.

For evaluating parallel methods, the full image is evaluated and depth is reported in the reference frame of the left camera by typical convention as shown in Figure 5.3.

Table 5.1: Distribution of fixations in all scenes in the left image parallel view by dividing the left image into a 4×3 grid.

9	17	18	25
3	38	42	46
6	11	19	15



(a) Mean absolute depth error (m) divided into fixation areas. (b) Mean absolute horizontal disparity error (pixels) divided into fixation areas.

Figure 5.4: Error distribution of all scenes divided into fixation areas.

The view of the left camera in the parallel configuration is divided up into a 4×3 grid. The distribution of fixation points across all scenes on this grid is shown in Table 5.1.

The leftmost part of the grid has few fixations due to there being less ground truth points available from camera-motion parallax. This effect can be seen in Figure 4.12 where the black pixels of the ground truth disparity map are unassigned. These are points that cannot be viewed in the right image. The goal during data creation and collection was to create a wide variety of interesting scenes with the focus centered in both cameras' fields of view. The fixations were also chosen to view scene objects; thus, there are more fixations in the middle of the image. Figure 5.4 shows for the convergent stereo method, the distribution of mean absolute depth error in (m) and the distribution of mean absolute horizontal disparity error (pixels). The top right corner for both metrics shows the highest amount of error.

Some examples of fixations in the upper right corner are shown in Figure 5.5. In these images, the large error areas are points that are not at a similar depth as the fixation point. Namely, the top two examples show that the areas with the most error come from the wall in the background, which is at a farther distance than the point of fixation and is not the central focus of the scene. The two examples in the middle show low errors in disparity and depth. These examples' ground truth does not have a large variation in depth, as in these scenes, only the foreground was detected using the structured light method for ground truth collection. The bottom two examples also show a large disparity and depth error, but towards the bottom of the scene. This is most likely due to the different depths of the points. Seemingly, the points in these examples that are closer than the point of fixation have the largest amount of error. At the point of fixation, most examples display a low depth and disparity error.

Taking a look at other examples that have higher disparity and depth errors in Figure 5.6, all of these examples have a wide range of depths in the scene structure. Pixels that exhibit higher disparity and depth error are those that are not at the same depth as the point of fixation. Pixels near or at the same depth as the point of fixation tend to have low error in both disparity and depth metrics. The top three examples have their largest error at points that are farther from the fixation point. This may be due to the fixation points being close to the closest point in the scene to the robot head. The bottom two examples show large errors in points that are closer than the fixation point. In these fixations, there are many objects that are closer to the robot head than the fixation point.

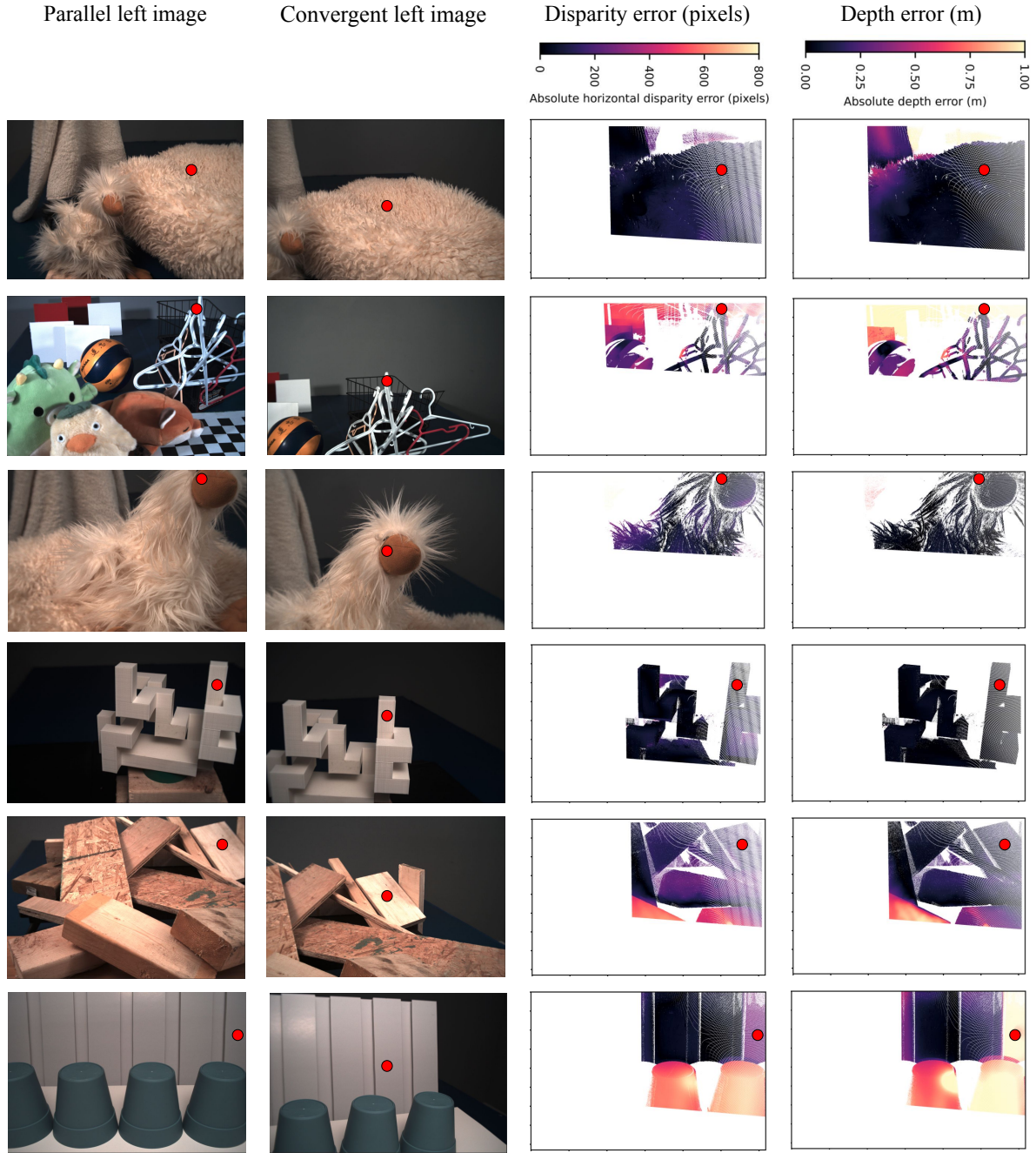


Figure 5.5: Example outputs of disparity and depth error maps rotated to parallel camera configuration with fixations in the upper right corner. Disparity error maps are scaled by the scale displayed at the top of the Disparity error (pixels) column, and depth error maps are scaled by the scale displayed at the top of the Depth error (m) column. The first column displays the left image of the scene in the parallel configuration, and the second column displays the left image in a fixation. The more yellow the pixel, the higher the error. Fixation points are marked with a red dot. In the Parallel left images, the red dot indicates where the Convergent left image will fixate on.

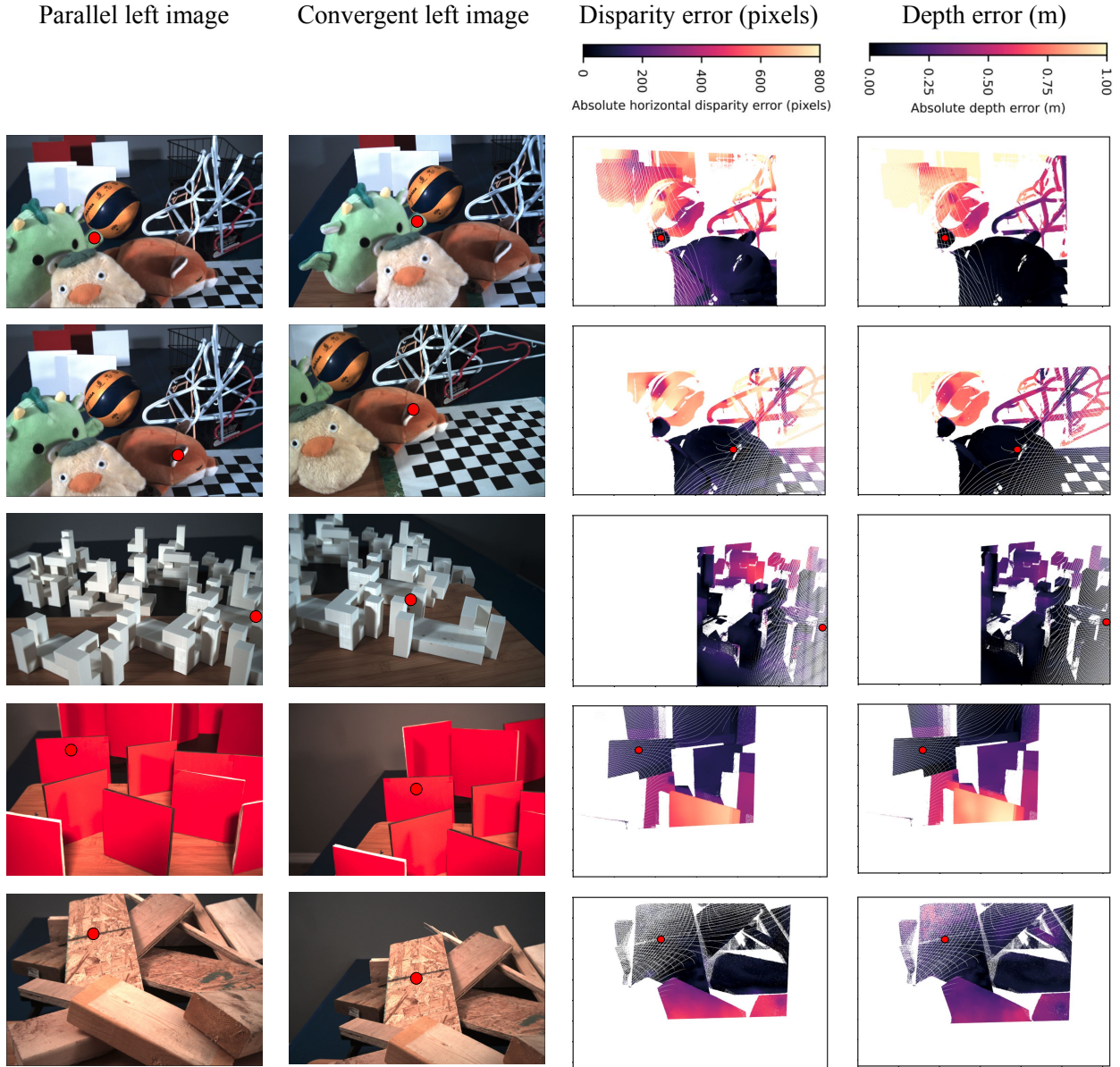


Figure 5.6: Example outputs of disparity and depth error maps rotated to parallel camera configuration. These examples show fixations with larger errors. Disparity error maps are scaled by the scale displayed at the top of the Disparity error (pixels) column, and depth error maps are scaled by the scale displayed at the top of the Depth error (m) column. The first column displays the left image of the scene in the parallel configuration, and the second column displays the left image in a fixation. The more yellow the pixel, the higher the error. Fixation points are marked with a red dot. In the Parallel left images, the red dot indicates where the Convergent left image will fixate on.

Referring back to Figure 5.4, it may be the case that fixations at the top right corner of the left parallel image have views that cover a wide range of depths, spanning from objects

close to the camera to the wall in the background. Fixations near the bottom of the image would require the camera to tilt downwards, and the camera’s fields of view in this case would not be able to view the background wall. These examples suggest that a limitation of the system is large depth variations with respect to the fixation point. In humans, it is also expected that the only areas with good binocular fusion are within the horopters, which centers around the fixation point [10, 11]. With the ability to fixate on different parts of the image to get depth estimates at points of interest, this limitation does not necessarily handicap the system. The robot can fixate on multiple parts of the scene to obtain good estimates of depth throughout the whole scene instead of relying on one fixation to estimate the depths of all scene objects.

Table 5.2: Absolute depth errors (m) and absolute disparity errors (pixels) across all scenes reported in mean, median, and STD values. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in yellow. The 2nd-best result for each column is highlighted in gray. The 3rd-best result for each column is highlighted in light brown.

	Depth error (m)			Disparity error (pixels)		
	Mean	Median	STD	Mean	Median	STD
CAS-200	0.13	0.07	0.20	122	96	115
CAS-600	0.15	0.08	0.22	138	106	132
CAS-1000	0.16	0.09	0.24	150	110	145
CAS-full	0.17	0.10	0.25	158	113	154
-----	-----	-----	-----	-----	-----	-----
SGBM	2.19	1.42	1.86	545	528	403
ELAS	1.98	1.19	1.92	449	453	358
IGEV++	0.67	0.08	1.20	254	74	343
MoCha	0.63	0.09	1.07	274	82	351
NeRF-RAFT	0.51	0.05	0.93	210	56	277
RAFT	1.27	0.60	1.59	338	287	328

To analyse the overall quantitative errors of the system compared to state-of-the-art methods, Table 5.2 shows the averaged mean, median, and standard deviation (STD) of the absolute depth error in metres and the absolute horizontal disparity error in pixels across all fixations and all scenes compared to the state-of-the-art. Average absolute disparity error and 50th percentile error are both common error metrics in stereo benchmarks. In terms of overall absolute disparity error across all scenes, the CAS-200 method performs the best with the lowest mean disparity error and the lowest standard deviation. CAS outperforms

all parallel methods in mean disparity and standard deviation at all pixel radii to fixation categories.

It is also interesting to analyse depth error as the proposed convergent stereo system is to be utilized in mobile robotics applications. Aggregating results over all scenes, in Table 5.2, the proposed CAS convergent stereo method performs the best in terms of mean depth error and the standard deviation of depth error. CAS-200 come as a close second to NeRF-RAFT in median depth error. Separating the CAS convergent method by radius to fixation point, depth errors at smaller radii are significantly lower than parallel methods, slowly increasing in error in terms of the mean, median, and standard deviation as the radius increases. CAS-200, CAS-600, and CAS-1000 perform better in mean depth error against all the parallel methods. CAS maintains the lowest standard deviation of error in both disparity and depth error. This means that even though the proposed method does not achieve perfect disparities, its estimates are consistently within an acceptable range of the ground truth values.

ELAS and SGBM often produced disparity maps where many parts of the image were empty or unassigned. Their parameters were never altered between different experiments in this thesis, yet they produced vastly different qualities of disparity maps. ELAS and SGBM performed particularly poorly with scenes that lacked many visual cues. Although SGBM would provide more disparity estimates in general, the estimates were in scattered patches and were not coherent most of the time. ELAS, on the other hand, tends to only match patches where it has a high level of confidence. However, in many subsets of the dataset, purposeful ambiguities were placed in the scene, causing ELAS to leave much of the image unassigned.

A typical metric used to evaluate stereo methods is the percentage of pixels over x horizontal disparity error [16, 49, 50, 51, 52, 54, 58, 59]. This metric describes the percentage of pixels in the output disparity maps which have a disparity error greater than a given horizontal x disparity threshold. It’s commonly used due to its simplicity and ability to provide a general sense of accuracy. A lower percentage of bad pixels means a larger number of correct pixels within some threshold. For example, for the disparity threshold of $x = 1$ in the first column of Table 5.3, the best result was ELAS which had a 95.63% score. This means that 95.63% of all of their pixels had a disparity error of over 1 pixel. Say a method

Table 5.3: Evaluations of the chosen stereo methods with the percentage of bad pixels metric. The values reported are the percentage of pixels above the specified disparity error. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in yellow. The 2nd-best result for each column is highlighted in gray. The 3rd-best result for each column is highlighted in light brown.

Disparity (pixels)	1	4	10	40	100	200	300	400
CAS-200	99.41	97.67	93.64	78.03	46.06	15.62	5.28	2.70
CAS-600	99.44	97.78	94.48	79.12	51.35	20.47	8.72	4.32
CAS-1000	99.47	97.89	84.78	79.70	52.57	22.99	10.82	5.79
CAS-full	99.49	97.98	95.05	80.28	53.97	26.11	13.76	7.96
SGBM	99.63	98.56	96.69	93.39	89.97	81.74	75.40	65.83
ELAS	95.63	92.01	86.38	76.70	71.13	66.13	63.95	57.06
IGEV++	96.99	87.96	73.57	56.82	47.71	39.84	33.52	28.78
MoCha	95.95	84.14	67.15	56.78	52.23	44.13	40.12	34.40
NeRF-RAFT	96.55	86.70	70.83	52.17	41.79	33.94	28.68	19.98
RAFT	97.52	89.81	76.49	62.36	54.22	49.95	46.90	41.20

had a score of 2% with a 1 pixel threshold, this would mean that of all their pixels in their disparity map, only 2% of these pixels had a higher disparity error than 1 pixel. This metric is suitable if the user only cares about the number of correctly assigned disparities. Ignoring depth, this metric purely evaluates the accuracy of a method in matching corresponding pixels. Benchmarks such as KITTI and Middlebury use the percentage of bad pixels as one of the primary evaluation metrics. Using this metric, it can be seen in Table 5.3 that for thresholds past 200 pixels of disparity error, the proposed convergent stereo method, CAS and its variants, for all pixel distances to the fixation point, has the lowest amount of disparity error. However, for smaller thresholds, deep learning parallel methods have a lower amount of disparity error. This is unsurprising as deep learning stereo methods have focused very heavily on performing well under this metric, as it is widely used in benchmarks. The convergent stereo method relies on an accurate estimate of camera orientation, and slight deviations can result in different disparity estimates when projecting disparity into 3D points and rotating back to the parallel orientation to compare to ground truth measurements. Despite this, the CAS method does have the lowest percentage of pixels with larger pixel error thresholds. This means that even though the proposed method does not achieve perfect disparities, its estimates are consistently within a small range of the ground truth values,

whereas deep methods may have more pixels within small disparity error ranges but also have extremely incorrect disparities outside of this range. As shown in Table 5.3, CAS performs best in the percentage of bad pixels metric for disparities at and over 200 pixels for all pixel radii to the fixation point, indicating that the proposed method maintains its disparity error to a more confined range than the parallel state-of-the-art methods.

Although most stereo benchmarks only evaluate disparity error as a measure for performance, for the purposes of depth perception for a mobile robot performing active vision, depth needs to be evaluated alongside disparity. A disparity error at a close target has a smaller consequence on depth estimation than the same disparity error on a farther target. Evaluations may use a percentage of pixels with a disparity error above x , x ranging from anywhere between 0.5 pixels to 4.0 as a metric. However, this method does not have any sense of the magnitude of error for the rest of the pixels that do not fall under x disparity error. For example, in an image with 1000 pixels, if the 0.5 bad pixel percentage is 2% it means that only 20 pixels have above 0.5 disparity error. Yet, it is not known what the errors of these 20 pixels are. They could have a 200 pixel disparity error, or they could have a 1 pixel disparity error, resulting in vastly different meanings in terms of performance. Additionally, a high percentage of bad pixels value is extremely uninformative of how well a system performs. ELAS having the lowest percentage at the 1 pixel disparity threshold with 95.63% of its pixels having a worse error than 1 pixel does not reveal much. It is still unknown what the error of the rest of the 95.63% of pixels have. For the purposes of evaluating convergent stereo in the context of mobile active robots, evaluation will extend to depth error as well.

To evaluate depth error, a new error metric is proposed: the percentage of pixels above x depth error in metres. This metric is similar to the bad pixels metric for disparity, but instead, the threshold is based on 3D depth. The percentage of pixels that have a 3D depth error of over the specified amount is reported. A lower number is better. The results are shown in Table 5.4, which indicates that CAS-200 outperforms all other methods at a 0.15m threshold and above. NeRF-RAFT has the best values for under 0.05m and 0.10m error. However, NeRF-RAFT also has around 20% of its pixels having errors larger than 0.40m. Even with evaluating the full image of the proposed convergent method CAS-full, it has less

Table 5.4: Evaluations of the chosen stereo methods with the percentage of bad depths metric. The values reported are the percentage of pixels above the specified depth error. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in yellow. The 2nd-best result for each column is highlighted in gray. The 3rd-best result for each column is highlighted in light brown.

Depth (m)	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40
CAS-200	58.39	38.77	25.23	16.52	11.87	8.99	6.96	5.84
CAS-600	63.44	43.50	29.60	20.45	14.51	10.21	7.60	6.06
CAS-1000	66.13	46.03	32.14	22.63	16.18	11.58	8.68	7.03
CAS-full	68.97	48.90	34.94	24.90	18.07	13.20	9.74	7.93
SGBM	95.16	93.33	91.80	89.32	86.93	85.03	83.18	81.66
ELAS	81.32	76.24	72.82	70.59	69.51	68.75	68.34	67.91
IGEV++	54.77	45.47	41.45	38.74	36.79	35.46	34.28	33.18
MoCha	57.77	51.54	45.34	40.38	36.62	35.06	33.88	32.62
NeRF-RAFT	46.26	36.60	29.23	24.73	22.82	21.55	20.89	20.45
RAFT	60.74	53.88	50.28	47.70	46.08	43.97	42.99	42.50

than 8% of its pixels with over 0.40m depth error.

Overall, the CAS convergent stereo algorithm has the lowest mean and standard deviation of depth and disparity error among the 33 scene sets in Table 5.2. The CAS method also performs well in the percentage of bad pixels metric for disparity and depth for higher thresholds. Some scenes are purposely made to be extraordinarily difficult for stereo matching, which may result in typically well-performing state-of-the-art methods failing. Having camera geometry and fixation information can give a valuable advantage in scenarios where image and scene context do not provide enough reliable information for matching. The CAS convergent stereo system performs well in easy stereo scenes, but not as well as state-of-the-art methods. However, simply evaluating stereo matching based on information-rich scenes alone avoids rigorous evaluation; rather, to fully test the limitations of a stereo matching method, difficult scenes and scenarios should be considered. The analysis above evaluates the stereo methods on all scenes of the dataset regardless of scene structure, depth of objects, and object surfaces. To better understand the performance of the proposed convergent stereo system, the CAS convergent stereo system was compared to parallel methods in different scenarios, including those that are considered typical in stereo benchmarks. These results are detailed in the sections that follow.

5.1.1 Normal Objects Subset

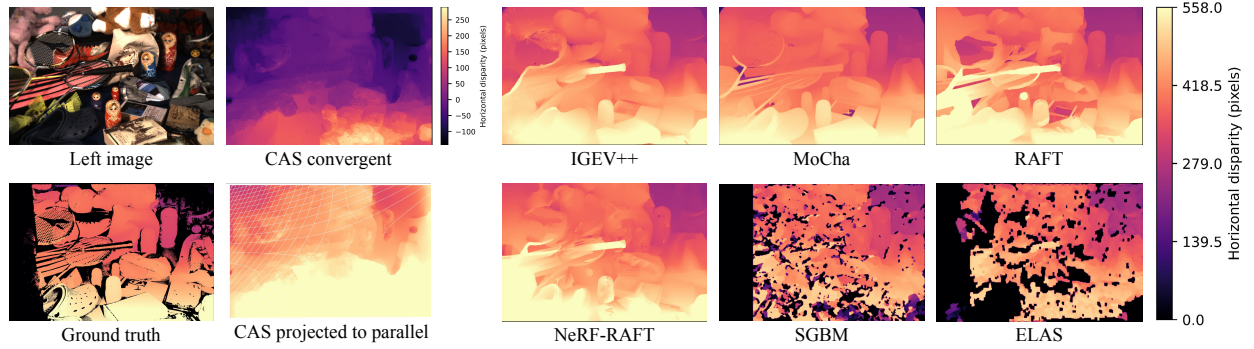


Figure 5.7: Example output disparity images of one of the scenes in the “Normal” subset. The “Left image” shows the left image of the parallel image pair. “Ground truth” is the ground truth horizontal disparity in the parallel configuration measured from structured light. “CAS convergent” shows the horizontal disparity output of one of the many fixations taken for the scene with a different color scale than the other disparity maps. “CAS projected to parallel” shows the results of transforming the convergent vertical and horizontal disparity to horizontal disparity in the parallel configuration to be able to compare to the ground truth measurements. The outputs of the state-of-the-art parallel methods are the parallel methods’ disparity outputs for this scene. The scale is shown with the color bar on the right in pixel units.

The subset of “Normal” scenes was selected from scenes that contained a large amount of textures, objects, and unique features, which is suitable for stereo matching. This set also contains scenes that include objects that would be typically seen in a household and in standard stereo matching benchmarks. Refer to Appendix A.6 for all parallel left images of scenes in this subset. An example scene and the disparity maps of each method are displayed in Figure 5.7. Unsurprisingly, deep methods perform very well on this scene, and although the traditional methods have a few unassigned patches, they perform decently in this scene as well.

Although the CAS method performs decently in mean depth error, mean disparity error, and has the lowest standard deviations in both metrics, deep learning methods like NeRF-RAFT perform the best overall in the metrics displayed in Table 5.5. Looking at Table 5.6 for the bad pixel disparity metric, it can be seen that deep methods typically perform the best in lower disparity error thresholds. This is expected of these deep learning methods as they are most typically evaluated with the bad pixels metric for the most prominent benchmarks. The proposed method at 200, 600, and 1000 pixel radii to fixation performs

Table 5.5: Absolute depth errors (m) and absolute disparity errors (pixels) across the “Normal” objects subset reported in mean, median, and STD values. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in yellow. The 2nd-best result for each column is highlighted in gray. The 3rd-best result for each column is highlighted in light brown.

	Depth error (m)			Disparity error (pixels)		
	Mean	Median	STD	Mean	Median	STD
CAS-200	0.14	0.08	0.21	113	90	107
CAS-600	0.16	0.10	0.24	124	100	121
CAS-1000	0.17	0.11	0.27	136	105	135
CAS-full	0.18	0.11	0.27	142	107	147
SGBM	1.76	0.77	1.93	516	380	581
ELAS	0.72	0.06	1.50	178	23	368
IGEV++	0.19	0.03	0.62	142	22	348
MoCha	0.24	0.02	0.51	200	12	382
NeRF-RAFT	0.06	0.02	0.14	76	19	155
RAFT	0.24	0.03	0.47	139	26	259

the best at and past the 200 pixel error mark, indicating a lower amount of highly wrong pixel disparities when compared to parallel methods.

In terms of percentage of bad pixels with depth error in Table 5.7, NeRF-RAFT performs the best with this metric, while the proposed CAS method at 200 and 600 pixel distance to fixation radius comes second and third best past the 0.20m threshold. The proposed method performs decently well in normal scenes.

Since this set contains typical images one would find in a tabletop stereo dataset, these results, which indicate that parallel stereo methods are stronger, are expected. However, when the performance difference across the board is decent, it is not an interesting evaluation of the limitations of each method. Rather, this subset serves to show that under normal situations, these methods behave similarly to their reported results.

Table 5.6: Evaluations of the chosen stereo methods with the percentage of bad pixels metric on the “Normal” subset. The values reported are the percentage of pixels above the specified disparity error. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in yellow. The 2nd-best result for each column is highlighted in gray. The 3rd-best result for each column is highlighted in light brown.

Disparity (pixels)	1	4	10	40	100	200	300	400
CAS-200	99.45	97.75	93.65	79.06	44.27	14.41	2.57	0.70
CAS-600	99.49	97.93	94.64	78.47	48.76	16.94	4.92	1.88
CAS-1000	99.5	98.00	94.91	79.17	50.71	18.99	6.74	2.99
CAS-full	99.49	97.98	95.04	80.28	53.97	26.10	13.76	7.96
SGBM	99.33	97.43	93.96	88.77	83.96	75.16	67.85	61.25
ELAS	91.61	86.58	77.31	60.92	52.54	47.79	45.49	43.90
IGEV++	97.10	88.55	72.42	50.53	39.93	31.82	28.17	26.59
MoCha	95.43	83.26	65.06	53.40	47.89	43.98	40.55	35.64
NeRF-RAFT	96.45	85.98	68.13	44.81	33.30	24.01	19.32	16.28
RAFT	96.97	88.24	71.52	49.57	38.23	33.18	30.10	27.90

Table 5.7: Evaluations of the chosen stereo methods with the percentage of bad depths metric on the “Normal” subset. The values reported are the percentage of pixels above the specified depth error. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in yellow. The 2nd-best result for each column is highlighted in gray. The 3rd-best result for each column is highlighted in light brown.

Depth (m)	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40
CAS-200	59.16	42.51	28.53	17.74	10.08	7.45	6.57	4.81
CAS-600	65.93	48.54	33.79	22.12	12.66	9.97	7.91	6.73
CAS-1000	69.10	50.73	35.93	24.31	15.81	11.59	9.23	7.87
CAS-full	68.97	48.89	34.94	24.89	18.07	13.20	9.73	7.93
SGBM	91.28	88.14	85.95	83.40	80.32	77.76	75.30	73.42
ELAS	69.39	59.89	55.90	53.80	52.11	51.08	50.57	50.17
IGEV++	50.45	36.90	32.00	29.79	28.76	27.82	26.95	26.24
MoCha	54.43	48.36	44.82	39.77	36.63	35.24	34.17	32.33
NeRF-RAFT	39.37	24.73	13.07	6.85	4.35	3.09	2.51	2.12
RAFT	47.27	34.59	28.41	24.39	22.43	19.04	17.68	17.21

5.1.2 Far Objects Subset

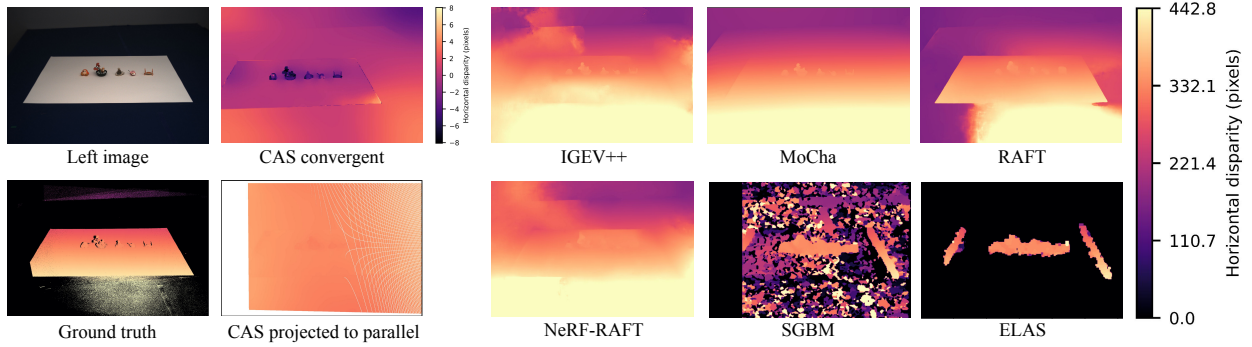


Figure 5.8: Example output disparity images of one of the scenes in the “Far” subset. The “Left image” shows the left image of the parallel image pair. “Ground truth” is the ground truth horizontal disparity in the parallel configuration measured from structured light. “CAS convergent” shows the horizontal disparity output of one of the many fixations taken for the scene with a different color scale than the other disparity maps. “CAS projected to parallel” shows the results of transforming the convergent vertical and horizontal disparity to horizontal disparity in the parallel configuration to be able to compare to the ground truth measurements. The outputs of the state-of-the-art parallel methods are the parallel methods’ disparity outputs for this scene. The scale is shown with the color bar on the right in pixel units.

The “Far” subset includes the scenes that have objects that are farther from the stereo head baseline and have smaller disparities. The depths of the points that were fixated on ranged between 1.5m to 3m in this set. Refer to Appendix A.4 for all images of the parallel left images of scenes in this subset. It is expected that the convergent stereo method may not perform as well with targets that are farther away, due to the error margin increasing with larger uncertainty from vergence at larger depths. Interestingly, vergence cue reliance in humans was found to decrease with increasing target depth [14]. The effects of decreased vergence cue reliability can be seen in the results shown in Tables 5.8 and 5.10. From the tables, ELAS had better results than SGBM, though from the example scene shown in Figure 5.8, both algorithms left a large amount of the image unassigned.

The relatively mediocre performance of the CAS algorithm on this set can be attributed to the increasing error expected at larger depths. Not only does the volume of space that a fixation at the center pixel represents increase with increased depth, increasing the uncertainty of the depth estimate, but the resolution of the motors becomes more relevant. The difference between fixating at infinity and fixating at 3m is a 1-degree rotation inwards hor-

Table 5.8: Absolute depth errors (m) and absolute disparity errors (pixels) across the “Far” objects subset reported in mean, median, and STD values. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in yellow. The 2nd-best result for each column is highlighted in gray. The 3rd-best result for each column is highlighted in light brown.

	Depth error (m)			Disparity error (pixels)		
	Mean	Median	STD	Mean	Median	STD
CAS-200	0.38	0.26	0.37	102	81	78
CAS-600	0.49	0.33	0.46	135	123	97
CAS-1000	0.49	0.34	0.47	145	130	104
CAS-full	0.47	0.30	0.45	145	130	105
SGBM	1.62	0.79	1.82	157	135	125
ELAS	0.47	0.04	1.2	43	10	80
IGEV++	0.37	0.14	0.60	124	45	177
MoCha	0.21	0.03	0.44	57	9	132
NeRF-RAFT	0.53	0.37	0.52	248	214	230
RAFT	0.40	0.22	0.45	62	30	71

izontally from both eyes. As such, a slight calibration noise becomes more consequential as well. The ability to resolve fine sub-degrees of motion is needed for larger depths. However, since this convergent stereo system is intended to be mounted on a mobile robotic platform to perform active vision tasks, the indication that objects are far can trigger a motion to move closer to the target, which would improve depth estimates significantly.

Table 5.9: Evaluations of the chosen stereo methods with the percentage of bad pixels metric on the “Far” subset. The values reported are the percentage of pixels above the specified disparity error. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in yellow. The 2nd-best result for each column is highlighted in gray. The 3rd-best result for each column is highlighted in light brown.

Disparity (pixels)	1	4	10	40	100	200	300	400
CAS-200	99.60	98.32	95.14	71.96	39.97	15.00	0.64	0.32
CAS-600	99.50	97.95	94.68	77.43	53.62	23.63	5.67	0.12
CAS-1000	99.54	98.13	84.21	80.02	57.69	27.35	8.60	3.83
CAS-full	99.56	98.19	95.38	81.10	59.02	27.41	9.30	4.75
SGBM	98.42	94.18	87.86	77.10	61.95	37.41	20.16	7.17
ELAS	90.13	67.48	43.87	19.49	14.89	8.94	3.34	0.21
IGEV++	96.35	86.54	63.32	33.09	24.78	16.50	14.65	12.33
MoCha	92.38	67.56	15.26	11.80	8.26	6.47	6.34	4.69
NeRF-RAFT	96.00	84.26	64.61	48.91	45.39	40.99	31.29	23.15
RAFT	96.67	86.60	59.89	30.29	18.66	7.57	1.72	0.00

Table 5.10: Evaluations of the chosen stereo methods with the percentage of bad depths metric on the “Far” subset. The values reported are the percentage of pixels above the specified depth error. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in yellow. The 2nd-best result for each column is highlighted in gray. The 3rd-best result for each column is highlighted in light brown.

Depth (m)	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40
CAS-200	89.24	79.05	69.03	59.17	50.69	44.16	36.49	30.89
CAS-600	93.19	85.96	77.28	66.94	59.49	52.78	46.02	40.51
CAS-1000	92.54	84.96	76.48	67.28	60.06	53.76	48.13	43.28
CAS-full	91.61	83.38	74.74	65.66	58.04	51.41	46.09	41.74
SGBM	86.16	81.28	77.77	74.87	72.50	70.52	68.15	65.98
ELAS	34.94	23.92	20.04	18.19	17.28	16.51	15.96	15.52
IGEV++	52.20	36.74	31.89	30.32	28.52	26.12	23.64	21.38
MoCha	23.10	17.44	15.84	14.54	13.43	12.54	11.77	11.11
NeRF-RAFT	58.44	53.99	50.87	48.19	45.88	44.66	43.48	41.51
RAFT	52.69	40.30	36.60	34.27	31.92	30.00	28.29	26.64

5.1.3 Close Subset

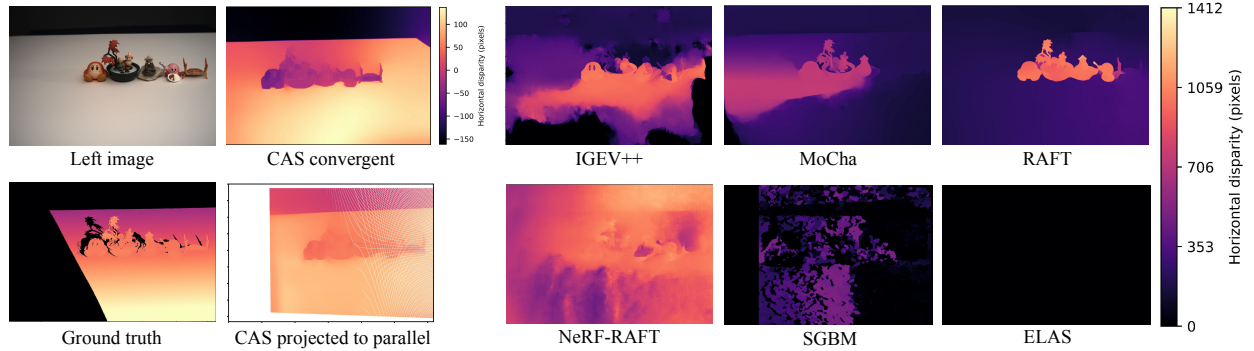


Figure 5.9: Example output disparity images of one of the scenes in the “Close” subset. The “Left image” shows the left image of the parallel image pair. “Ground truth” is the ground truth horizontal disparity in the parallel configuration measured from structured light. “CAS convergent” shows the horizontal disparity output of one of the many fixations taken for the scene with a different color scale than the other disparity maps. “CAS projected to parallel” shows the results of transforming the convergent vertical and horizontal disparity to horizontal disparity in the parallel configuration to be able to compare to the ground truth measurements. The outputs of the state-of-the-art parallel methods are the parallel methods’ disparity outputs for this scene. The scale is shown with the color bar on the right in pixel units.

A set of scenes which the main objects in the scene were in close proximity of less than 50cm from the camera baseline was selected to form the “Close” subset. Please refer to Appendix A.2 for all of the left parallel images of this set. In Table 5.11, CAS at all fixation radii of 200, 600, and 1000 pixels outperforms all other methods in depth error except for NeRF-RAFT in median depth error. In terms of disparity error, the proposed method at 200, 600, and 1000 fixation radii outperforms all other methods in terms of mean disparity error and standard deviation. NeRF-RAFT has the lowest median disparity error.

Figure 5.9 shows an example scene from this set and the outputs of disparity maps from the methods evaluated. The disparity map shown for the proposed convergent stereo method is the horizontal disparity map from one fixation out of the many fixations taken. It is interesting to compare the disparity outputs with the error results. ELAS and SGBM were not able to handle this scene. This is expected as the maximum disparity of the implementation of these algorithms do not span past a quarter of the image size, indicating a limit perceivable depth range for these methods. Looking at just the mean depth and disparity errors shown in Table 5.9, the convergent stereo method CAS and NeRF-RAFT

perform the best on the close subsets. However, analysing the images qualitatively, it can be seen that NeRF-RAFT suffers from hallucinations in areas of the platform that do not have many image features. Many of the deep methods suffer from not being able to assign reasonable disparity values to these parts of the image. RAFT, for example, can assign most of the individual objects correctly, yet they are floating in space with adjacent locations having small disparity values, meaning a large distance away from the cameras. MoCha was only able to assign half of the objects and part of the scene correctly, while the rest of the scene was assigned to be very far away. IGEV++ also suffers from assigning patches of the image to extremely incorrect disparity values.

Table 5.11: Absolute depth errors (m) and absolute disparity errors (pixels) across the “Close” objects subset reported in mean, median, and STD values. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in yellow. The 2nd-best result for each column is highlighted in gray. The 3rd-best result for each column is highlighted in light brown.

	Depth error (m)			Disparity error (pixels)		
	Mean	Median	STD	Mean	Median	STD
CAS-200	0.050	0.020	0.160	135	104	157
CAS-600	0.060	0.030	0.160	151	115	172
CAS-1000	0.080	0.040	0.190	170	126	193
CAS-full	0.090	0.050	0.220	186	130	214
SGBM	2.680	2.130	1.960	963	902	422
ELAS	2.610	2.080	1.880	1049	942	535
IGEV++	0.860	0.310	1.300	597	398	559
MoCha	1.100	0.970	0.870	832	833	395
NeRF-RAFT	0.080	0.020	0.190	176	46	260
RAFT	0.710	0.060	1.080	434	212	449

Table 5.12 shows the performance of each method using the percentage of bad pixels above x disparity error metric. The table suggests that ELAS has the best performance out of all the methods in disparity errors under 10 pixels. This is misleading since ELAS will leave large portions of the image unassigned. Those pixels that are assigned have a high confidence value. At and past 100 pixels, CAS at all fixation radii outperforms all other methods. At the 400 pixel threshold, the next best parallel method has around 20% more bad pixels than CAS. This may be due to the deep methods’ tendency to hallucinate clouds of extremely wrong disparities.

Table 5.12: Evaluations of the chosen stereo methods with the percentage of bad pixels metric on the “Close” subset. The values reported are the percentage of pixels above the specified disparity error. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in yellow. The 2nd-best result for each column is highlighted in gray. The 3rd-best result for each column is highlighted in light brown.

Disparity (pixels)	1	4	10	40	100	200	300	400
CAS-200	99.53	98.07	94.46	77.93	51.19	17.95	4.73	1.47
CAS-600	99.45	97.77	94.37	79.98	54.32	22.21	8.11	3.53
CAS-1000	99.48	97.89	94.77	81.25	56.84	25.33	10.32	5.29
CAS-full	99.48	97.91	94.86	81.47	58.33	28.41	13.22	7.98
SGBM	99.96	99.83	99.54	99.11	98.65	98.17	97.77	97.05
ELAS	83.25	83.00	82.39	81.06	79.82	79.48	79.36	79.23
IGEV++	99.83	99.31	98.52	95.91	90.06	78.71	73.47	70.50
MoCha	99.84	99.31	98.71	98.50	98.47	98.41	96.95	93.26
NeRF-RAFT	98.88	95.00	88.36	73.27	61.93	52.73	45.36	39.85
RAFT	99.34	97.25	93.56	82.80	71.98	66.75	64.43	63.45

Table 5.13: Evaluations of the chosen stereo methods with the percentage of bad depths metric on the “Close” subset. The values reported are the percentage of pixels above the specified depth error. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in yellow. The 2nd-best result for each column is highlighted in gray. The 3rd-best result for each column is highlighted in light brown.

Depth (m)	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40
CAS-200	25.61	12.01	2.77	2.09	0.82	0.46	0.42	0.41
CAS-600	30.42	15.08	5.74	3.34	1.59	0.85	0.77	0.72
CAS-1000	35.89	17.50	7.25	4.30	2.49	1.60	1.43	1.23
CAS-full	41.91	20.52	9.26	5.50	3.75	2.82	2.63	2.43
SGBM	99.46	99.37	99.27	99.20	99.13	99.05	98.85	98.56
ELAS	81.61	81.06	80.60	80.46	80.39	80.35	80.33	80.31
IGEV++	88.78	76.21	73.27	71.94	70.78	69.52	68.43	67.39
MoCha	98.80	98.55	98.41	96.06	91.72	90.03	87.10	82.36
NeRF-RAFT	53.09	33.88	14.56	6.38	3.65	2.54	2.16	1.95
RAFT	67.08	57.38	48.25	40.76	37.07	29.47	26.85	26.23

In terms of percentage of bad pixels over x depth error, Table 5.13 indicates that the CAS method at fixation radii of 200, 600, and 1000 outperform all other methods at all depth error thresholds. NeRF-RAFT has similar results as CAS-full. While the proposed method does not exhibit the lowest disparity error, it dominates in having the lowest depth error. Again, while disparity is widely used in stereo matching research, it is not entirely

representative of the resulting depth error, which is important in robotics applications.

Overall, the CAS convergent stereo method performs the best in the “Close” objects subset.

5.1.4 Smooth Planes Subset

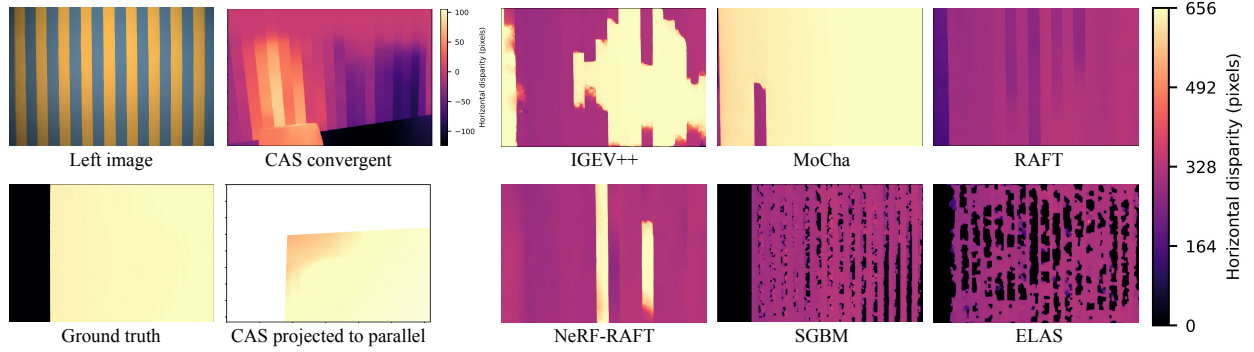


Figure 5.10: Example output disparity images of one of the scenes in the “Smooth Planes” subset. The “Left image” shows the left image of the parallel image pair. “Ground truth” is the ground truth horizontal disparity in the parallel configuration measured from structured light. “CAS convergent” shows the horizontal disparity output of one of the many fixations taken for the scene with a different color scale than the other disparity maps. “CAS projected to parallel” shows the results of transforming the convergent vertical and horizontal disparity to horizontal disparity in the parallel configuration to be able to compare to the ground truth measurements. The outputs of the state-of-the-art parallel methods are the parallel methods’ disparity outputs for this scene. The scale is shown with the color bar on the right in pixel units.

A set of images that contained one plane with very few variations in image features or many smooth planes was selected to form the “Smooth Planes” subset. Refer to Appendix A.5 for all the left parallel images of this subset. This subset was created to evaluate performance in the absence of many features and textures, a common area of difficulty in stereo matching algorithms. Table 5.14 shows the results of this subset with overall absolute depth error (m) and absolute disparity error (pixels). While CAS at 200, 600, and 1000 fixation radii achieve best results in mean and standard deviation depth error, IGEV++, MoCha, and NeRF-RAFT achieve best results in median depth error, while CAS-200 and CAS-600 come in second and third. In terms of disparity error, deep learning methods IGEV++, MoCha, and NeRF-RAFT perform the best in mean and median error. CAS-200 and CAS-600 have the top lowest disparity error standard deviations.

Figure 5.10 shows an example scene from this set and the outputs of disparity maps from the methods. The disparity map shown for the proposed convergent stereo method is the horizontal disparity map from one fixation out of the many fixations taken. MoCha often produced the best appearing disparity map given planes with few features. The other deep learning methods often would assign vastly different disparity values to certain patches in this set. NeRF-RAFT and IGEV++ both had some areas of their disparity map assigned correctly, while other parts were vastly incorrect. ELAS and SGBM disparity maps were incorrectly assigned due to the repeated patterns of this plane. The output of CAS tends to segregate disparity values based on the image information, as each strip tends to its own disparity value. The gradient expected from a plane did not have a smooth transition. However, the convergent stereo method generally does well in smooth planes by leveraging fixation and camera geometry information.

Table 5.14: Absolute depth errors (m) and absolute disparity errors (pixels) across the “Smooth Planes” objects subset reported in mean, median and STD values. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in yellow. The 2nd-best result for each column is highlighted in gray. The 3rd-best result for each column is highlighted in light brown.

	Depth error (m)			Disparity error (pixels)		
	Mean	Median	STD	Mean	Median	STD
CAS-200	0.09	0.06	0.09	132	113	120
CAS-600	0.10	0.07	0.10	133	102	131
CAS-1000	0.12	0.08	0.12	141	98	147
CAS-full	0.13	0.08	0.14	149	98	158
SGBM	2.33	1.27	1.94	492	498	220
ELAS	2.31	0.93	1.99	421	361	171
IGEV++	0.43	0.01	1.20	122	15	199
MoCha	0.19	0.01	0.74	104	8	169
NeRF-RAFT	0.14	0.01	0.26	98	14	141
RAFT	1.53	0.72	1.94	308	324	278

Table 5.15 shows the results of evaluating the stereo methods using the percentage of bad pixels over x disparity value metric. MoCha generally performed the best with this metric, while IGEV++ and NeRF-RAFT follow closely. For pixels at and over the 200 bad-pixel threshold, the CAS methods have generally better results than the other methods. This metric can be a bit misleading as IGEV++, NeRF-RAFT, and MoCha all have extremely

Table 5.15: Evaluations of the chosen stereo methods with the percentage of bad pixels metric on the “Smooth Planes” subset. The values reported are the percentage of pixels above the specified disparity error. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in yellow. The 2nd-best result for each column is highlighted in gray. The 3rd-best result for each column is highlighted in light brown.

Disparity (pixels)	1	4	10	40	100	200	300	400
CAS-200	99.76	99.05	97.73	84.18	60.08	16.10	3.68	3.54
CAS-600	99.57	98.27	95.97	81.44	53.00	17.44	5.95	4.48
CAS-1000	99.56	98.18	95.58	80.21	50.49	19.53	7.60	5.19
CAS-full	99.53	98.09	95.42	79.76	49.93	21.83	9.89	6.59
SGBM	99.84	99.31	98.62	97.09	96.25	90.63	85.24	63.07
ELAS	99.15	97.00	94.63	89.64	88.27	85.85	81.85	55.00
IGEV++	95.26	80.16	56.63	39.38	29.52	24.71	20.79	10.77
MoCha	94.89	76.57	46.03	38.52	37.62	17.31	13.64	7.98
NeRF-RAFT	95.27	81.05	56.54	38.43	30.47	25.13	20.28	2.35
RAFT	97.23	86.94	67.96	59.63	55.81	53.16	48.94	29.39

Table 5.16: Evaluations of the chosen stereo methods with the percentage of bad depths metric on the “Smooth Planes” subset. The values reported are the percentage of pixels above the specified depth error. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in yellow. The 2nd-best result for each column is highlighted in gray. The 3rd-best result for each column is highlighted in light brown.

Depth (m)	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40
CAS-200	58.91	28.47	15.27	7.25	7.05	6.45	2.85	2.01
CAS-600	61.06	32.91	20.22	11.59	9.13	6.39	3.56	1.39
CAS-1000	63.52	36.74	24.05	14.94	10.65	7.39	4.89	2.64
CAS-full	64.75	39.70	27.31	18.03	13.06	9.51	6.99	4.51
SGBM	97.79	97.12	96.75	95.27	93.66	92.55	91.59	90.06
ELAS	92.07	89.91	89.42	89.06	88.48	87.60	87.01	86.00
IGEV++	33.93	28.17	24.28	22.33	21.29	20.23	19.73	19.38
MoCha	38.59	30.10	16.64	11.05	8.30	6.55	5.98	5.71
NeRF-RAFT	34.01	27.77	22.31	18.06	17.15	16.42	15.56	15.16
RAFT	58.47	55.34	53.64	52.27	51.12	50.60	50.06	49.80

wrong disparity values at select parts of the image in Figure 5.10. The percentage of bad pixels above x disparity error metric does not give a good sense of how poorly assigned the other pixels are. Rather, it focuses only on the percentage of pixels that have an error over a certain threshold as deep learning methods sometimes have largely incorrect pixel patches. It is unsurprising that the proposed method has better performance than the deep parallel

methods past the 200-pixel threshold. While the CAS methods do not achieve the lowest percentage of bad pixels for lower pixel thresholds, it maintains that the majority of its pixels do not exceed large disparity errors. See Table 5.14, where CAS-200 has the lowest standard deviation of both depth and disparity error.

In terms of the percentage of pixels with over x depth error in Table 5.16, the CAS methods performs the best for depth error thresholds at and over 0.15m. MoCha trails closely to the performance of CAS-200. For CAS-600, only 1.39% of its pixels have a depth error of over 0.4m. IGEV++, MoCha, and NeRF-RAFT, however, have 19.30%, 5.71%, and 15.16% of their pixels over 0.4m, respectively. Even though these deep learning methods obtained many of the lowest disparity bad pixel results, when it comes to depth, they have a much higher amount of errors than the proposed convergent stereo method.

5.1.5 TEOS Objects Subset

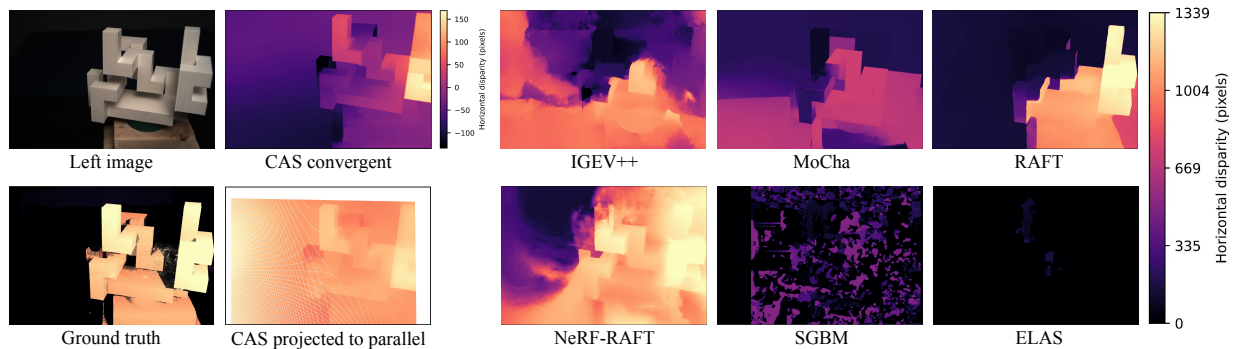


Figure 5.11: Example output disparity images of one of the scenes in the “TEOS Objects” subset. The “Left image” shows the left image of the parallel image pair. “Ground truth” is the ground truth horizontal disparity in the parallel configuration measured from structured light. “CAS convergent” shows the horizontal disparity output of one of the many fixations taken for the scene with a different color scale than the other disparity maps. “CAS projected to parallel” shows the results of transforming the convergent vertical and horizontal disparity to horizontal disparity in the parallel configuration to be able to compare to the ground truth measurements. The outputs of the state-of-the-art parallel methods are the parallel methods’ disparity outputs for this scene. The scale is shown with the color bar on the right in pixel units.

TEOS objects [4] were created to investigate the role of self-occlusion in the context of object classification. TEOS objects have three levels of object complexity, which increase in the dimension of the object as well as the amount of self-occlusion the object presents. These

shapes contain a base in which cuboids are attached to form a complex shape of the same color and texture, but include many points of self-occlusion. For the images of all of the scenes that included TEOS objects, please refer to Appendix A.8. These objects were used to test the performance of stereo matching methods in the presence of high self-occlusions in the absence of other typical image cues such as texture differences and color variation. These objects are non-existent in typical stereo datasets and would be a good measure of how well stereo methods generalize to unfamiliar objects and scenes.

Table 5.17: Absolute depth errors (m) and absolute disparity errors (pixels) across the “TEOS Objects” objects subset reported in mean, median and STD values. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in yellow. The 2nd-best result for each column is highlighted in gray. The 3rd-best result for each column is highlighted in light brown.

	Depth error (m)			Disparity error (pixels)		
	Mean	Median	STD	Mean	Median	STD
CAS-200	0.13	0.07	0.21	125	96	119
CAS-600	0.19	0.08	0.31	144	119	125
CAS-1000	0.23	0.09	0.36	150	119	134
CAS-full	0.23	0.09	0.38	153	119	141
SGBM	1.96	1.13	1.84	467	410	355
ELAS	1.47	0.45	1.89	366	172	418
IGEV++	0.54	0.14	0.94	265	122	330
MoCha	0.54	0.20	0.84	302	223	327
NeRF-RAFT	0.27	0.03	0.47	145	50	198
RAFT	0.61	0.20	0.93	234	80	320

The proposed convergent stereo method outperforms parallel methods in mean depth error, standard deviation of depth error, mean disparity error, and standard deviation of disparity error in Table 5.17. Only NeRF-RAFT is able to have a lower median depth error than any of the pixel radii to fixation categories of the CAS method. In Table 5.18, NeRF-RAFT maintains the best performance in the percentage of bad pixels disparity metric up to the 100 pixel threshold. Past this, CAS-200 takes the lead. A similar pattern occurs in Table 5.19, where NeRF-RAFT performs the best for smaller depth error thresholds, while CAS-200 performs the best for higher depth error thresholds. The deep learning methods tend to assign some areas correctly while having extremely large errors for other areas of the block. Visually, this can be seen in Figure 5.11, where many sections of the scene have

incorrect disparity values. Table 5.18 also reflects this observation as CAS-200 only has more than a 400 pixel disparity error in 1.18% of its pixels, while the next best parallel method has 11.31% of their pixels with a higher error than this. This is again reflected in the bad pixel depth metric in Table 5.19, where pixel radii to fixation categories of CAS have a low amount of pixels with a depth error larger than 0.40m relative to the parallel methods.

Table 5.18: Evaluations of the chosen stereo methods with the percentage of bad pixels metric on the “TEOS Objects” subset. The values reported are the percentage of pixels above the specified disparity error. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in yellow. The 2nd-best result for each column is highlighted in gray. The 3rd-best result for each column is highlighted in light brown.

Disparity (pixels)	1	4	10	40	100	200	300	400
CAS-200	99.50	97.82	93.78	71.86	46.30	18.10	4.43	1.18
CAS-600	99.51	97.99	94.78	79.26	55.33	24.87	8.53	2.59
CAS-1000	99.53	98.08	95.09	80.62	56.28	26.19	9.83	3.13
CAS-full	99.52	98.06	95.09	80.88	56.45	26.95	10.83	3.95
SGBM	99.74	99.02	97.96	93.57	86.80	71.16	62.87	55.50
ELAS	97.34	91.12	84.64	75.93	65.96	50.19	47.89	42.98
IGEV++	95.53	83.18	71.27	58.49	48.10	35.91	28.74	21.22
MoCha	95.14	82.06	70.25	57.96	53.83	48.71	42.08	30.56
NeRF-RAFT	94.41	81.13	65.64	45.65	31.19	23.01	17.19	11.31
RAFT	96.31	84.98	72.55	56.78	42.04	32.49	26.79	20.23

Table 5.19: Evaluations of the chosen stereo methods with the percentage of bad depths metric on the “TEOS Objects” subset. The values reported are the percentage of pixels above the specified depth error. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in yellow. The 2nd-best result for each column is highlighted in gray. The 3rd-best result for each column is highlighted in light brown.

Depth (m)	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40
CAS-200	55.88	37.66	25.51	19.63	12.14	7.63	5.40	3.87
CAS-600	60.07	43.75	32.29	25.48	19.96	15.27	12.13	9.91
CAS-1000	61.79	45.57	34.06	27.28	22.03	17.93	14.94	12.86
CAS-full	63.69	47.35	35.55	28.49	23.06	18.89	15.92	13.97
SGBM	96.86	95.05	92.27	86.46	83.44	81.63	80.02	78.70
ELAS	78.06	75.62	65.70	57.65	56.26	55.40	54.93	54.57
IGEV++	53.83	46.40	42.66	39.53	36.86	33.66	30.98	28.83
MoCha	62.70	58.46	54.85	46.92	36.29	33.58	30.38	28.49
NeRF-RAFT	34.40	28.24	25.42	23.66	22.63	21.52	20.90	20.07
RAFT	54.55	51.30	48.32	45.06	41.65	38.58	36.90	35.36

5.1.6 Repeated Patterns Subset

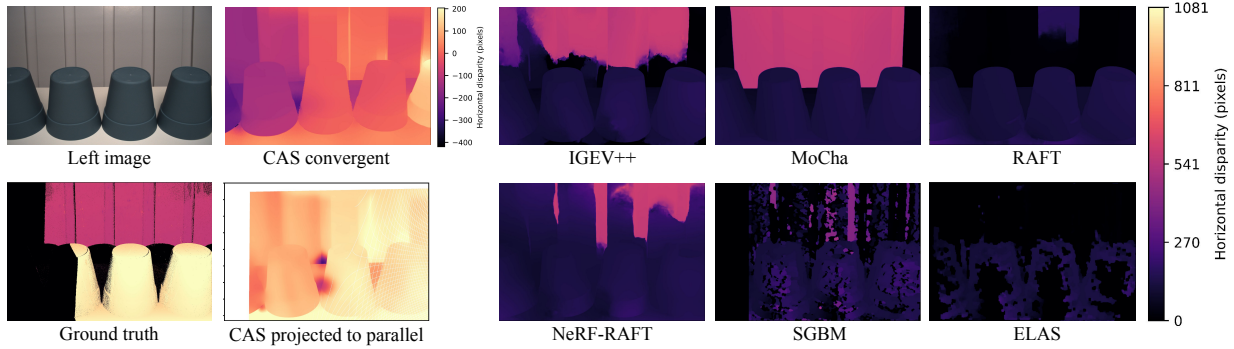


Figure 5.12: Example output disparity images of one of the scenes in the “Repeated Patterns” subset. The “Left image” shows the left image of the parallel image pair. “Ground truth” is the ground truth horizontal disparity in the parallel configuration measured from structured light. “CAS convergent” shows the horizontal disparity output of one of the many fixations taken for the scene with a different color scale than the other disparity maps. “CAS projected to parallel” shows the results of transforming the convergent vertical and horizontal disparity to horizontal disparity in the parallel configuration to be able to compare to the ground truth measurements. The outputs of the state-of-the-art parallel methods are the parallel methods’ disparity outputs for this scene. The scale is shown with the color bar on the right in pixel units.

The “Repeated Patterns” subset was created to test the stereo methods under situations where objects are repeated and corresponding objects are ambiguous. Although these scenarios may not appear often in real-world settings, it is still important to test edge cases and limitations of methods under various conditions, not just typical and simple conditions. These scenes included repeated patterns on planes and repeated patterns in objects that were created to be close to identical. Please refer to Appendix A.5 for all the left parallel images of this subset.

Here is where an active convergent stereo system excels where others fail. As can be seen in Table 5.20, the CAS methods’ mean depth error is almost 1m less than the next best method, and the mean disparity error is around 200 pixels more accurate than the next best method. All fixation radii categories of CAS maintain the lowest standard deviation of error in both metrics as well. As in the example scene in Figure 5.12, IGEV++, MoCha, and NeRF-RAFT assigned the board in the back to be very close to the camera while the cups are at a farther distance. ELAS and SGBM also struggled quite a bit in assigning disparity values, as the disparity values assigned indicated that the objects in the scene were at a very

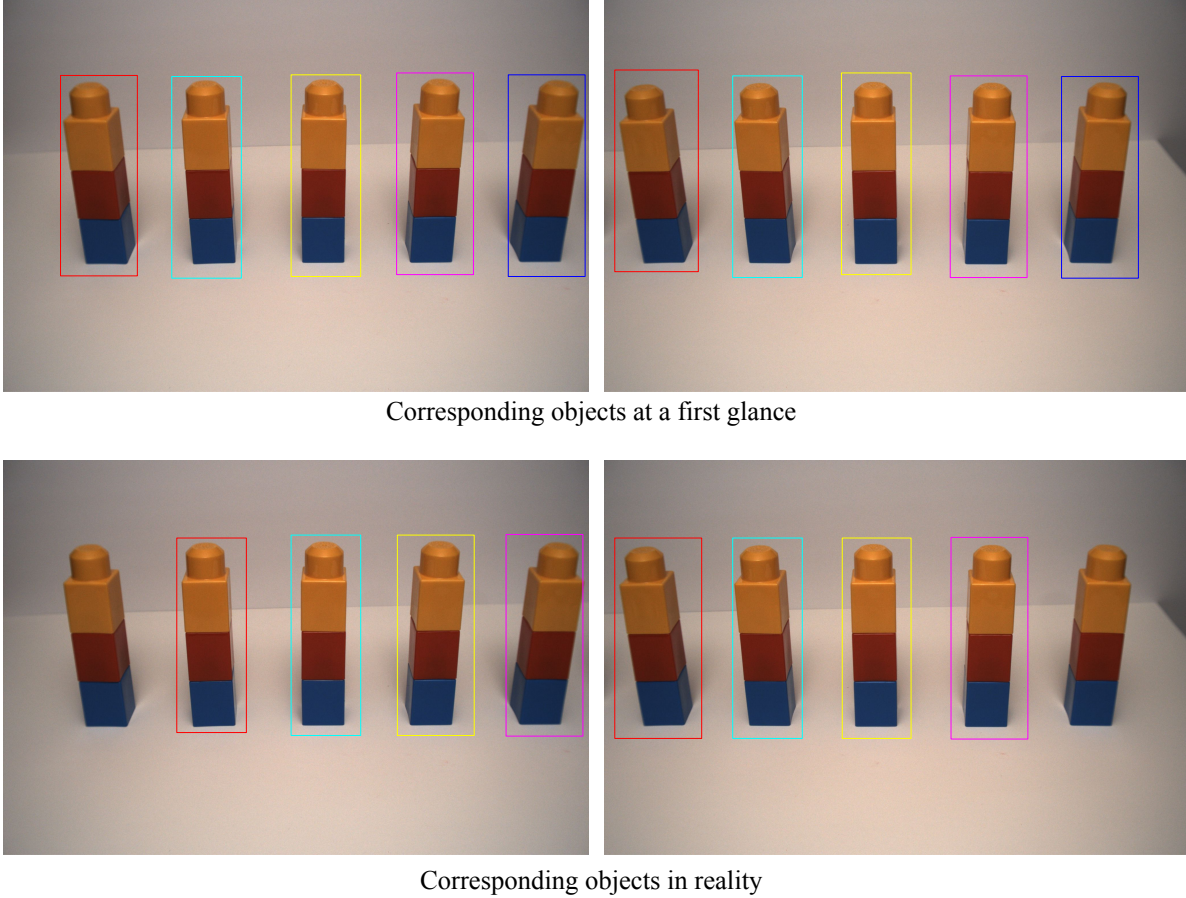


Figure 5.13: A demonstration of the difficulty of determining correspondence in the “Repeated patterns” subset. At first glance, one may assume the correspondences are as with the top set of images. The corresponding colors on the boxes around the block towers in the left and right images represent assumed corresponding objects. However, the top assignment is incorrect. The true correspondence is shown in the bottom pair of images, where it is revealed that there are 6 towers in the scene.

large distance away, essentially matching the wrong objects as corresponding.

In terms of bad disparity pixels in Table A.7, deep learning models do the best for lower disparity thresholds. However, this does not reveal much about the performance of their methods. If 66.31% of a method’s pixels have a higher than 40 pixels disparity error, it is not informative of how well the other 66.31% pixels are assigned. Starting at the 100 pixels threshold, CAS-200, CAS-600, CAS-1000, and CAS-full all outperform all other methods, often having around a 20% lead compared to other methods. In terms of bad depth pixels in Table 5.22, all top scores belong to one of the radii to fixation categories of CAS. Only 4.58% of pixels from CAS-full have a depth error larger than 0.40m, while the next best

Table 5.20: Absolute depth errors (m) and absolute disparity errors (pixels) across the “Repeated Patterns” subset reported in mean, median, and STD values. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in yellow. The 2nd-best result for each column is highlighted in gray. The 3rd-best result for each column is highlighted in light brown.

	Depth error (m)			Disparity error (pixels)		
	Mean	Median	STD	Mean	Median	STD
CAS-200	0.13	0.07	0.20	121	94	107
CAS-600	0.13	0.08	0.17	143	105	131
CAS-1000	0.13	0.09	0.17	155	110	144
CAS-full	0.14	0.09	0.16	162	113	148
SGBM	2.59	2.20	1.71	596	578	224
ELAS	2.81	2.69	1.69	626	592	216
IGEV++	1.21	0.60	1.49	370	323	322
MoCha	1.09	0.37	1.37	352	281	333
NeRF-RAFT	1.06	0.81	1.19	374	341	312
RAFT	2.55	2.08	1.63	565	578	230

Table 5.21: Evaluations of the chosen stereo methods with the percentage of bad pixels metric on the “Repeated Patterns” subset. The values reported are the percentage of pixels above the specified disparity error. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in yellow. The 2nd-best result for each column is highlighted in gray. The 3rd-best result for each column is highlighted in light brown.

Disparity (pixels)	1	4	10	40	100	200	300	400
CAS-200	99.30	97.35	93.04	79.09	43.18	16.20	7.49	3.98
CAS-600	99.34	97.40	93.82	79.27	50.32	22.11	11.32	5.97
CAS-1000	99.40	97.60	94.29	79.55	51.59	24.79	13.51	7.96
CAS-full	99.45	97.80	94.76	80.24	53.17	27.72	16.05	9.77
SGBM	100.00	100.00	100.00	99.99	99.73	96.30	93.09	78.41
ELAS	100.00	100.00	100.00	99.99	99.98	99.95	99.72	89.32
IGEV++	97.97	90.93	81.79	71.43	65.60	59.73	49.65	41.30
MoCha	97.68	88.63	74.81	66.31	62.13	47.22	44.35	41.53
NeRF-RAFT	98.19	92.62	84.95	75.86	69.31	63.01	56.43	35.81
RAFT	99.58	97.88	95.60	94.06	92.88	91.67	90.34	79.71

parallel method has 43.23% of their pixels with a higher depth error.

Figure 5.13 demonstrates the difficulty with matching repeated objects and patterns in these scenes. The top image is what one could assume the correspondences for this scene are at a first glance. It would make sense that the first object in the left image would

Table 5.22: Evaluations of the chosen stereo methods with the percentage of bad depths metric on the “Repeated Patterns” subset. The values reported are the percentage of pixels above the specified depth error. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in yellow. The 2nd-best result for each column is highlighted in gray. The 3rd-best result for each column is highlighted in light brown.

Depth (m)	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40
CAS-200	58.68	37.79	23.65	15.53	13.99	11.10	9.18	7.90
CAS-600	62.72	39.31	25.09	17.18	13.03	8.51	5.92	4.79
CAS-1000	65.37	42.44	28.25	19.39	13.97	8.82	5.80	4.60
CAS-full	67.12	45.46	31.38	21.60	15.21	9.81	5.94	4.58
SGBM	100.00	99.99	99.77	98.94	97.96	96.98	95.90	94.98
ELAS	100.00	100.00	100.00	100.00	100.00	99.99	99.99	99.92
IGEV++	68.17	64.30	61.84	58.68	55.59	54.58	53.52	52.10
MoCha	65.69	58.16	47.31	45.72	44.83	44.10	43.62	43.23
NeRF-RAFT	72.04	66.98	62.45	58.58	56.81	55.25	54.54	54.21
RAFT	93.24	92.23	91.66	90.97	90.39	90.26	90.22	90.20

correspond with the first object in the right image, which is just slightly to the left of it in the image plane, satisfying the horizontal disparity direction expectation. However, the actual corresponding objects are shown in the bottom set of left and right images where it is revealed that there are 6 towers in the scene and that the first tower to the left in the left image and the last tower on the right in the right image do not have corresponding towers in both cameras’ fields of view. Because of this confusion and lack of other image cues, many of the tested methods would assign disparities following the upper incorrect assignment. This large error in disparity can result in a very large depth estimation error, placing objects much farther away than they actually are. However, with a convergent stereo camera system that can fixate on specific individual objects, this ambiguity can be solved. The proposed convergent stereo method can leverage camera orientation information as well as potential new information obtained from looking at different parts of the scene to accurately assign disparity values.

5.1.7 Dark Subset

The “Dark” subset of scenes was obtained by lowering the luminance of the LED light panels for a few scenes in the dataset. Traditional methods sometimes use pixel intensity

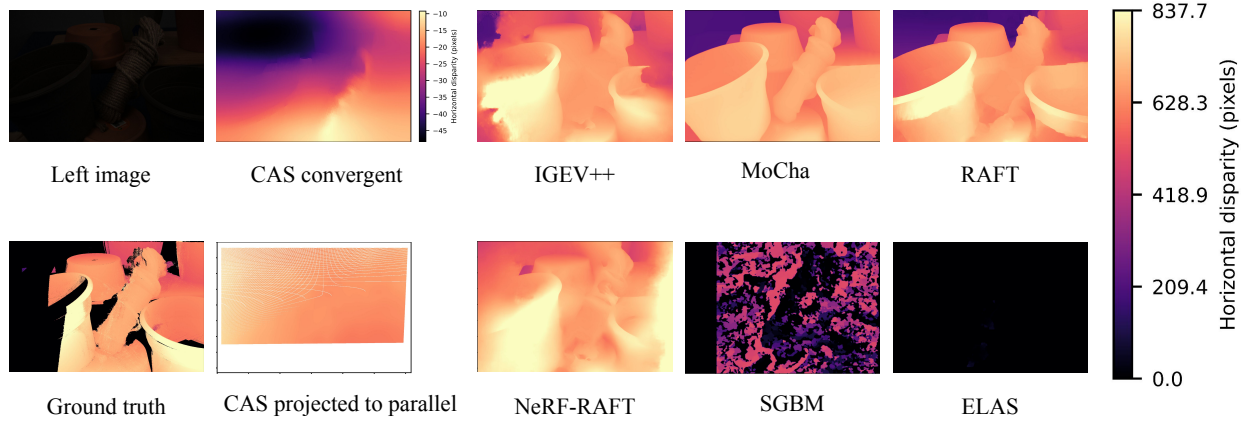


Figure 5.14: Example output disparity images of one of the scenes in the “Dark” subset. The “Left image” shows the left image of the parallel image pair. “Ground truth” is the ground truth horizontal disparity in the parallel configuration measured from structured light. “CAS convergent” shows the horizontal disparity output of one of the many fixations taken for the scene with a different color scale than the other disparity maps. “CAS projected to parallel” shows the results of transforming the convergent vertical and horizontal disparity to horizontal disparity in the parallel configuration to be able to compare to the ground truth measurements. The outputs of the state-of-the-art parallel methods are the parallel methods’ disparity outputs for this scene. The scale is shown with the color bar on the right in pixel units.

differences to calculate disparity, and as such, images with lower contrast in intensity levels may pose issues to the matching algorithm. This subset’s purpose is to evaluate the proposed method against other traditional methods and see how well deep learning methods perform in a poorly illuminated scene. CAS utilizes pixel intensities similar to traditional methods however, the use of Gabor filter responses encodes the intensities of surrounding pixels indicating a mean illumination change as opposed to comparing pixel-wise intensities.

Although visually the CAS methods do not appear to have good disparity maps, the error plots in Tables 5.23, and 5.25 indicate that CAS had the lowest depth error in this set. Deep learning methods had the lowest errors in smaller thresholds for the bad pixel disparity metric in Table 5.24, but CAS at all pixel radii to fixation performs best at and past the 100 pixel threshold. The large amount of error that the parallel stereo method had from the “Repeated patterns” subset and the “Close” subset may have influenced their average results in the “Dark” set. The convergent stereo method does not do too well in distinguishing edges and features in dark images due to relying on the difference in amplitude in the Gabor filter responses for matching. It can be seen in the example scene shown in Figure 5.14 that CAS

Table 5.23: Absolute depth errors (m) and absolute disparity errors (pixels) across the “Dark” subset reported in mean, median, and STD values. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in yellow. The 2nd-best result for each column is highlighted in gray. The 3rd-best result for each column is highlighted in light brown.

	Depth error (m)			Disparity error (pixels)		
	Mean	Median	STD	Mean	Median	STD
CAS-200	0.12	0.08	0.12	115	92	98
CAS-600	0.13	0.09	0.14	133	100	123
CAS-1000	0.14	0.10	0.16	144	103	136
CAS-full	0.14	0.10	0.16	151	106	141
SGBM	2.41	1.96	1.79	562	568	255
ELAS	3.49	3.59	1.45	749	780	188
IGEV++	0.98	0.27	1.29	327	250	315
MoCha	1.05	0.17	1.41	327	172	331
NeRF-RAFT	0.96	0.21	1.30	323	205	323
RAFT	1.78	1.48	1.73	430	497	321

did not do well in this scene however it could be argued that the disparity estimates were in the general vicinity of the ground truth disparity since the convergent system can leverage the knowledge of the disparity at the fixation point. ELAS and SGBM also did not perform too well, with SGBM doing better than ELAS, which often had most of its disparity outputs unassigned. NeRF-RAFT also seems to have generated clouds of incorrect disparities in the dark sets, possibly indicating that their training set, collected with handheld cameras, did not have many dark images.

In an active camera system, darkness can be a cue for the cameras to adjust camera parameters accordingly to better view the scene. Parameters which would affect exposure value can be shutter speed and aperture size. In CODY, the cameras have the ability to change these parameters.

Table 5.24: Evaluations of the chosen stereo methods with the percentage of bad pixels metric on the “Dark” subset. The values reported are the percentage of pixels above the specified disparity error. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in gold. The 2nd best result for each column is highlighted in silver. The 3rd best result for each column is highlighted in bronze.

Disparity (pixels)	1	4	10	40	100	200	300	400
CAS-200	99.13	96.75	91.76	78.82	41.40	15.87	5.49	2.38
CAS-600	99.30	97.28	93.51	77.85	49.10	20.02	9.29	5.27
CAS-1000	99.33	97.36	93.71	77.64	49.85	21.37	11.46	7.21
CAS-full	99.36	97.48	93.98	78.26	51.00	23.85	13.72	8.96
SGBM	99.98	99.93	99.84	99.39	97.64	88.18	80.18	72.07
ELAS	100.00	100.00	100.00	99.99	99.97	98.31	98.14	94.70
IGEV++	98.93	95.82	90.03	72.81	61.40	52.94	42.17	36.24
MoCha	97.13	90.13	81.32	67.77	58.85	47.78	43.20	37.35
NeRF-RAFT	98.84	95.10	88.33	71.43	57.95	46.23	39.74	29.91
RAFT	99.17	96.43	91.38	80.94	70.24	65.79	63.84	60.55

Table 5.25: Evaluations of the chosen stereo methods with the percentage of bad depths metric on the “Dark” subset. The values reported are the percentage of pixels above the specified depth error. CAS-200, CAS-600, and CAS-1000 indicate the pixel radius to the fixation point that was included in the evaluation. CAS-full evaluated the error with respect to all pixels in the image. The best result for each column is highlighted in yellow. The 2nd-best result for each column is highlighted in gray. The 3rd-best result for each column is highlighted in light brown.

Depth (m)	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40
CAS-200	61.77	44.76	31.01	18.67	11.06	6.03	3.99	3.17
CAS-600	66.33	47.28	30.75	20.13	12.64	6.78	3.93	3.12
CAS-1000	67.92	48.43	32.56	21.73	14.19	8.02	4.32	3.35
CAS-full	69.19	49.67	34.55	23.23	15.58	9.32	4.67	3.55
SGBM	99.62	99.27	98.35	96.03	92.43	89.23	86.69	84.67
ELAS	100.00	100.00	100.00	100.00	100.00	99.99	99.99	99.94
IGEV++	69.99	61.03	55.97	51.09	47.74	46.12	45.03	43.64
MoCha	65.83	57.01	48.84	44.77	41.82	41.20	40.87	40.64
NeRF-RAFT	63.98	55.54	50.06	44.97	42.50	39.97	39.28	38.99
RAFT	78.94	70.91	67.55	65.99	64.60	63.55	63.03	62.57

5.1.8 Summary

This section of the analysis aims to answer **RQ1**: “How do the resulting depth maps of fixed parallel stereo compare to convergent stereo with multiple fixation points in terms of accuracy and map fidelity in various regions?” and **RQ2**: “Is there an advantage to having the ability to fixate on different areas in the scene for depth perception? If so, in which scenarios are the added degrees of freedom particularly useful?”. The average depth and disparity errors were analysed as well as the percentage of bad pixels over x disparity threshold and the percentage of bad pixels over x depth threshold. These metrics were evaluated for all scenes in the dataset as well as various subsets. The CAS algorithm, which takes advantage of motor calibration that provides a known geometry of the cameras, demonstrates the lowest mean depth error, mean disparity error, and standard deviation across all scenes. The CAS algorithm performs well in the other subsets, apart from the known limitation of “Far” targets, by achieving comparably low error. However, its main advantage was shown in the “Repeated Patterns” subset.

RQ1. Overall, the resulting depth and disparity maps obtained from a convergent stereo system with multiple fixation points are comparable and even outperform all deep learning stereo methods aggregated across all scenes in mean and standard deviation of disparity and depth error as well as larger thresholds for depth and disparity for percentage of bad pixel metrics. It consistently can outperform the parallel state-of-the-art methods, IGEV++, MoCha, and RAFT, in many metrics, especially depth error in all subsets, apart from the “Normal” and “Far” subsets. While some deep learning methods performed better in some subsets, such as MoCha in “Smooth Planes”, they did not consistently perform better than the CAS convergent method. The proposed convergent stereo method can compete with NeRF-RAFT in many subsets and obtains similar, if not slightly better, performance across the board. However, in the “Repeated pattern” subset, the convergent method can differentiate itself from other stereo methods with significantly lower error than other methods. Analysing the outputs of each method’s disparity maps, deep learning methods tend to assign clouds of extremely incorrect disparities. This causes the deep methods’ large standard deviation of error across all scenes. CAS’s standard deviation error range is lower, indicating

a more contained and reliable stereo method.

The CAS methods were able to consistently perform close to, if not the best, for mean depth error, and standard deviation of depth error for all subsets except for the “Far” subset. The CAS methods were able to achieve best results for percentage of bad pixels over x disparity error for larger disparities for all subsets except for the “Far” subset. In terms of the percentage of bad pixels over x depth error measurements, the CAS methods were able to have close to, if not the best, performance for all depth thresholds except for smaller thresholds for all subsets except for “Far” and “Normal”.

Analysing the performance when fixations are separated into different regions across the parallel left image, the upper right corner had the highest amount of error. This is speculated to be caused by being able to view a larger range of depths of the scene in these fixation points. Visually comparing other depth and disparity maps with a higher amount of error reveals that the commonality between some of these sets is that the points with higher error often are not at a similar depth as the point of fixation. Fixating on an extremely close object can cause the objects in the distance to have an extremely large disparity value, and likewise, fixating on a far object would cause a larger disparity for closer objects.

RQ2. The proposed convergent stereo system demonstrates significant advantages in challenging scenarios involving close objects, smooth surfaces, highly self-occluded novel objects, and repeated patterns. The ability to actively fixate and utilize camera geometry provides a clear advantage when it comes to scenes and images that lack a rich environment of visual cues. It can use vergence cues to overcome ambiguities that parallel traditional and deep learning methods are not able to overcome.

5.2 Rectifying vs Not Rectifying

It is often an assumption in the stereo community that rectifying images is a necessary step in stereo computation, especially in stereo configurations where the cameras are not parallel. It allows the use of any well-developed parallel stereo matching method and makes it so that the stereo matching method deployed correctly assumes parallel epipolar lines along with its simplifications. Previously, other robotic binocular camera systems would rectify their convergent images to have parallel epipolar lines, then apply a parallel stereo algorithm, which can handle disparities in both horizontal directions to obtain their disparity map [95, 40]. iCub used SGBM and ELAS while Wang and Jonker used SGBM for their disparity maps. However, neither mentions any possible negative effects on image information due to image information compression and expansion from rectification. There may be adverse effects when parallel rectification of images is assumed to produce the same results as disparity calculations using the original image. There have been very few works that evaluate the consequences of rectifying images in terms of pixel matching error and depth estimation error using natural images. This section aims to answer **RQ3**. “What are the consequences of rectifying convergent images to be parallel, and is it better for depth accuracy to perform disparity calculations using original convergent images?”.

To test the consequences of parallel rectification, an experiment was created where, for each fixation, the image pair would be rectified so that the new images have parallel epipolar lines as well as new projection matrices $P_{l-rectified}$ and $P_{r-rectified}$. It was empirically found that using only motor values to rectify the images gave better results for rectification, as can be seen in Appendix C.1, whereas using the fundamental matrix obtained through SIFT features could result in largely warped images that changed the geometry of the objects in the scene significantly.

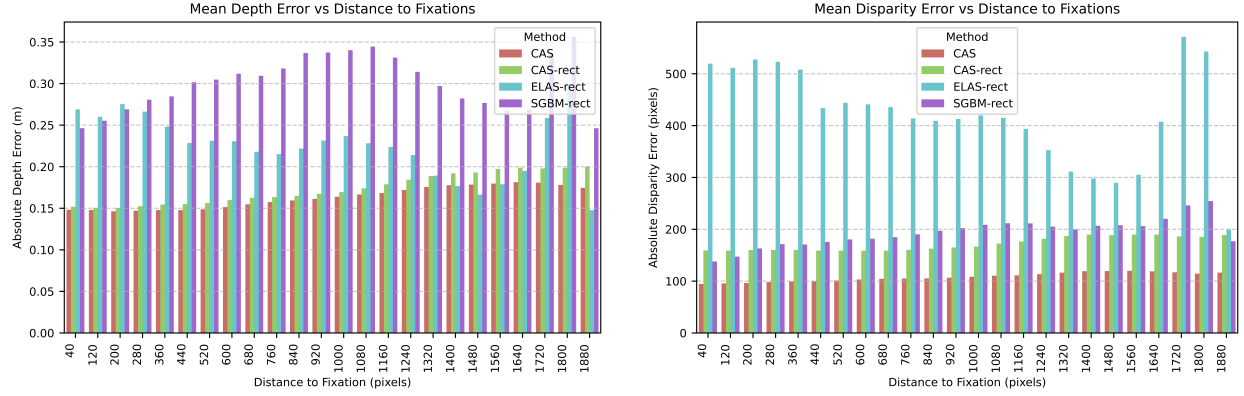
Then, ELAS, SGBM, and the parallel version of the proposed algorithm, which only searches in the horizontal direction for every pixel, were used to compute disparity maps for each method. The error evaluation was then made by using the same method described in Sections 4.3.2.2 and 5.1.

Table 5.26 shows a comparison of errors between the convergent method (CAS) in the first

Table 5.26: Absolute depth errors (m) and absolute disparity errors (pixels) across all scenes, including outputs of CAS, ELAS, and SGBM on rectified convergent images reported in mean, median, and STD values. “x-200”, “x-600”, and “x-1000” indicate the pixel radius to the fixation point that was included in the evaluation. “x-full” evaluated the error with respect to all pixels in the image. CAS-rect-x, SGBM-x, and ELAS-x, entries are the methods named applied to convergent images, which were rectified to have parallel epipolar lines.

	Depth error (m)			Disparity error (pixels)		
	Mean	Median	STD	Mean	Median	STD
CAS-200	0.130	0.070	0.200	122	96	115
CAS-600	0.150	0.080	0.220	138	106	132
CAS-1000	0.160	0.090	0.240	150	110	145
CAS-full	0.170	0.100	0.250	158	113	154
CAS-rect-200	0.150	0.090	0.170	203	160	181
CAS-rect-600	0.160	0.090	0.180	208	159	188
CAS-rect-1000	0.160	0.100	0.200	214	161	195
CAS-rect-full	0.170	0.100	0.220	227	172	207
SGBM-200	0.260	0.090	0.480	154	113	149
SGBM-400	0.290	0.120	0.500	173	125	164
SGBM-600	0.310	0.130	0.510	186	130	180
SGBM-full	0.310	0.140	0.510	192	132	187
ELAS-200	0.270	0.240	0.130	521	552	196
ELAS-600	0.240	0.230	0.160	476	483	236
ELAS-1000	0.230	0.210	0.180	442	444	253
ELAS-full	0.220	0.190	0.180	407	393	253
SGBM	2.040	1.160	1.810	527	502	349
ELAS	1.900	1.120	1.780	461	464	221
IGEV++	0.640	0.060	1.230	211	54	290
MoCha	0.530	0.050	1.090	201	16	306
NeRF-RAFT	0.440	0.050	0.760	191	52	244
RAFT	1.400	0.720	1.740	367	383	269

section of the table, the CAS convergent method using rectified parallel images (CAS-rect) in the second section of the table, SGBM using rectified parallel images (SGBM-x) in the third section of the table, and ELAS using rectified parallel images (ELAS-x) in the fourth section of the table along with the parallel methods applied on the parallel images in the last section of the table. The difference in depth error between CAS using original convergent images compared with CAS-rect using rectified parallel images was not significant, with CAS performing slightly better in terms of average depth error and noticeably better in terms of average disparity error. It was expected that the error in CAS-rect would increase as the distance to the center of the image increased. However, the performance did not significantly



(a) Mean absolute depth error as a function of distance to fixation

(b) Mean absolute disparity error as a function of distance to fixation

Figure 5.15: Results of the rectified methods compared to CAS convergent non-rectified method. The results are displayed with absolute depth and disparity error as a function of pixel distance from the fixation point. These methods all used fixated images.

differ in the different pixel distances to the fixation point categories. To visualize this effect more effectively, Figure 5.15 was created, where the depth error (m) and disparity error (pixels) were displayed with respect to the bins of pixel distance to the fixation point. It is similar to the process described in Figure 5.2 except each bin only collects pixels from the previous bin boundary to its current bin boundary, essentially collecting donut rings of error. This is different than in the process described in Figure 5.2, where each category included all pixels that fall within a certain radius. The least affected area of image rectification in a convergent pair of images fixated at the same point is often the center of the image. The image contents near the center of the image do not change significantly to make the epipolar lines parallel. However, parts of the image that are farther away from the center would experience rectification effects more. While some parts of the image need to be expanded, other parts need to be compressed, resulting in information loss and interpolation to warp the image to fit the horizontal epipolar line constraint. It can be assumed that the deviation in error between the two convergent stereo methods CAS and CAS-rect, at points farther from the center of the image is a combination of rectification effects in information loss and possible errors in rectification itself from fundamental matrix estimation noise. The difference in disparity error seems to be consistently lower in CAS using the original convergent images. This result indicates that while the proposed convergent stereo method

using original convergent images produces less disparity error, the effect on depth estimations near the point of fixation was not as significant. The disparity errors in the rectified parallel outputs may stem from slight errors in rectification, which would result in overlay errors when transforming disparity maps to the parallel configuration to compare against ground truth.

In terms of ELAS and SGBM, they do not perform as well as the proposed stereo method in fixated images in terms of depth error. In terms of disparity error, SGBM performs better than CAS-rect but not better than the original convergent CAS method. SGBM and ELAS do not take into consideration that the disparity at the center of the image is 0. Thus, they are searching for a wider range of possible disparities at each pixel. For depth error in Table 5.26, the four methods with the lowest mean depth error include CAS-x, CAS-rect-x, SGBM-x, and ELAS-x. It is interesting to see that results that use fixated and convergent images, regardless of rectification, perform better than methods using parallel camera images in terms of depth error, even outperforming deep learning models, suggesting that there is an advantage to convergent images over parallel images.

Looking at the percentage of bad pixels above x disparity error metric displayed in Table 5.27, deep learning methods using parallel images continue to perform the best up until the 200 pixel threshold. There and onward, CAS-200 using non-rectified convergent images performs the best. In terms of bad pixels above x depth error, Table 5.28 reveals that CAS at 200, 600, and 1000 maintains the top 3 performance in most depth thresholds. ELAS-200 outperforms CAS at 0.40m, and NeRF-RAFT outperforms CAS at 0.05m and 0.10m.

As can be seen in both depth error and disparity error, the parallel methods (SGBM and ELAS), used on rectified convergent images, perform significantly better than the same methods on parallel images. This suggests that there is an advantage to viewing scenes in convergent images. The advantage may come from placing corresponding points closer in image space, reducing possible ambiguities in matches, and viewing the scene from a different angle, which provides more image and scene information.

To summarize, and to answer **RQ3**, rectification of convergent images to have parallel epipolar lines appears to slightly negatively affect the accuracy of depth and disparity. How-

Table 5.27: Evaluations of the chosen stereo methods with the percentage of bad pixels disparity metric across all scenes, including outputs of CAS, ELAS, and SGBM on rectified convergent images. The values reported are the percentage of pixels above the specified depth error. “x-200”, “x-600”, and “x-1000” indicate the pixel radius to the fixation point that was included in the evaluation. “x-full” evaluated the error with respect to all pixels in the image. CAS-rect-x, SGBM-x, and ELAS-x, entries are the methods named applied to convergent images, which were rectified to have parallel epipolar lines.

Disparity (pixels)	1	4	10	40	100	200	300	400
CAS-200	99.41	97.67	93.64	78.03	46.06	15.62	5.28	2.70
CAS-600	99.44	97.78	94.48	79.12	51.35	20.48	8.72	4.32
CAS-1000	99.47	97.89	94.78	79.70	52.57	22.99	10.82	5.79
CAS-full	99.49	97.95	94.98	80.16	53.66	25.32	13.01	7.43
CAS-rect-200	99.66	98.64	96.60	86.53	66.93	38.99	20.59	11.88
CAS-rect-600	99.66	98.66	96.65	86.74	67.23	39.40	21.50	12.71
CAS-rect-1000	99.66	98.66	96.65	86.70	67.20	40.05	22.63	13.64
CAS-rect-full	99.68	98.72	96.80	87.21	68.15	42.19	24.63	15.08
SGBM-200	99.54	98.14	95.25	79.29	52.69	24.63	10.40	4.85
SGBM-600	99.54	98.15	95.38	81.88	57.46	31.05	15.34	7.92
SGBM-1000	99.56	98.21	95.52	82.53	58.62	32.91	17.70	9.79
SGBM-full	99.58	98.30	95.73	83.12	59.62	34.11	19.47	11.28
ELAS-200	99.84	99.48	99.27	99.10	97.95	93.57	92.14	79.44
ELAS-600	99.78	99.24	98.38	96.71	93.46	88.32	83.01	68.67
ELAS-1000	99.74	98.99	97.74	94.37	89.24	82.69	76.95	61.74
ELAS-full	99.77	99.08	97.89	94.17	87.62	77.86	69.82	52.55
SGBM	99.63	98.56	96.69	93.39	89.97	81.74	75.40	65.83
ELAS	95.63	92.01	86.38	76.70	71.13	66.13	63.95	57.06
IGEV++	96.99	87.96	73.57	56.82	47.71	39.84	33.52	28.78
MoCha	95.95	84.14	67.15	56.78	52.23	44.13	40.12	34.40
NeRF-RAFT	96.55	86.70	70.83	52.17	41.79	33.94	28.68	19.98
RAFT	97.52	89.81	76.49	62.36	54.22	49.95	46.90	41.20

ever, just being able to fixate on different parts of a scene seems to provide a significant advantage over using only one pair of images taken in a parallel camera configuration. The benefits of convergence are seen to improve results significantly. Fixating on different objects in a scene can bring corresponding points closer in image space, resolve ambiguities in the scene, and provide more image information by obtaining a different view of the scene.

Table 5.28: Evaluations of the chosen stereo methods with the percentage of bad depths metric across all scenes, including outputs of CAS, ELAS, and SGBM on rectified convergent images. The values reported are the percentage of pixels above the specified depth error. “x-200”, “x-600”, and “x-1000” indicate the pixel radius to the fixation point that was included in the evaluation. “x-full” evaluated the error with respect to all pixels in the image. CAS-rect-x, SGBM-x, and ELAS-x, entries are the methods named applied to convergent images, which were rectified to have parallel epipolar lines.

Depth (m)	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40
CAS-200	58.39	38.77	25.23	16.52	11.87	8.99	6.96	5.84
CAS-600	63.44	43.50	29.60	20.45	14.51	10.21	7.60	6.06
CAS-1000	66.13	46.03	32.14	22.64	16.18	11.58	8.68	7.03
CAS-full	68.26	48.23	34.33	24.38	17.64	12.83	9.50	7.73
CAS-rect-200	69.28	46.71	33.34	25.62	19.63	15.14	10.97	8.19
CAS-rect-600	70.01	47.55	33.69	25.71	19.28	14.87	11.06	8.37
CAS-rect-1000	70.35	48.05	34.33	26.31	19.84	15.31	11.43	8.58
CAS-rect-full	71.48	49.71	36.23	27.66	20.92	16.14	12.18	9.23
SGBM-200	62.61	45.50	32.93	27.31	23.90	20.53	17.25	15.66
SGBM-600	70.09	53.03	40.96	33.44	28.20	23.60	19.90	17.56
SGBM-1000	73.20	56.43	44.37	36.27	30.28	25.37	21.35	18.67
SGBM-full	74.81	58.19	46.12	37.81	31.44	26.30	22.04	19.06
ELAS-200	97.39	93.06	89.00	67.41	47.62	41.51	27.64	3.85
ELAS-600	92.30	85.32	78.60	61.24	41.83	36.26	25.53	8.64
ELAS-1000	88.59	79.88	72.19	56.29	38.05	33.33	22.30	9.11
ELAS-full	86.40	75.81	66.14	51.62	34.82	30.21	20.45	9.85
SGBM	95.16	93.33	91.80	89.32	86.93	85.02	98.38	97.03
ELAS	81.32	76.24	72.83	70.59	69.51	68.75	91.69	87.48
IGEV++	54.77	45.57	41.45	38.74	36.79	35.46	81.79	71.55
MoCha	57.77	51.54	45.34	40.38	36.62	35.06	78.08	67.03
NeRF-RAFT	46.26	36.60	29.23	24.73	22.82	21.55	78.57	66.41
RAFT	60.74	53.88	50.28	47.70	46.08	43.97	83.57	74.42

5.3 Depth Ordering

Depth ordering can be a good way to analyse how well a method is able to determine relative depth between objects. To evaluate each method’s ability to have correct depth ordering, each fixation was paired with all other fixations in the scene. Then, the ground truth depth at each fixation was used to create an order of fixations from closest to farthest away. To determine the depth estimated at each fixation for the proposed convergent stereo method, the averaged depth value estimated for the center of the image was taken within a 10-pixel radius. For parallel methods, the depth estimates were averaged within a 10-pixel

radius at the fixation points. Each fixation was paired with another to determine a True False boolean for the pair. If the ordering of the pair is correct, ie, fixation 1 is closer to fixation 2 in the ground truth and the method, then the depth order is True for this particular fixation pair. There were 1366 pairs of depth orders that were evaluated for the full set. The numbers reported in Table 5.29 are the percentage of correct depth orders.

Table 5.29: Results of the percentage of fixation pairs with the correct depth ordering in each set of scenes. A higher percentage is better. The top percentage is highlighted in yellow gold, and the second-best percentage is highlighted in silver gray.

Set	CAS-full	SGBM	ELAS	IGEV++	MoCha	NeRF-RAFT	RAFT
All	68.4	50.4	53.3	67.9	64.6	69.9	74.9
Normal	66.0	53.1	59.7	73.6	72.6	72.9	74.9
Far	64.51	48.4	70.9	64.5	67.7	38.7	67.7
Close	80.7	46.8	50.5	50.5	32.1	66.9	73.4
Smooth Plane	83.3	19.0	4.7	76.2	83.3	90.5	61.9
TEOS	80.8	42.5	45.9	63.0	54.1	80.1	78.1
Repeated Pat...	63.0	59.4	59.4	60.9	55.7	53.1	75.5
Dark	57.7	52.7	60.4	66.7	65.3	60.3	74.3

The results in Table 5.29 indicate that RAFT’s disparity outputs had the most correct depth ordering in the scenes overall. The proposed stereo method comes second best with having 2 top percentages and 2 second-best percentages. Although RAFT did not have great results in depth and disparity errors compared to other methods in Chapter 5.1’s analysis, their method has the most consistent and accurate depth ordering.

5.4 Random-dot Stereogram

Random-dot stereograms were first invented by Julesz in 1959 while investigating stereopsis [24]. The random-dot stereogram pair consists of images composed of random dots with sections of one image transposed horizontally in the other image. When viewing these images monocularly, there is no recognizable pattern, but when viewed stereoscopically, a distinct 3D shape can be seen. It was shown that humans can perceive depth from binocular disparity in the absence of recognizable objects and features. The invention of random-dot stereograms had an impact on the development of stereo matching algorithms in the computer vision community. Testing with random dot stereograms allows for controlled disparity,

and isolating the stereo matching process by removing recognizable features and objects. In particular, among those interested in creating stereo matching algorithms based on human vision, if their algorithm is not able to handle random-dot stereograms, then they should not make this claim, since humans can perceive depth in these images.

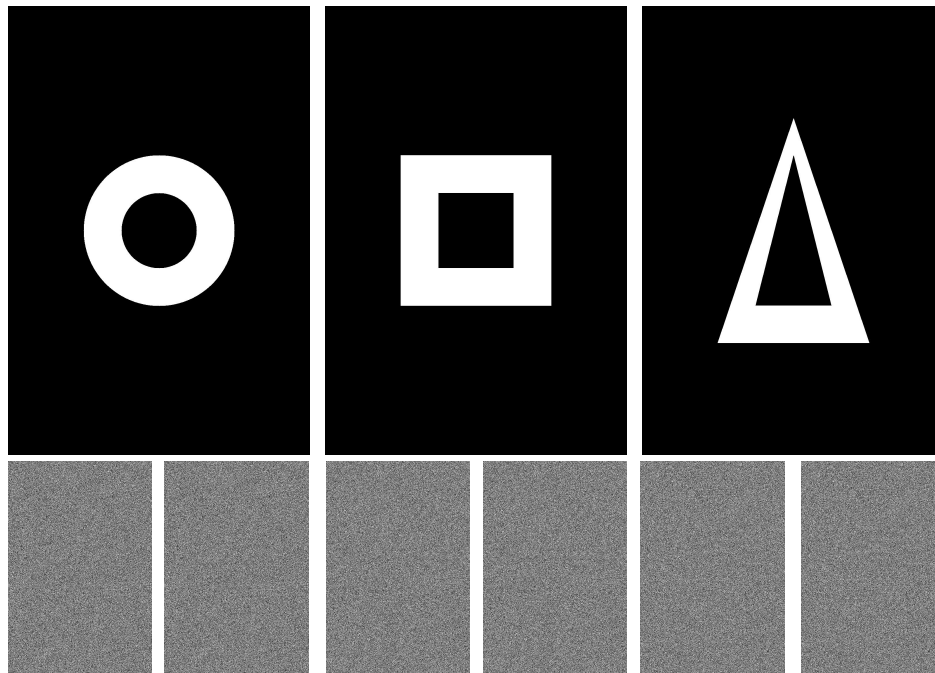


Figure 5.16: The shapes used to test random-dot stereograms with their left and right images below.

In this section, random dot stereograms were created with 3 different shapes: donut, hollow square, and hollow triangle. The disparity image, as well as sample image pairs, are shown in Figure 5.16. The image size is 1544×1038 , and for each shape, 3 different disparity values were given: 30, 50, and 60. 9 pairs of images in total were generated. To generate these images, a desired disparity map was first created from the shapes. Then, an empty image was filled with randomly selected pixels to be black. A copy of the image was created, and iterating through each pixel, if the current pixel is a pixel within the desired shape, the pixel value was shifted by the desired disparity amount. The gaps of these shifted pixels are filled with random dots. The stereograms were all tested by a human viewer to verify that humans can perceive depth in these images.

An example set of disparity results comparing different methods is shown in Figure 5.17. The proposed CAS method, along with NeRF-RAFT, ELAS, and SGBM, did reasonably

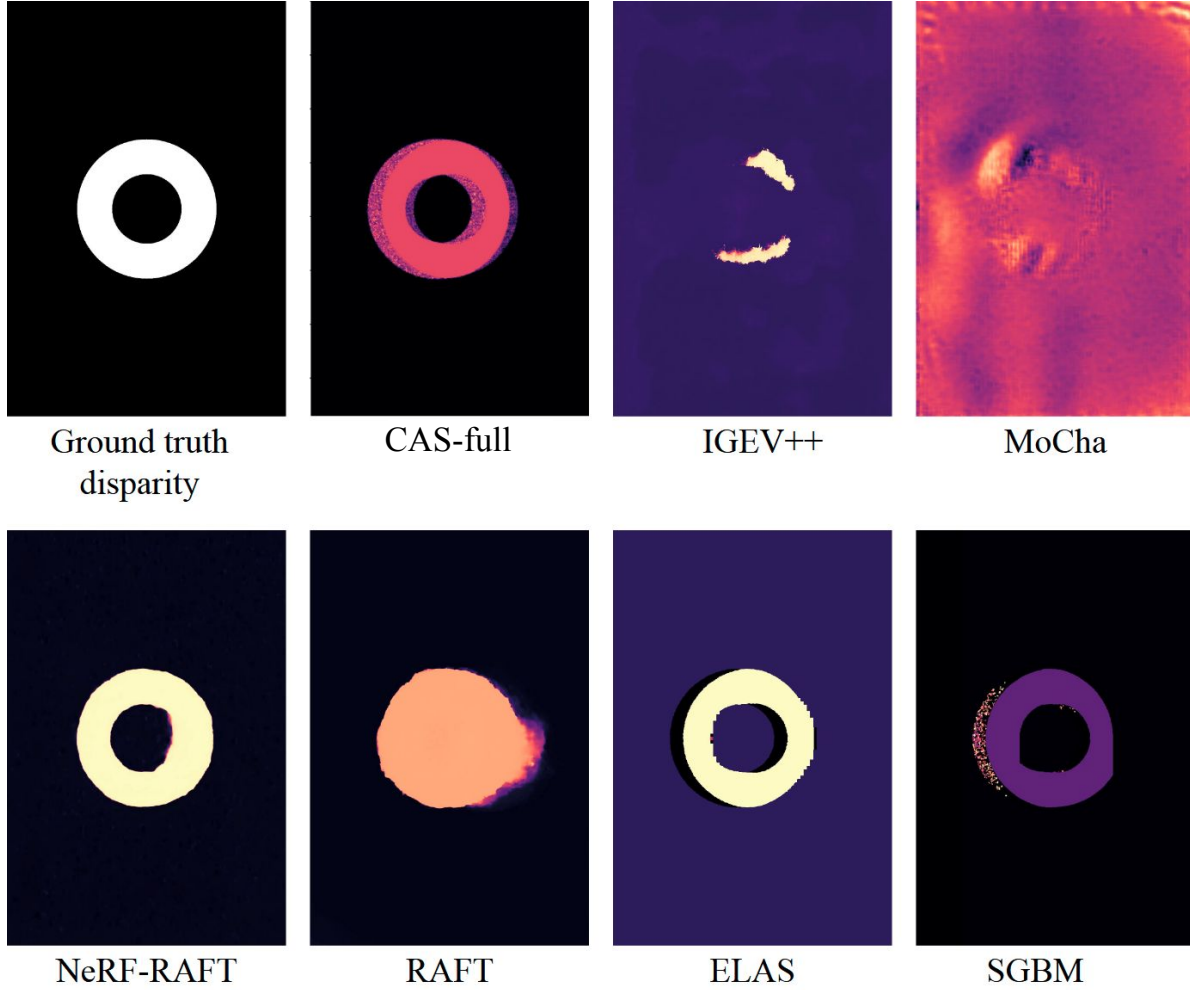


Figure 5.17: Disparity output results on the donut random-dot stereogram

well on this set. IGEV++, MoCha, and RAFT were not able to produce good disparity maps from random-dot stereograms.

Table 5.30: Stereogram disparity analysis with percentage of bad pixels metric.

Bad pix	CAS-full	SGBM	ELAS	IGEV++	MoCha	NeRF-RAFT	RAFT
0.5	11.43	6.30	6.00	77.87	15.47	12.63	17.59
1.0	11.32	6.54	5.99	46.77	15.09	10.23	16.21
2.0	11.17	6.51	5.91	11.62	10.41	7.50	10.16
5.0	10.88	6.4	5.79	10.41	14.53	5.84	12.49

CAS convergent stereo method performs the best when comparing methods against mean and median disparity error, as well as the standard deviation of disparity error. Results from all methods tested across all stereogram sets are displayed in Table 5.31. When comparing

Table 5.31: Stereogram disparity analysis with mean disparity error, median disparity error, and disparity error standard deviation metrics.

	CAS-full	SGBM	ELAS	IGEV++	MoCha	NeRF-RAFT	RAFT
Mean	20.1	22.11	20.41	21.76	22.96	22.16	22.29
Median	0	0.03	0	0	0.12	0.13	0.1
std	76.3	87.41	83.84	87.84	87.16	88.31	87.32

using the percentage of bad pixels metric, ELAS performs the best across all random-dot stereogram sets.

Random-dot stereograms, when viewed by humans, allow for many adjustments of fixation and vergence of the eyes to obtain a perception of depth. However, when tested by computer vision algorithms, the free viewing process and vergence adjustments are eliminated, and the problem becomes a parallel stereo correspondence problem with only negative horizontal disparity, which is not particularly suited for the convergent stereo system at hand. The results from Table 5.30 are consistent with this, as ELAS, a parallel traditional stereo method which does not suffer from random-dot stereograms being out of typical stereo training set domains, performs the best with random-dot stereogram images. However, the proposed algorithm is still able to tackle these images well despite not having ideal viewing conditions and access to camera geometry information, while deep learning methods sometimes do poorly at the task.

5.5 Middlebury Parallel Images

As a test to confirm that the proposed convergent stereo matching method can be used on scenes outside of the created convergent stereo dataset, the proposed algorithm was tested on some parallel images in the Middlebury 2021 dataset. Example disparity outputs are shown for a scene in Figure 5.18. The parallel stereo methods give very nice results in these images. ELAS and SGBM, which often gave large parts of the image blotchy or unassigned, produced nice disparity maps in these images compared to the parallel images in the dataset that lacked a large variety of visual cues.

The disparity results shown in Table 5.32 and Table 5.33 are similar to the results displayed on the Middlebury benchmark. However, the images chosen were not from the eval-

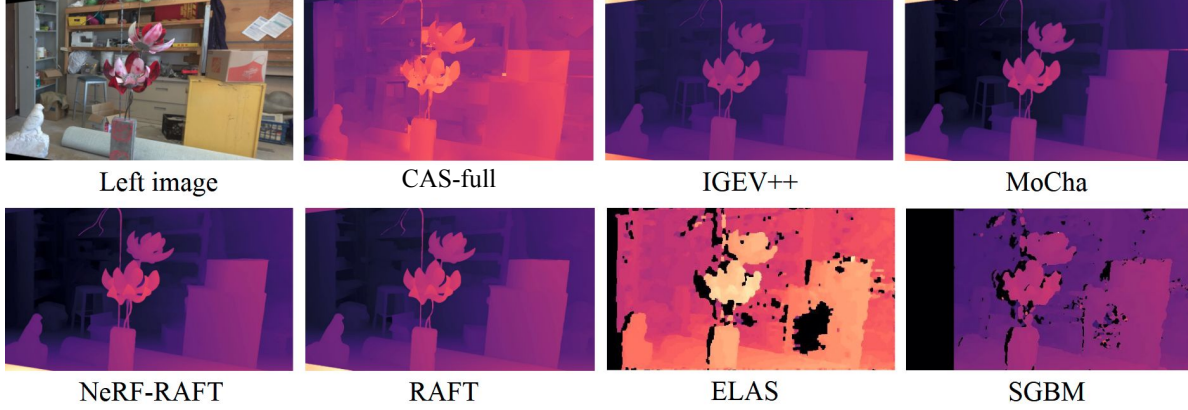


Figure 5.18: Example output disparity images of one of the scenes from the Middlebury stereo benchmark. The “Left image” shows the left image of the image pair.

Table 5.32: Middlebury disparity analysis with mean disparity error, median disparity error metrics.

	CAS-full	SGBM	ELAS	IGEV++	MoCha	NeRF-RAFT	RAFT
Mean	7.1	19.5	14.2	0.8	1.1	1.0	0.9
Median	2.8	1.5	1.7	0.3	0.3	0.3	0.3

Table 5.33: Middlebury image disparity analysis with percentage of bad pixels metric.

	CAS-full	SGBM	ELAS	IGEV++	MoCha	NeRF-RAFT	RAFT
0.5	87.61	76.56	81.99	46.28	49.17	49.48	43.98
1.0	82.16	63.62	69.99	28.25	30.21	32.07	24.84
2.0	73.61	51.92	51.74	17.83	18.50	21.58	13.09
5.0	58.47	43.65	39.11	12.62	13.09	15.87	6.48
8.0	50.29	41.33	36.31	11.42	11.35	13.66	4.38
10.0	46.42	40.36	35.35	10.98	10.72	12.74	3.35
15.0	39.36	38.82	34.04	10.34	9.93	10.60	2.05
20.0	34.27	37.56	33.12	9.87	9.57	8.28	1.66

uation set, which is why the ordering of best results may be different. The deep learning methods did the best on the Middlebury images, while CAS and the traditional methods lagged behind. The proposed CAS method performs slightly worse than SGBM and ELAS on the Middlebury images since the proposed convergent method does not make any assumptions that parallel stereo methods make, such as ordering constraints and local support assumptions with the same-sized patches.

Although the CAS convergent stereo method does not give the best results in the Middlebury dataset, it is still important to show that the proposed convergent stereo algorithm

can compute parallel image disparity maps and produce a decent output. With the proposed method of convergent stereo computation, it stands that the convergent stereo system has a versatile system that can be applied to many different cases without completely failing in any of them. The same cannot be said about the parallel stereo methods analysed throughout this thesis.

5.6 Summary

This chapter evaluates and compares CAS against traditional parallel stereo methods like SGBM [37] and ELAS [39], as well as several state-of-the-art deep learning methods including IGEV++ [46], MoCha [45], RAFT-Stereo [44], and RAFT-Stereo which was trained on NeRF outputs [48].

The performance was first assessed using absolute depth and disparity errors, as well as the percentage of bad pixels for both disparity and depth errors across the entire dataset and specific subsets designed to test challenging scenarios. Overall, the CAS convergent stereo algorithm demonstrated strong performance in mean depth error, mean disparity error and standard deviation of errors across all scenes in the dataset when compared to other parallel methods. CAS-200 consistently showed stronger performance against parallel methods in absolute mean depth error and bad pixel depth error across many subsets including “Close”, “Smooth Planes”, “TEOS Objects”, “Repeated Patterns” and “Dark”, while having relatively weaker performance in “Far” and “Normal” subsets. Advantages of using a convergent stereo system can be seen in close objects and repeated patterns in scenes.

Experiments showed that rectifying convergent images to have epipolar lines slightly negatively affected the accuracy of depth and disparity estimates compared to using original convergent images. This suggests that the warping involved in rectification especially away from the image center can lead to higher errors in stereo. However, it was also observed that traditional parallel methods performed significantly better when applied to rectified convergent images than standard parallel images indicating an inherent advantage of viewing scenes in a convergent configuration. This benefit may come from placing corresponding points closer in image space to reduce matching ambiguities.

To evaluate each method’s ability to determine relative depth between objects, a depth ordering analysis was performed. For each scene, all fixations were paired, and their ground truth depths were used to establish a correct depth order from closest to farthest fixations. The estimated depth at each fixation was averaged within a 10-pixel radius of the fixation point. RAFT’s disparity output yielded the most correct depth ordering across all scenes with 74.9% accuracy while CAS came second overall with 68.4%.

Experiments were conducted to test each method’s ability to resolve disparities on random-dot stereograms. Stereograms have many features but no recognizable objects. Most data-driven methods fail on these images indicating that they rely on scenes which look similar to their training domain to perform well. Traditional methods along with CAS performed well in random-dot stereograms. Quantitatively, the CAS method obtained the best performance in terms of mean, median, and standard deviation of disparity errors across all random-dot stereogram sets. ELAS obtained the best results in the percentage of bad pixels metric.

CAS was also evaluated on parallel Middlebury images from the 2021 dataset to confirm that the proposed CAS algorithm can be used on scenes outside of the proposed convergent stereo dataset. The results indicated that deep learning methods generally performed the best while CAS and traditional methods SGBM and ELAS lagged behind. The slightly weaker performance of CAS compared to SGBM and ELAS is attributed to the fact that CAS does not take advantage of parallel stereo simplifications and assumptions such as ordering constraints and same-sized patches. The Middlebury dataset is designed for fixed parallel stereo camera configurations with parallel epipolar lines. Despite not achieving best results on this benchmark, CAS is still able to compute parallel image disparity maps and produces decent outputs, highlighting the versatility of convergent stereo systems.

The results suggest that an active convergent stereo system offers advantages to fixed parallel systems in challenging scenarios and provides more versatility in unfamiliar scene structures. While data driven methods excel in well-represented training domains, the CAS method exhibits a more consistent and reliable performance across diverse scenarios which is an important feature in robotics applications where robustness and predictability is needed.

Chapter 6

Conclusion

6.1 Implications

The work performed in this thesis has several implications. The first stereo dataset of convergent natural stereo images with fixated images, ground truth disparity, and the scene with parallel images as well. The dataset includes a multitude of scenes that cover a wide variety of scene structures and image features. Uniquely, the dataset includes some scenes that specifically target common difficult stereo scenarios. The dataset can be used to further investigate the relationship between eye/camera positions and static scene perception. Using this dataset, the first experiment that evaluates different aspects of stereo performance between an active convergent pair of cameras on a robot with similar degrees of freedom as the human visual system and traditional fixed parallel stereo cameras was conducted. The importance of added degrees of freedom in camera orientations in discriminating depth in a binocular robot was quantified.

The proposed convergent stereo algorithm CAS which leverages the known camera geometry provided by motor calibration achieved the lowest mean depth error, mean disparity error, and standard deviation of errors across all scenes in the dataset when compared to traditional parallel methods SGBM and ELAS and several parallel state of the art deep learning methods (IGEV++, MoCha, NeRF-RAFT, RAFT). While deep learning methods generally performed better than traditional ones, CAS maintained a lower average error, demonstrating clear advantages within the dataset, particularly in the “Repeated Patterns”

subset. While NeRF-RAFT performed similarly to the convergent stereo method in many of the other subsets, CAS consistently gave noticeably better depth results compared to all other methods tested in the “Dark”, “TEOS Objects”, “Smooth Planes”, and “Close” subsets. CAS also performed well in depth ordering, coming only second to RAFT.

In scenes where corresponding pixels were ambiguous due to repeated patterns, lack of image features and textures, and self-occlusions, the proposed CAS method gave significantly lower average depth and disparity error compared to parallel stereo methods. This suggests that active convergent systems can excel where parallel methods struggle with correspondence ambiguity. Having the ability to fixate on a target gave an advantage to the convergent stereo system by taking advantage of the known depth at the point of fixation from camera geometry, avoiding disparity ambiguities at this point and its surrounding pixels. Additionally, the ability to fixate and provide additional views of the scene, especially at the parallel image edges and corners, gives more image information to better inform stereo matching. These findings support the findings of Tresilian [14] where the authors found that in human vision vergence cues play a larger role in depth estimation in two scenarios: when fixation distance becomes closer, and retinal image cues decrease.

The process of rectifying non-parallel stereo image pairs to be parallel appears to slightly negatively affect the accuracy of depth and disparity estimates. Previously, many methods ignored the possible consequences of image rectification, however, the experiments indicate that using the original convergent images in the proposed convergent stereo method showed better accuracy than the same method applied to rectified images. This suggests that while rectification simplifies the stereo matching search by enforcing the horizontal epipolar line constraint, the warping involved, especially farther away from the image center, can lead to interpolation errors. For maintaining depth and disparity accuracy, performing disparity calculation on original convergent images may be preferable. However, when given rectified convergent images as opposed to parallel images, traditional methods performed significantly better with convergent images which indicates that there is an advantage of using convergent images over parallel images.

One of the issues that can be highlighted for the widely preferred deep learning stereo methods is the limitation of the training set’s domain as shown by the poor performance of

data-driven methods in many subset experiments. While datasets are becoming larger and larger, it is still difficult to capture all possible objects, scenes, and scenarios that a stereo system may come across. One could argue that a few of the scenes constructed in the dataset and the random-dot stereograms are out-of-domain objects and scenarios, which affected the performance of the deep stereo methods. Although this may be true, the reliance of deep learning methods on a perfectly encompassing training dataset for their performance and ability to generalize well is an indication of an inherent flaw of these methods.

The CAS convergent stereo method, along with traditional methods, does not rely on any particular training dataset to perform disparity computations. Overall, it was able to perform decently across all tests, indicating its ability to handle many different difficult scenarios. The parallel methods chosen for comparison excelled in a few tasks and failed significantly in others. The convergent stereo system is unlike the parallel methods tested, namely the ability to change camera geometry and the ability, although not tested, to move closer to objects of interest. In robotics applications where robustness and reliability are important considerations for selecting methods, the convergent stereo method may offer a better alternative than deep stereo methods, where robustness and consistency are essential.

This work presents the first convergent stereo algorithm (CAS) that computes both horizontal and vertical disparities resulting in a dense disparity map. CAS has been tested, alongside other methods, for a quantitative evaluation on a new convergent image dataset obtained with a robotic pair of cameras. Throughout many experiments conducted, CAS performs the best in general with decent results throughout. State-of-the-art methods, while excelling in some tasks, fail in other challenging or out-of-domain scenarios, revealing the short comings of data-driven approaches. The promising results of this work indicate that active convergent stereo can be a fruitful area of research for future works.

6.2 Limitations

Although the results of the proposed algorithm are promising, this research has a few limitations and several avenues that can be strengthened. One limitation of the proposed algorithm, as with many other traditional methods, is its sensitivity to chosen Gabor filter

parameters and the defined distance of correspondence search. These values were carefully hand-tuned before conducting experiments however, there may be better parameters that provide better results. Machine learning may be used to optimize these parameters. Additionally, a small number of parameters were chosen for efficiency but in human brains, there are many more orientations of Gabor filters than those chosen. Including more orientations may improve the performance of the algorithm.

The binocular convergent system is reliant on accurate motor calibration results. Many aspects of the pipeline rely on the estimated fundamental matrix. These aspects include determining the epipolar lines and defining the search space, projecting disparity calculations into depth estimates, and rotating point clouds to the parallel configuration to evaluate against ground truth measurements. If there is noise or an error in the fundamental matrix calculation, many aspects of the pipeline would be affected with errors that would propagate until the final output. Additionally, through a prolonged period of use, mechanical parts may shift in a way that affects the offline calibration performed at the start of experiments. Thus, it is recommended to perform offline calibration before any long-term experiment. The effects of any mechanical shifts were not observed throughout the collection of 100+ fixations, though if CODY is mounted on an active mobile robot, this may change. There may also be ways to adapt constantly with online calibrations to maintain correct calibration throughout the entire duration of experiments.

The system does not perform the best with target fixations at larger distances, shown in Chapter 5.1.2. Another cause of CAS’s relatively poor performance compared to the state of the art parallel methods is that scenes with far objects are more reflective of the datasets that data-driven methods are trained on. With this, it is expected that data-driven methods would perform best in these scenarios. CAS’s poor performance at far distances reinforces the findings of [14], which indicate that the reliability of depth estimates based on vergence diminishes as the distance of the target increases. When applied on a robotic platform, depth estimates within 3m can be considered reliable with an increasing degree of error as a function of distance, while depth estimates beyond this threshold can be subsidized with other information, such as monocular depth cues. Rebuilding the robot with motors with higher precision would help with the confidence and accuracy levels at farther fixations.

Additionally, by monitoring the angles of convergence, an active convergent stereo head mounted on a mobile platform can be moved closer to the target if the convergence angle indicates a far object.

6.3 Future Work

An avenue to explore in the future may be to include cyclotorsion in data collection. Including Helmholtz torsion at fixation points and analysing its effects on disparity and depth estimation would be interesting. Applying torsion in ways to reduce vertical disparity between pairs of fixated images may help reduce the search space and better align images for more confident matching.

When combining estimates for rotations in the calibration process, a better method to average the rotations would be to use quaternions. A single 3D rotation can be represented by multiple sets of Euler angles. However, when filtering out rotations that are too unlike the other estimates, this problem is avoided, but may result in discarding good estimates. Quaternions have a unique representation for 3D orientations, which may be better suited for this task.

The way the overlapping SIFT features as an indication of prior disparities is resolved can be improved by adding a factor of confidence depending on the SIFT feature size. A smaller size indicates a more confident disparity estimate, as with larger SIFT areas, one disparity estimate may not be able to accurately describe all the pixels in the area.

Adding a measure of uncertainty to the computed disparity maps may be helpful in real-world applications. Uncertainty can be a function of the difference in SSD vectors and image content. Providing an uncertainty map would allow users to decide a threshold at which they can trust the output.

A future investigation can be to evaluate disparity and depth maps at each fixation based on the pixel's distance to the horizontal and vertical horopters. In human binocular vision, binocular fusion only occurs on the horopters [10, 11]. It is expected that the depth of 3D points that fall outside of these horopters have poor depth estimation. It would be interesting to see if this holds true in a binocular convergent robotic system as well.

Finally, although the data collected was rich in various scenarios and scene complexity, the division of subsets had an uneven number of scenes in each set, as this division was not the main goal when collecting data. For example, “Normal” objects contained 15 scenes while “TEOS objects” only had 5. This imbalance might have resulted in a bias in the overall results if one type of scene was dominating over others. However, by evaluating each subset individually, the bias was addressed. In the future, more data for more scenes can be collected to maintain an even distribution among all subsets.

6.4 Conclusion

The motivation for this thesis came from the observed lack of research regarding the differences in depth perception between a robot head with cameras that have the active ability to fixate on different 3D points of the scene and a traditional fixed parallel stereo camera. While extensive research exists on stereo matching algorithms for fixed parallel stereo systems, there is limited exploration of active convergent stereo cameras. Investigating depth perception in robots with added camera degrees of freedom can reveal advantages that come with these degrees of freedom.

This research aimed to answer 3 research questions to provide insights into the differences in performance between fixed parallel stereo and active convergent stereo. In regards to **RQ1** “How do the resulting depth maps of fixed parallel stereo compare to convergent stereo with multiple fixation points in terms of accuracy and map fidelity in various regions?”, the proposed convergent stereo achieved the lowest mean depth error, mean disparity error, and standard deviation of errors across all scenes. The proposed convergent stereo method achieved the lowest absolute depth and disparity error in many of the divided subsets as well as maintained low percentage of pixels above x depth and disparity error thresholds. CAS and performed significantly better than all methods tested in the “Repeated patterns” subset. Regarding disparity map fidelity, deep learning methods tend to hallucinate clouds of seemingly random disparities in areas where disparity matching had ambiguities, such as a plane tabletop. ELAS resorted to not assigning any disparity in difficult areas, while SGBM would assign incorrect blotches of disparity. Although the convergent method also

would struggle in these difficult regions, due to the advantage of fixation, it would be able to estimate disparities much closer to the ground truth. Additionally, the proposed convergent method does not tend to assign extremely wrong disparities, shown by the low standard deviation of error in depth error and disparity error, unlike the parallel stereo methods that were tested against.

To answer **RQ2** “Is there an advantage to having the ability to fixate on different areas in the scene for depth perception? If so, in which scenarios are the added degrees of freedom particularly useful?”, the advantages to using a convergent configuration with multiple fixation points, became particularly evident in challenging scenarios. Scenes with a lack of visual cues and complex occlusions proved difficult for parallel methods, while the convergent method was able to leverage camera geometry information in fixations to estimate disparity and depth more accurately. Scenes with repeated patterns, close objects, and objects with a high amount of self-occlusion differentiate the proposed convergent stereo method from the majority of the parallel methods.

Regarding **RQ3** “What are the consequences of rectifying convergent images to be parallel and is it better for depth accuracy to perform disparity calculations using original convergent images?”, methods that performed disparity calculations using the original convergent images (CAS) generally showed better accuracy than when the same method (CAS rect) was applied to rectified images. This suggests that while rectification simplifies the stereo matching search space by enforcing the horizontal epipolar line constraint, the warping involved, particularly away from the image center, can lead to information loss and interpolation errors.

In conclusion, the findings support the idea that an active convergent stereo system offers geometric advantages over fixed parallel systems, particularly in scenarios that challenge traditional and deep learning parallel methods, such as scenes with repeated patterns or significant self-occlusion. The ability to perform computations on multiple fixations and utilize original convergent images without rectification contributes to improved accuracy in various conditions. These results provide valuable insights for the design of vision systems in humanoid robots, suggesting that mimicking human-like camera degrees of freedom and employing convergent stereo processing can lead to more effective depth perception, especially in complex and ambiguous environments.

Appendix A

Stereo Dataset

The following displays examples of the dataset with each subset. All images are shown in the left parallel configuration. Each subset were selected from the full dataset. Many scenes in the dataset were created to represent a complex and difficult type of scene for stereo matching. The subsets categorize them into their respective subsets. There are scenes which fall into multiple subset categories. Splitting the dataset into different subcategories allows for deeper analysis into the behaviour of each stereo method. It highlights the strengths and weaknesses in each method in relation to the types of objects in the scene and scene structured which is important to know when choosing a method to use for particular applications.

Details on how to download the openly available dataset can be found here:

<https://jtl.lassonde.yorku.ca/>.

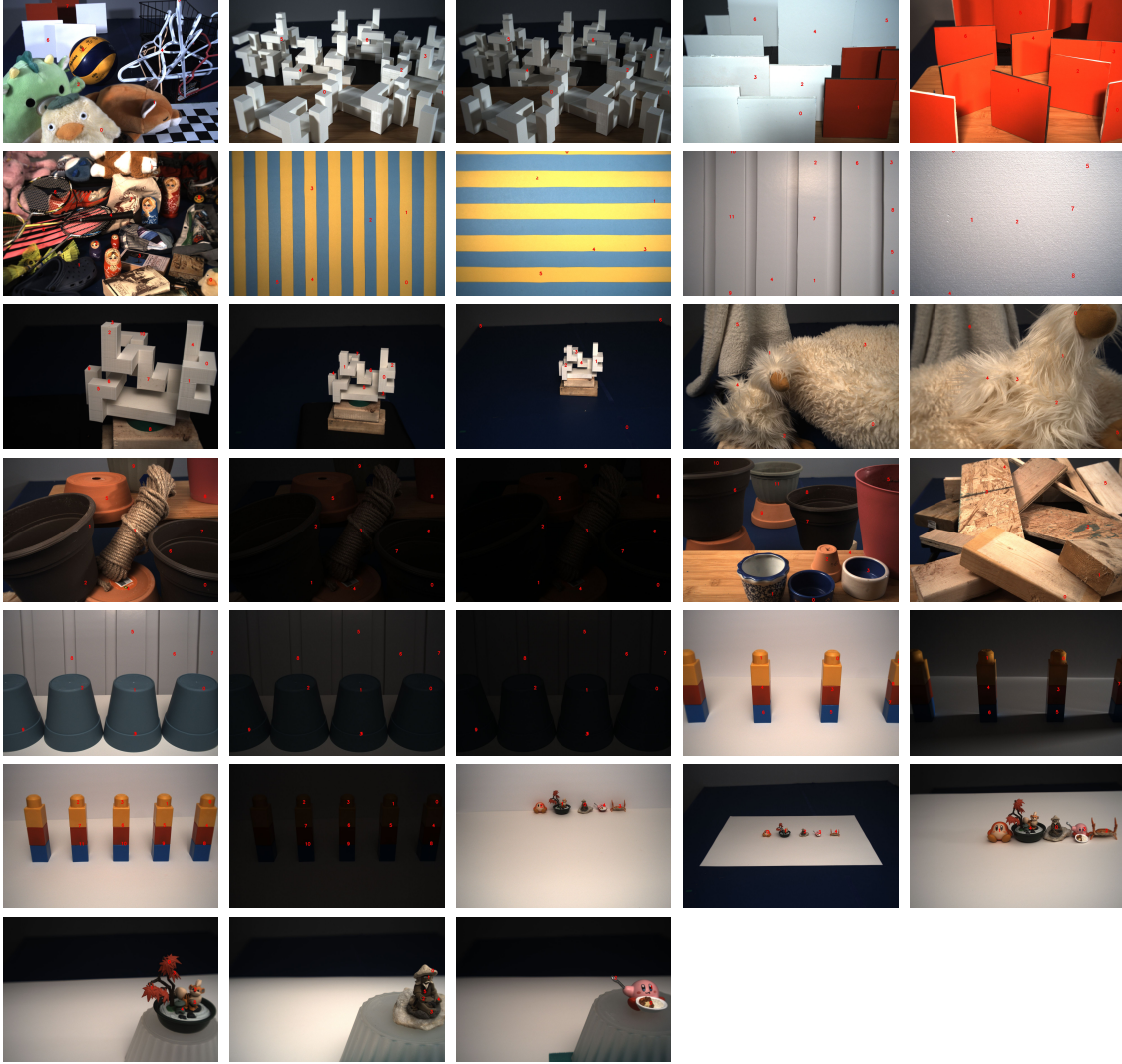


Figure A.1: The full dataset of 33 scenes with their fixations visualized in red numbers on their parallel camera configuration left image.

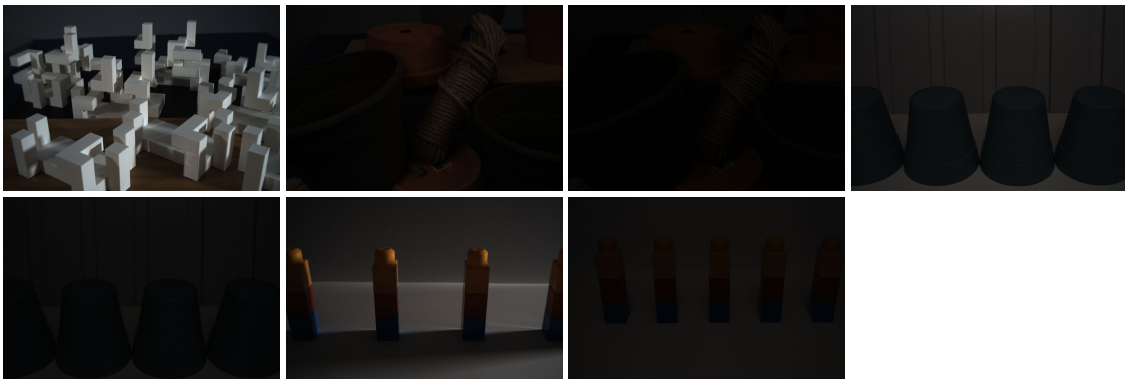


Figure A.3: The subset of scenes that has darker illumination.



Figure A.2: The subset of scenes that contain objects that are close to the robot head.



Figure A.4: The subset of scenes that contain objects that are far from the robot head. Objects were placed at 1.5m or more away from the robot’s cameras’ baseline.

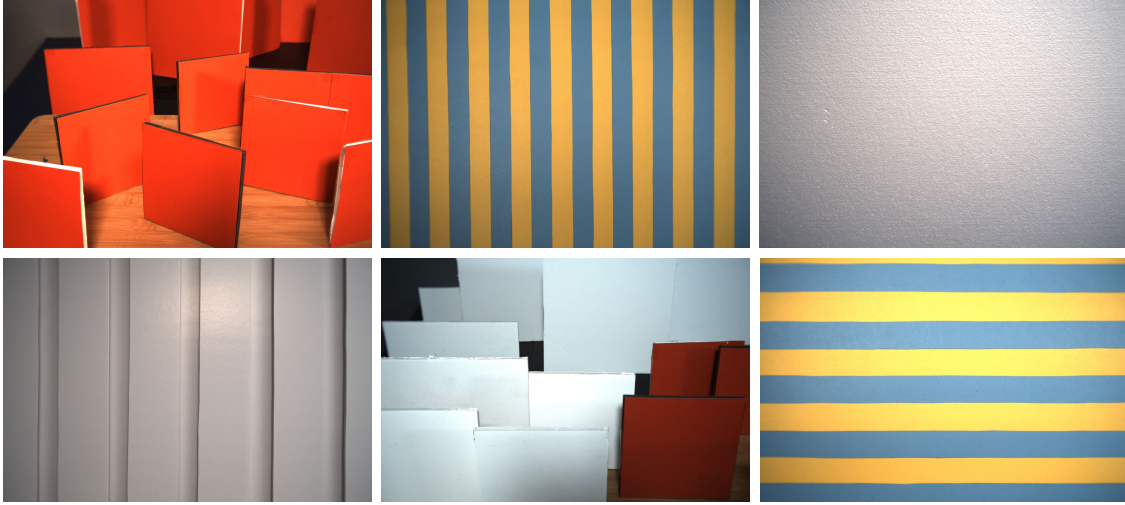


Figure A.5: The subset of scenes that contain objects that had smooth planes without many features and textures.

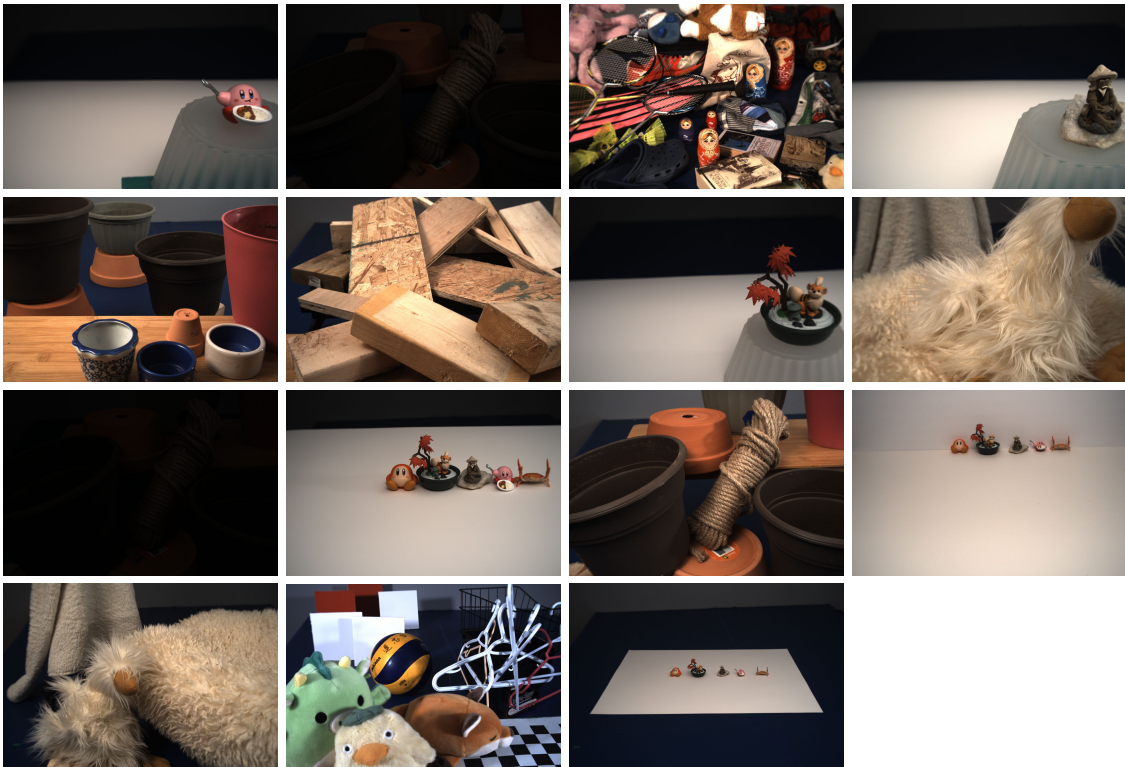


Figure A.6: The subset of scenes that contain objects that are typically found in stereo training sets and benchmarks. These objects should be similar to the objects in the domain of the training set for deep learning stereo models.

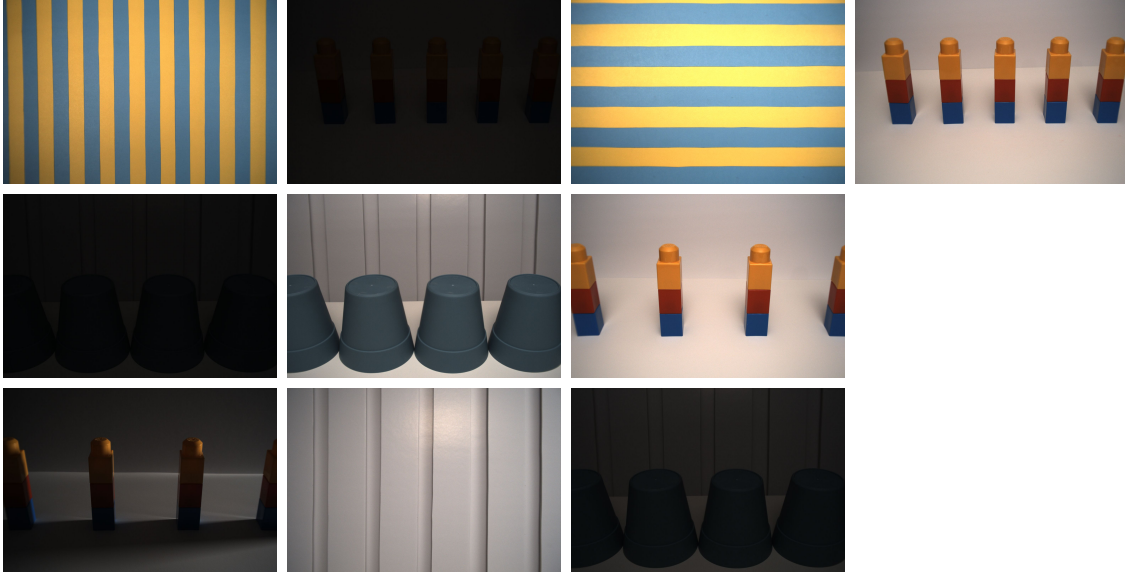


Figure A.7: The subset of scenes that contain objects that had repeated patterns.

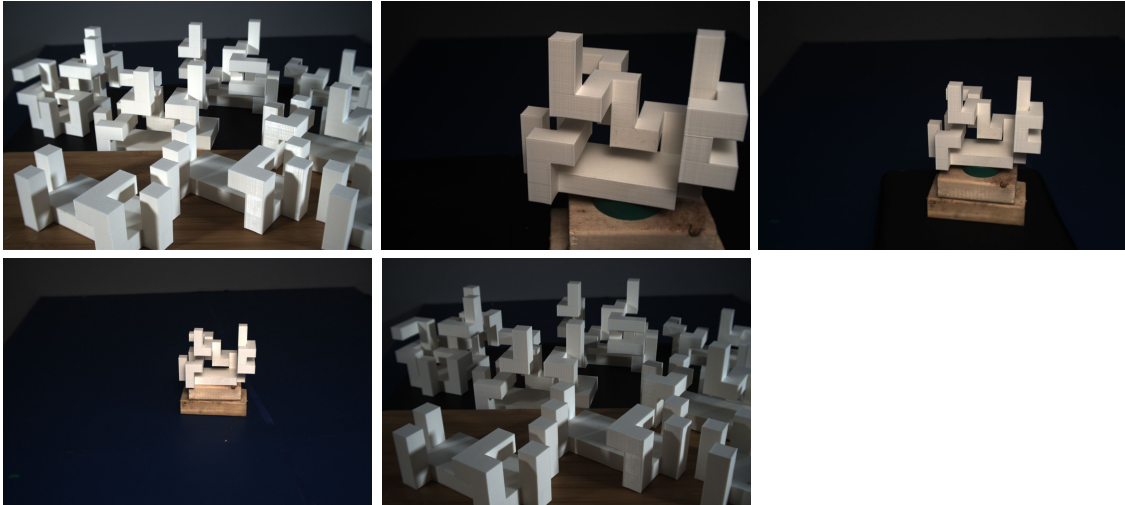


Figure A.8: The subset of scenes which contain TEOS [4] self-occlusion blocks

Appendix B

Image Size and Stereo Methods

The following shows the effects of image size on traditional stereo methods, SGBM and ELAS. Both methods used recommended parameters. Each Figure shows the levels of a Gaussian pyramid of the left image in a stereo pair in the top row and the results of the algorithm applied on the multiple image resolutions in the bottom row. A Gaussian pyramid is a multi-resolution image representation created by iteratively applying a Gaussian blur and down-sampling the image. Both traditional algorithms performed best at lower resolutions, reflective of the typical size of stereo images around the time which these algorithms were published. The size of the image that SGBM and ELAS used in the analysis were the down-sampled versions of the original images to reflect the image sizes which these algorithms were intended for. The results were then upsampled to match the image size used in the analysis.

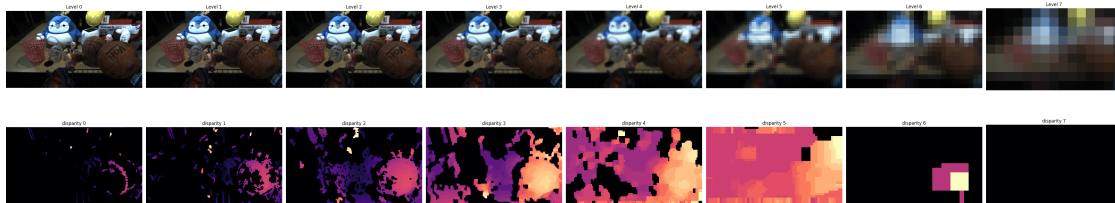


Figure B.1: ELAS results on different-sized images. Level 0 is the full-sized image, and every level is downsized by a factor of 2 from the previous level.

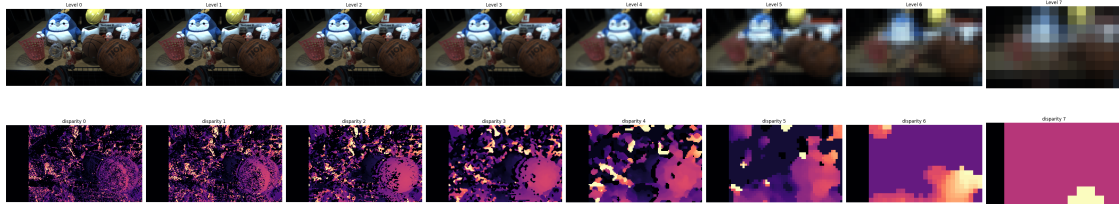


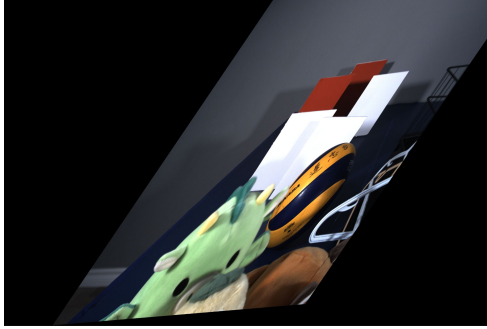
Figure B.2: SGBM results on different-sized images. Level 0 is the full-sized image, and every level is downsized by a factor of 2 from the previous level.

Appendix C

Rectification Methods

There are various methods to estimate the fundamental matrix of an image pair. The fundamental matrix provides information on the epipolar lines between two images and is needed for rectifying images to have parallel epipolar lines. The fundamental matrix alongside the camera calibration matrix for both cameras can be used to compute the essential matrix which contains information on the extrinsic relationship between two cameras. The extrinsic relationship along with intrinsic camera parameters are needed to triangulate matched points correctly into 3D space.

A good fundamental matrix estimation is important in terms of image rectification for a non-parallel pair of images since the rectified images are inputs to parallel stereo algorithms. Three fundamental matrices were used to rectify a few sample images, and the most consistent and visually best method was chosen to be used in the algorithm. Some methods while keeping matching points on the same horizontal epipolar line, would dramatically skew the image which would affect matching algorithms which use patch matching techniques. Warped and slanted images would violate the assumption of patches having the same shape and size that many traditional parallel methods use. Ultimately, rectifying using the motor-camera calibration process yielded best results, preserving scene structure the most.



a) rectification using only SIFT features



b) rectification using SIFT features filtered with F



c) rectification using only motor values



Figure C.1: Example rectification results from each method of finding the fundamental matrix. Only using SIFT features in method a. resulted in an incorrect rectification. In method b., with motor value fundamental matrix filtering of SIFT feature, the corresponding points in both images in lines are indeed on parallel epipolar lines, but the image is still rectified incorrectly as it distorts the structure of some objects. Method c. was found to have the best rectification.

Bibliography

- [1] E. S. Perkins and H. Davson, “Human Eye.” <https://www.britannica.com/science/human-eye>, 2024. [Accessed 26-Jul-2024].
- [2] A. Canessa, A. Gibaldi, M. Chessa, M. M. Fato, F. Solari, and S. P. Sabatini, “A dataset of stereoscopic images and ground-truth disparity mimicking human fixations in peripersonal space,” *Scientific Data*, vol. 4, 2017.
- [3] H. Sahabi and A. Basu, “Analysis of Error in Depth Perception with Vergence and Spatially Varying Sensing,” *Computer Vision and Image Understanding*, vol. 63, no. 3, pp. 447–461, 1996.
- [4] M. D. Solbach and J. K. Tsotsos, “Blocks World Revisited: The Effect of Self-Occlusion on Classification by Convolutional Neural Networks,” *CoRR*, vol. abs/2102.12911, 2021.
- [5] I. P. Howard, “,” in *Perceiving in Depth: Volume 1 Basic Mechanisms*, ch. Vergence eye movements, Oxford University Press, 2012.
- [6] B. Cumming and G. Deangelis, “The physiology of stereopsis,” *Annual review of neuroscience*, vol. 24, pp. 203–38, 2001.
- [7] T. Pansell, J. Ygge, and H. D. Schworm, “Conjugacy of Torsional Eye Movements in Response to a Head Tilt Paradigm,” *Investigative Ophthalmology Visual Science*, vol. 44, no. 6, p. 2557, 2003.
- [8] J. Read, “The place of human psychophysics in modern neuroscience,” *Neuroscience*, vol. 296, pp. 116–129, 2015.

- [9] I. P. Howard, “1 Introduction,” in *Perceiving in Depth: Volume 2 Basic Mechanisms*, Oxford University Press, 2012.
- [10] C. W. Tyler and A. B. Scott., “Binocular Vision,” *R. E. Records*, pp. 643–674, 03 1979.
- [11] P. L. Panum, “Physiologische Untersuchungen über das Sehen mit zwei Augen,” *Kiel Schwesers*, 1858.
- [12] B. T. Backus, M. S. Banks, R. van Ee, and J. A. Crowell, “Horizontal and vertical disparity, eye position, and stereoscopic slant perception,” *Vision Research*, vol. 39, no. 6, pp. 1143–1170, 1999.
- [13] J. A. D. Silva, “Scales for Perceived Egocentric Distance in a Large Open Field: Comparison of Three Psychophysical Methods,” *The American Journal of Psychology*, vol. 98, no. 1, pp. 119–144, 1985.
- [14] Tresilian, J. R., Mon-Williams, M., Kelly, B. M., “Increasing confidence in vergence as a cue to distance,” in *Biological Sciences*, p. 39–44, 1999.
- [15] H. Ono, A. P. Mapp, and I. P. Howard, “The cyclopean eye in vision: the new and old data continue to hit you right between the eyes,” *Vision Research*, vol. 42, no. 10, pp. 1307–1324, 2002.
- [16] D. Scharstein, R. Szeliski, and R. Zabih, “A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms,” *IEEE Workshop on Stereo and Multi-Baseline Vision (SMBV)*, p. 131–140, 2001.
- [17] R. A. Hamzah and H. Ibrahim, “Literature Survey on Stereo Vision Disparity Map Algorithms,” *Journal of Sensors*, vol. 2016, p. 1–23, 2016.
- [18] B. Tippetts, D. J. Lee, K. Lillywhite, and J. Archibald, “Review of stereo vision algorithms and their suitability for resource-limited systems,” *Journal of Real-Time Image Processing*, vol. 11, no. 1, p. 5–25, 2016.

- [19] U. Dhond and J. Aggarwal, “Structure from stereo-a review,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 19, no. 6, p. 1489–1510, 1989.
- [20] K. Zhou, X. Meng, and B. Cheng, “Review of Stereo Matching Algorithms Based on Deep Learning,” *Computational Intelligence and Neuroscience*, p. e8562323, 2020.
- [21] N. Lazaros, G. C. Sirakoulis, and A. Gasteratos, “Review of Stereo Vision Algorithms: From Software to Hardware,” *International Journal of Optomechatronics*, vol. 2, no. 4, p. 435–462, 2008.
- [22] K. Y. Kok and P. Rajendran, “A Review on Stereo Vision Algorithm: Challenges and Solutions,” *ECTI Transactions on Computer and Information Technology (ECTI-CIT)*, vol. 13, no. 2, p. 112–128, 2020.
- [23] G. F. Poggio and T. Poggio, “The Analysis of Stereopsis,” *Annual Review of Neuroscience*, vol. 7, no. 1, p. 379–412, 1984.
- [24] B. Julesz, “Binocular depth perception of computer-generated patterns,” *The Bell System Technical Journal*, vol. 39, no. 5, pp. 1125–1162, 1960.
- [25] D. Marr and T. Poggio, “Cooperative Computation of Stereo Disparity,” *Science*, vol. 194, no. 4262, pp. 283–287, 1976.
- [26] D. Marr and T. Poggio, “A computational theory of human stereo vision,” *Proc. R. Soc. Lond. B*, vol. 204, pp. 301–328, 1979.
- [27] F. Campbell and J. Robson, “Application of Fourier analysis to the visibility of gratings,” *The Journal of Physiology*, vol. 197, p. 551–566, August 1968.
- [28] B. Julesz and J. E. Miller, “Independent Spatial-Frequency-Tuned Channels in Binocular Fusion and Rivalry,” *Perception*, vol. 4, pp. 125 – 143, 1975.
- [29] L. Maffei and A. Fiorentini, “The visual cortex as a spatial frequency analyser,” *Vision Research*, vol. 13, no. 7, pp. 1255–1267, 1973.
- [30] T. D. Sanger, “Stereo disparity computation using Gabor filters,” *Biological Cybernetics*, vol. 59, p. 405–418, 1988.

- [31] T. D. Sanger, “Computation of component image velocity from local phase information,” *International Journal of Computer Vision*, vol. 5, pp. 1573–1405, 1990.
- [32] M. R. Jenkin, A. D. Jepson, and J. K. Tsotsos, “Techniques for disparity measurement,” *CVGIP: Image Understanding*, vol. 53, no. 1, pp. 14–30, 1991.
- [33] D. Fleet and A. Jepson, “Stability of phase information,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 12, pp. 1253–1268, 1993.
- [34] K. Langley, T. Atherton, R. Wilson, and M. Larcombe, “Vertical and horizontal disparities from phase,” *Image and Vision Computing*, vol. 9, no. 5, pp. 296–302, 1991.
- [35] H. H. Baker and T. O. Binford, “Depth from edge and intensity based stereo,” in *International Joint Conference on Artificial Intelligence*, p. 631–636, 1981.
- [36] H. Hirschmuller, “Stereo Processing by Semiglobal Matching and Mutual Information,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 328–341, 2008.
- [37] Itseez, “Open source computer vision library.” <https://github.com/itseez/opencv>, 2015.
- [38] S. Birchfield and C. Tomasi, “A pixel dissimilarity measure that is insensitive to image sampling,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 4, pp. 401–406, 1998.
- [39] G. Andreas, R. Martin, and U. Raquel, “Efficient Large-Scale Stereo Matching,” in *Asian Conference on Computer Vision (ACCV)*, 2010.
- [40] S. Fanello, U. Pattacini, I. Gori, V. Tikhonoff, M. Randazzo, A. Roncone, F. Odone, and G. Metta, “3D Stereo Estimation and Fully Automated Learning of Eye-Hand Coordination in Humanoid Robots,” *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, p. 1028–1035, 2014.

- [41] J. Zbontar and Y. LeCun, “Computing the stereo matching cost with a convolutional neural network,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, 2015.
- [42] N. Mayer, E. Ilg, P. Häusser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox, “A Large Dataset to Train Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation,” in *IEEE/CVF International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [43] L. Lipson, Z. Teed, and J. Deng, “RAFT-Stereo: Multilevel Recurrent Field Transforms for Stereo Matching,” in *International Conference on 3D Vision (3DV)*, 2021.
- [44] L. Lipson, Z. Teed, and J. Deng, “RAFT-Stereo: Multilevel Recurrent Field Transforms for Stereo Matching,” in *2021 International Conference on 3D Vision (3DV)*, pp. 218–227, 2021.
- [45] Z. Chen, W. Long, H. Yao, Y. Zhang, B. Wang, Y. Qin, and J. Wu, “MoCha-Stereo: Motif Channel Attention Network for Stereo Matching,” in *IEEE/CVF International Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 27768–27777, 2024.
- [46] G. Xu, X. Wang, Z. Zhang, J. Cheng, C. Liao, and X. Yang, “IGEV++: Iterative Multi-range Geometry Encoding Volumes for Stereo Matching,” *arXiv:2409.00638*, 2024.
- [47] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, “NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis,” in *European Conference on Computer Vision (ECCV)*, pp. 405–421, 2020.
- [48] F. Tosi, A. Tonioni, D. De Gregorio, and M. Poggi, “NeRF-Supervised Deep Stereo,” in *IEEE/CVF International Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 855–866, 2023.

- [49] D. Scharstein and R. Szeliski, “High-accuracy stereo depth maps using structured light,” in *IEEE/CVF International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2003.
- [50] D. Scharstein and C. Pal, “Learning Conditional Random Fields for Stereo,” in *IEEE/CVF International Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–8, 2007.
- [51] H. Hirschmuller and D. Scharstein, “Evaluation of Cost Functions for Stereo Matching,” in *IEEE/CVF International Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–8, 2007.
- [52] D. Scharstein, H. Hirschmüller, Y. Kitajima, G. Krathwohl, N. Nešić, X. Wang, and P. Westling, “High-Resolution Stereo Datasets with Subpixel-Accurate Ground Truth,” in *German Conference on Pattern Recognition*, pp. 31–42, 2014.
- [53] G. Xu, X. Wang, X. Ding, and X. Yang, “Iterative Geometry Encoding Volume for Stereo Matching,” in *IEEE/CVF International Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21919–21928, 2023.
- [54] T. Schöps, J. L. Schönberger, S. Galliani, T. Sattler, K. Schindler, M. Pollefeys, and A. Geiger, “A Multi-View Stereo Benchmark with High-Resolution Images and Multi-Camera Videos,” in *IEEE/CVF International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [55] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, “Indoor Segmentation and Support Inference from RGBD Images,” in *European Conference on Computer Vision (ECCV)*, pp. 746–760, 2012.
- [56] A. Janoch, S. Karayev, Y. Jia, J. Barron, M. Fritz, K. Saenko, and T. Darrell, “A Category-Level 3D Object Dataset: Putting the Kinect to Work,” in *IEEE International Conference on Computer Vision (ICCV)*, pp. 1168–1174, 11 2011.

- [57] J. Xiao, A. Owens, and A. Torralba, “SUN3D: A Database of Big Spaces Reconstructed Using SfM and Object Labels,” in *IEEE/CVF International Conference on Computer Vision*, pp. 1625–1632, 2013.
- [58] M. Menze, C. Heipke, and A. Geiger, “Joint 3D Estimation of Vehicles and Scene Flow,” *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 2, pp. 427–434, 2015.
- [59] M. Menze, C. Heipke, and A. Geiger, “Object Scene Flow,” *ISPRS Journal of Photogrammetry and Remote Sensing (JPRS)*, vol. 140, pp. 60–76, 2018.
- [60] D. Scharstein, “2021 Mobile stereo datasets with ground truth.” <https://vision.middlebury.edu/stereo/data/scenes2021/>, 2021. [Accessed 26-Jul-2024].
- [61] J. Batlle, E. Mouaddib, and J. Salvi, “Recent progress in coded structured light as a technique to solve the correspondence problem: a survey,” *Pattern Recognition*, vol. 31, no. 7, pp. 963–982, 1998.
- [62] E. Horn and N. Kiryati, “Toward optimal structured light patterns,” *Image and Vision Computing*, vol. 17, no. 2, pp. 87–97, 1999.
- [63] S. B. Kang, J. Webb, C. Zitnick, and T. Kanade, “A multibaseline stereo system with active illumination and real-time image acquisition,” in *IEEE International Conference on Computer Vision (ICCV)*, pp. 88–93, 1995.
- [64] M. Gupta, A. Agrawal, A. Veeraraghavan, and S. G. Narasimhan, “Structured light 3D scanning in the presence of global illumination,” in *IEEE/CVF International Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 713–720, 2011.
- [65] M. Gupta, A. Agrawal, A. Veeraraghavan, and S. G. Narasimhan, “A Practical Approach to 3D Scanning in the Presence of Interreflections, Subsurface Scattering and Defocus,” *International Journal of Computer Vision*, vol. 102, no. 1–3, p. 33–55, 2013.
- [66] C. Chang, S. Chatterjee, and P. Kube, “A quantization error analysis for convergent stereo,” in *IEEE International Conference on Image Processing*, pp. 735–739, 1994.

- [67] S. D. Blostein and T. S. Huang, “Error Analysis in Stereo Determination of 3-D Point Positions,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 6, pp. 752–765, 1987.
- [68] Y. Xiong and L. Matthies, “Error analysis of a real-time stereo system,” in *IEEE/CVF International Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1087–1093, 1997.
- [69] H. Zhang and S. Lee, “Robot Bionic Vision Technologies: A Review,” *Applied Sciences*, vol. 12, no. 16, 2022.
- [70] J. Rojas-Quintero and M. Rodríguez-Liñán, “A literature review of sensor heads for humanoid robots,” *Robotics and Autonomous Systems*, vol. 143, p. 103834, 2021.
- [71] M. R. M. Jenkin, *Computer Vision: A Reference Guide*, ch. Evolution of Robotic Heads, pp. 263–265. Boston, MA: Springer US, 2014.
- [72] N. Banerjee, X. Long, R. Du, F. Polido, S. Feng, C. G. Atkeson, M. Gennert, and T. Padir, “Human-supervised control of the ATLAS humanoid robot for traversing doors,” in *IEEE-RAS Conference on Humanoid Robots (Humanoids)*, pp. 722–729, 2015.
- [73] ROBOTIS, “Robotis THORMANG.” <https://www.robotis.us/thormang/>, 2016. [Accessed 26-Jul-2024].
- [74] C. Knabe, R. Griffin, J. Burton, G. Cantor-Cooke, L. Dantanarayana, G. Day, O. Ebeling-Koning, E. Hahn, M. Hopkins, J. Neal, J. Newton, C. Nogales, V. Orekhov, J. Peterson, M. Rouleau, J. Seminatore, Y. Sung, J. Webb, N. Wittenstein, J. Ziglar, A. Leonessa, B. Lattimer, and T. Furukawa, “Team VALOR’s ESCHER: A Novel Electromechanical Biped for the DARPA Robotics Challenge,” *Journal of Field Robotics*, vol. 34, no. 5, p. 912–939, 2017.
- [75] N. G. Tsagarakis, D. G. Caldwell, F. Negrello, W. Choi, L. Baccelliere, V. Loc, J. Noorden, L. Muratore, A. Margan, A. Cardellino, L. Natale, E. M. Hoffman, H. Dallali,

- N. Kashiri, J. Malzahn, J. Lee, P. Kryczka, D. Kanoulas, M. Garabini, M. Catalano, M. Ferrati, V. Varricchio, L. Pallottino, C. Pavan, A. Bicchi, A. Settini, A. Rocchi, and A. Ajoudani, “WALK-MAN: A High-Performance Humanoid Platform for Realistic Environments,” *Journal of Field Robotics*, vol. 34, no. 7, p. 1225–1259, 2017.
- [76] M. Lee, Y. Heo, J. Park, H.-D. Yang, H.-D. Jang, P. Benz, H. Park, I. S. Kweon, and J.-H. Oh, “Fast Perception, Planning, and Execution for a Robotic Butler: Wheeled Humanoid M-Hubo,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5444–5451, 2019.
- [77] Y. Sakagami, R. Watanabe, C. Aoyama, S. Matsunaga, N. Higaki, and K. Fujimura, “The intelligent ASIMO: system overview and integration,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2478–2483, 2002.
- [78] T. Asfour, R. Dillmann, N. Vahrenkamp, M. Do, M. Wächter, C. Mandery, P. Kaiser, M. Kröhnert, and M. Grotz, *The Karlsruhe ARMAR Humanoid Robot Family*, pp. 1–32. Springer Dordrecht, 2016.
- [79] N. A. Radford, P. Strawser, K. Hambuchen, J. S. Mehling, W. K. Verdeyen, A. S. Donnan, J. Holley, J. Sanchez, V. Nguyen, L. Bridgwater, R. Berka, R. Ambrose, M. M. Markee, N. J. Fraser-Chanpong, C. McQuin, J. D. Yamokoski, S. Hart, R. Guo, A. Parsons, B. Wightman, P. Dinh, B. Ames, C. Blakely, C. Edmondson, B. Sommers, R. Rea, C. Tobler, H. Bibby, B. Howard, L. Niu, A. Lee, M. Conover, L. Truong, R. Reed, D. Chesney, R. Platt, G. Johnson, C.-L. Fok, N. Paine, L. Sentis, E. Cousineau, R. Sinnet, J. Lack, M. Powell, B. Morris, A. Ames, and J. Akinyode, “Valkyrie: NASA’s First Bipedal Humanoid Robot,” *Journal of Field Robotics*, vol. 32, no. 3, p. 397–419, 2015.
- [80] R. Ambrose, H. Aldridge, R. Askew, R. Burrige, W. Bluethmann, M. Diftler, C. Lovchik, D. Magruder, and F. Rehnmark, “Robonaut: NASA’s space humanoid,” *IEEE Intelligent Systems and their Applications*, vol. 15, no. 4, pp. 57–63, 2000.
- [81] H. Iwata and S. Sugano, “Design of human symbiotic robot TWENDY-ONE,” in *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 580–586, 2009.

- [82] S. Nefti-Meziani, U. Manzoor, S. Davis, and S. K. Pupala, “3D perception from binocular vision for a low cost humanoid robot NAO,” *Robotics and Autonomous Systems*, vol. 68, pp. 129–139, 2015.
- [83] R. Tellez, F. Ferro, S. Garcia, E. Gomez, E. Jorge, D. Mora, D. Pinyol, J. Oliver, O. Torres, J. Velazquez, and D. Faconti, “Reem-B: An autonomous lightweight human-size humanoid robot,” in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pp. 462–468, 2008.
- [84] S. AI, “Building machines that think like people.” <https://sanctuary.ai/product/>, 2024. [Accessed 26-Jul-2024].
- [85] A. Andreopoulos, S. Hasler, H. Wersing, H. Janssen, J. K. Tsotsos, and E. Korner, “Active 3D Object Localization Using a Humanoid Robot,” *IEEE Transactions on Robotics*, vol. 27, no. 1, pp. 47–64, 2011.
- [86] G. Metta, L. Natale, F. Nori, G. Sandini, D. Vernon, L. Fadiga, C. von Hofsten, K. Rosander, M. Lopes, J. Santos-Victor, A. Bernardino, and L. Montesano, “The iCub humanoid robot: An open-systems platform for research in cognitive development,” *Neural Networks*, vol. 23, no. 8, pp. 1125–1134, 2010.
- [87] M. Zecca, N. Endo, S. Momoki, K. Itoh, and A. Takanishi, “Design of the humanoid robot KOBIAN - preliminary analysis of facial and whole body emotion expression capabilities,” in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pp. 487–492, 2008.
- [88] T. Asfour, K. Welke, P. Azad, A. Ude, and R. Dillmann, “The Karlsruhe Humanoid Head,” in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pp. 447–453, 2008.
- [89] R. A. Brooks, C. Breazeal, M. Marjanović, B. Scassellati, and M. M. Williamson, “The Cog Project: Building a Humanoid Robot,” in *Computation for Metaphors, Analogy, and Agents* (C. L. Nehaniv, ed.), Springer Berlin Heidelberg, 1999.

- [90] C. Breazeal and B. Scassellati, “Infant-like Social Interactions between a Robot and a Human Caregiver,” *Adaptive Behavior*, vol. 8, no. 1, pp. 49–74, 2000.
- [91] T. P. Spexard, M. Hanheide, and G. Sagerer, “Human-Oriented Interaction With an Anthropomorphic Robot,” *IEEE Transactions on Robotics*, vol. 23, no. 5, pp. 852–862, 2007.
- [92] I. Lütkebohle, F. Hegel, S. Schulz, M. Hackel, B. Wrede, S. Wachsmuth, and G. Sagerer, “The bielefeld anthropomorphic robot head “Flobi”,” in *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3384–3391, 2010.
- [93] M. Lopes, R. Beira, M. Praca, and J. Santos-Victor, “An anthropomorphic robot torso for imitation: design and experiments,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 1, pp. 661–667 vol.1, 2004.
- [94] R. Manzotti, A. Gasteratos, G. Metta, and G. Sandini, “Disparity Estimation on Log-Polar Images and Vergence Control,” *Computer Vision and Image Understanding*, vol. 83, no. 2, pp. 97–117, 2001.
- [95] X. Wang and P. Jonker, “An Advanced Active Vision System with Multimodal Visual Odometry Perception for Humanoid Robots,” *International Journal of Humanoid Robotics*, vol. 14, no. 03, p. 1750006, 2017.
- [96] G. Milighetti, L. Vallone, and A. De Luca, “Adaptive predictive gaze control of a redundant humanoid robot head,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3192–3198, 2011.
- [97] G. Pasquale, T. Mar, C. Ciliberto, L. Rosasco, and L. Natale, “Enabling Depth-Driven Visual Attention on the iCub Humanoid Robot: Instructions for Use and New Perspectives,” *Frontiers in Robotics and AI*, vol. 3, 2016.
- [98] J. Leitner, A. Bernardino, and J. Santos-Victor, “A benchmark on stereo disparity estimation for humanoid robots,” *8th Conference on Autonomous Robot Systems and Competitions Aveiro*, 2008.

- [99] K. Kaneko, F. Kanehiro, M. Morisawa, K. Miura, S. Nakaoka, and S. Kajita, “Cybernetic human HRP-4C,” in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pp. 7–14, 2009.
- [100] C. Atkeson, J. Hale, F. Pollick, M. Riley, S. Kotosaka, S. Schaul, T. Shibata, G. Tevatia, A. Ude, S. Vijayakumar, E. Kawato, and M. Kawato, “Using humanoid robots to study human behavior,” *IEEE Intelligent Systems and their Applications*, vol. 15, no. 4, pp. 46–56, 2000.
- [101] G. Cheng, S.-h. Hyon, J. Morimoto, A. Ude, G. Colvin, W. Scroggin, and S. C. Jacobsen, “CB: A Humanoid Research Platform for Exploring NeuroScience,” in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pp. 182–187, 2006.
- [102] H. Lim and A. Takanishi, “Mechanism and Control of Anthropomorphic Biped Robots,” in *Mobile Robotics, Moving Intelligence* (J. Buchli, ed.), IntechOpen, 2006.
- [103] K. Pahlavan and J.-O. Eklundh, “A head-eye system—Analysis and design,” *CVGIP: Image Understanding*, vol. 56, no. 1, pp. 41–56, 1992.
- [104] I.-W. Park, J.-Y. Kim, J. Lee, and J.-H. Oh, “Mechanical design of humanoid robot platform KHR-3 (KAIST Humanoid Robot 3: HUBO),” in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pp. 321–326, 2005.
- [105] A. Takanishi, T. Matsuno, and I. Kato, “Development of an anthropomorphic head-eye robot with two eyes-coordinated head-eye motion and pursuing motion in the depth direction,” in *IEEE/RSJ International Conference on Intelligent Robot and Systems (IROS)*, vol. 3, pp. V27–V28 vol.3, 1997.
- [106] S. Hashimoto, S. Narita, H. Kasahara, K. Shirai, T. Kobayashi, A. Takanishi, S. Sugano, J. Yamaguchi, H. Sawada, H. Takanobu, K. Shibuya, T. Morita, T. Kurata, N. Onoe, K. Ouchi, T. Noguchi, Y. Niwa, S. Nagayama, H. Tabayashi, I. Matsui, M. Obata, H. Matsuzaki, A. Murasugi, H. Kobaysashi, S. Haruyama, T. Okada, Y. Hidaki, Y. Taguchi, K. Hoashi, E. Morikawa, Y. Iwano, D. Araki, J. Suzuki, M. Yokoyama, I. Dawa, D. Nishino, S. Inoue, T. Hirano, E. Soga, S. Gen, T. Yanada,

- K. Kato, S. Sakamoto, Y. Ishii, S. Matsuo, Y. Yamamoto, K. Sato, T. Hagiwara, T. Ueda, N. Honda, D. Hashimoto, T. Hanamoto, S. Kayaba, T. Kojima, H. Iwata, H. Kubodera, R. Matsuki, T. Nakajima, K. Nitto, D. Yamamoto, Y. Kamizaki, S. Nagaike, and Y. Kunitake, “Humanoid robots in Waseda University - Hadaly-2 and WABIAN,” *Autonomous Robots*, vol. 12, no. 1, pp. 25–38, 2002.
- [107] N. Pateromichelakis, A. Mazel, M. A. Hache, T. Koumpogiannis, R. Gelin, B. Maisonier, and A. Berthoz, “Head-eyes system and gaze analysis of the humanoid robot Romeo,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1374–1379, 2014.
- [108] Q. Wang, Y. Yin, W. Zou, and D. Xu, “Measurement error analysis of binocular stereo vision: effective guidelines for bionic eyes,” *IET Science, Measurement Technology*, vol. 11, no. 7, p. 829–838, 2017.
- [109] J. Rajruangrabin and D. O. Popa, “Robot Head Motion Control with an Emphasis on Realism of Neck-Eye Coordination during Object Tracking,” *Journal of Intelligent and Robotics Systems*, vol. 63, no. 2, p. 163–190, 2011.
- [110] W. Klarquist and A. Bovik, “FOVEA: a foveated vergent active stereo vision system for dynamic three-dimensional scene recovery,” *IEEE Transactions on Robotics and Automation*, vol. 14, no. 5, pp. 755–770, 1998.
- [111] X. Zhang, “An Object Tracking System Based on Human Neural Pathways of Binocular Motor System,” in *2006 9th International Conference on Control, Automation, Robotics and Vision*, pp. 1–8, 2006.
- [112] Z. Zhang and B. Yuan, “NJU head: an active head-eye system,” in *IEEE International Conference on Intelligent Processing Systems*, vol. 2, pp. 1760–1764, 1997.
- [113] S. Huber, B. Selby, and B. Tripp, “OREO: An Open-Hardware Robotic Head That Supports Practical Saccades and Accommodation,” *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2640–2645, 2018.

- [114] Y. Kuniyoshi, S. Rougeaux, O. Stasse, G. Cheng, and A. Nagakubo, “A Humanoid Vision System for Versatile Interaction,” in *Biologically Motivated Computer Vision*, pp. 512–526, Springer Berlin Heidelberg, 2000.
- [115] S. Rougeaux and Y. Kuniyoshi, “Velocity and disparity cues for robust real-time binocular tracking,” in *IEEE/CVF International Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–6, 1997.
- [116] A. Maki, T. Uhlin, and J. O. Eklundh, “Phase-Based Disparity Estimation in Binocular Tracking,” in *Scandinavian Conference on Image Analysis*, 1993.
- [117] M. R. M. Jenkin and A. D. Jepson, *The Measurement of Binocular Disparity*, pp. 237–244. Boston, MA: Springer US, 1988.
- [118] A. Nagakubo, Y. Kuniyoshi, and G. Cheng, “ETL-Humanoid-A high-performance full body humanoid system for versatile actions,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 2, pp. 1087–1092 vol.2, 2001.
- [119] E. Milios, M. Jenkin, and J. Tsotsos, “Design and performance of TRISH, a binocular robot head with torsional eye movements,” *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 7, no. 1, pp. 51–68, 1993.
- [120] D. Biamino, G. Cannata, M. Maggiali, and A. Piazza, “MAC-EYE: a tendon driven fully embedded robot eye,” in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pp. 62–67, 2005.
- [121] G. Cannata, M. D’Andrea, and M. Maggiali, “Design of a Humanoid Robot Eye: Models and Experiments,” in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pp. 151–156, 2006.
- [122] X. Chen, C. Wang, T. Zhang, C. Hua, S. Fu, and Q. Huang, “Hybrid Image Stabilization of Robotic Bionic Eyes,” in *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 808–813, 2018.

- [123] S. Gou, X. Chen, and W. Zhang, “Multimodal Perception based Autonomous Exploration with Active Camera Control in Unknown Environments,” in *Chinese Control and Decision Conference (CCDC)*, pp. 5577–5582, 2022.
- [124] M. Jenkin and J. Tsotsos, “Active stereo vision and cyclotorsion,” *IEEE/CVF International Conference on Computer Vision and Pattern Recognition (CVPR)*, p. 806–811, 1994.
- [125] M. K. Tabrizi, M. Chi, B. B. Dey, Y. Q. Yuan, M. Solbach, Y. Liu, M. Jenkin, and J. K. Tsotsos, “The DIJIT Binocular Robotic Head: Design and Performance.” 2025.
- [126] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and M. J. Marín-Jiménez, “Automatic generation and detection of highly reliable fiducial markers under occlusion,” *Pattern Recognition*, vol. 47, pp. 2280–2292, 2014.
- [127] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, Second ed., 2004.
- [128] K. Herakleous and C. Poullis, “3DUNDERWORLD-SLS: An Open-Source Structured-Light Scanning System for Rapid Geometry Acquisition,” *CoRR*, vol. abs/1406.6595, 2014.
- [129] H. C. Longuet-Higgins, “A computer algorithm for reconstructing a scene from two projections,” *Nature*, vol. 293, pp. 133–135, 1981.
- [130] R. Hartley, “In defense of the eight-point algorithm,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 6, pp. 580–593, 1997.
- [131] D. G. Lowe, “Object recognition from local scale-invariant features,” in *IEEE International Conference on Computer Vision (ICCV)*, vol. 2, pp. 1150–1157, Ieee, 1999.
- [132] H. Bay, T. Tuytelaars, and L. Van Gool, “SURF: Speeded Up Robust Features,” in *European Conference in Computer Vision (ECCV)*, pp. 404–417, 2006.

- [133] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “ORB: An efficient alternative to SIFT or SURF,” in *2011 International Conference on Computer Vision*, pp. 2564–2571, 2011.
- [134] S. Leutenegger, M. Chli, and R. Y. Siegwart, “BRISK: Binary Robust invariant scalable keypoints,” in *2011 International Conference on Computer Vision*, pp. 2548–2555, 2011.
- [135] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, “BRIEF: Binary Robust Independent Elementary Features,” in *2010 European Conference on Computer Vision*, vol. 6314, pp. 778–792, 09 2010.
- [136] R. I. Hartley, “Chirality,” *Jour. Computer Vision IJCV*, vol. 26, no. 1, pp. 41–61, 1998.
- [137] A. Rosenfeld, “Some Uses of Pyramids in Image Processing and Segmentation,” in *DARPA Image Understanding Workshop*, pp. 112–120, 1980.
- [138] S. Marçelja, “Mathematical description of the responses of simple cortical cells*,” *J. Opt. Soc. Am.*, vol. 70, pp. 1297–1300, Nov 1980.
- [139] A. Jepson and M. Jenkin, “The fast computation of disparity from phase differences,” in *IEEE/CVF International Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 398–403, 1989.