

**Mathematisch-
Naturwissenschaftliche
Fakultät**

Computergrafik

Bachelorarbeit

The One Hough Space to scan them all

Eberhard Karls Universität Tübingen
Mathematisch-Naturwissenschaftliche Fakultät
Wilhelm-Schickard-Institut für Informatik
Computergrafik
Fridtjof Siemssen, fridtjof.siemssen@student.uni-tuebingen.de, 2024

Bearbeitungszeitraum: 01.05.2024 - 31.08.2024

Betreuer/Gutachter: Prof. Dr. Hendrik Lensch, Universität Tübingen

Selbstständigkeitserklärung

Hiermit versichere ich, dass ich die vorliegende Bachelorarbeit selbständig und nur mit den angegebenen Hilfsmitteln angefertigt habe und dass alle Stellen, die dem Wortlaut oder dem Sinne nach anderen Werken entnommen sind, durch Angaben von Quellen als Entlehnung kenntlich gemacht worden sind. Diese Bachelorarbeit wurde in gleicher oder ähnlicher Form in keinem anderen Studiengang als Prüfungsleistung vorgelegt.

Fridtjof Siemssen (Matrikelnummer 5660586), August 10, 2024

Abstract

While many photogrammetric approaches for three-dimensional scene reconstruction exceed for many applications, they are often limited by Lambertian material properties. In this thesis, we present an adaptation of an algorithm that is robust to intensity variations over multiple images due to specularities and other types of reflectance distribution. Based on a Hough transformation, this algorithm detects the object point's trajectories across a 360° rotation of said object. While this existing algorithm is limited to cameras viewing the object from the front only, this thesis offers a method to extend the acquisition process to allow for arbitrary camera angles. This enables multi-view scans to be taken for greater coverage of object parts otherwise not visible from the front. On top, we propose strategies for scaling our scans to the world coordinate system, with a focus on minimal calibration requirements.

Contents

1	Introduction	9
1.1	Motivation	9
1.2	Problem statement	9
1.3	Organisation of the Thesis	10
2	Related Work	11
2.1	Background	12
2.1.1	Epipolar Plane Images	12
2.1.2	Hough Transformation	13
2.1.3	Circular Epipolar Plane Image Analysis	13
2.1.4	Robust Hough Transform Based 3D Reconstruction from Circular Light Fields	14
2.2	Aruco Markers	16
2.3	Otsu's Method	16
3	Ellipsoid based Trajectory determination	17
3.1	Creation of the Hough Spaces	18
3.2	Post processing and interpretation	19
3.3	Scaling and combination of Hough Spaces	21
3.3.1	ARuco Marker Attachment	22
4	Results	25
4.1	Real-World Dataset	25
4.2	Synthetic Dataset - DAVID	26
5	Discussion	39
5.1	Comparison to the original method	39
5.2	Interpretation of Real-World Dataset	40
5.2.1	ARuco-marker Attachment	40
5.2.2	Bird 4.2	40
5.2.3	Toad 4.3	41
5.2.4	Glitter Shoe 4.4	41
5.3	Quadrant decision Problem	41
5.3.1	Platforms out of thin air and strands of Hair	42
5.4	Further Complications	44
5.4.1	Usage of Global Thresholding	44
5.4.2	Data imbalance upon Hough space combination	44

Contents

5.4.3	Perspective consideration	44
6	Conclusion	45
7	Future work	47
7.1	Consideration of Perspective distortion	47
7.2	Availability and Parrallelization	48
7.3	Mesh, Color and Material	49

1 Introduction

This Thesis covers a method for 3D scanning of Objects with a focus on mixed material properties. We have based our work on a Hough transform to find trajectories in circular epipolar plane images proposed by Vianello *et al.* in 2018 [VADJ18]. We extended the use case to multi-angle data acquisition to gather more crucial data and to potentially improve accuracy. Furthermore, we propose strategies to keep calibration at a minimum and make the technique as accessible as possible.

1.1 Motivation

Three-dimensional depth reconstruction plays an important role in Computer Graphics. It's applications range from scientific purposes such as topological scanning or digital artifact preservance to consumer products such as games and movies.

However, a complex problem is the different material properties and how they distribute Light. Diffuse Object detection is already widely available with remarkable performance and low cost and can be easily implemented using Photogrammetry, the depth acquisition from images. Other materials proved harder to 3D-scan. These properties include specular, translucent and transparent objects, as the different light-paths have to be accounted for. Although many techniques have been proposed to solve these problems, they each have their advantages and drawbacks.

Of course, there is always the possibility to alter the reflactance functions of Objects. For example, if a car company wanted to scan their latest model, they would spray the car such that it would become diffuse, in order to make easier, more accurate scans. However, this doesn't make any of the other methods obsolete. There are objects you can not change the material properties for fear of not being able to reverse them, such as artefacts like ancient Egyptian gold statues.

1.2 Problem statement

It is possible to pinpoint the difficulty levels of 3d-scanning different material properties, diffuse objects as the easiest, specular or translucent in the middle range, and inhomogeneous Objects with several materials can be considered as one of the most difficult [IKL⁺10].

1 Introduction

Furthermore, different acquisition methods often highlight different material properties and their respective BRDFs (Bidirectional Reflectance Distribution Functions) and thus find themselves at different levels of difficulty in the taxonomy of object classes. The reconstruction of diffuse objects is a widely studied area and is often done using photogrammetry and stereo matching, where features of two or more Images get matched and triangulated to calculate their depth. However, when it comes to other materials that distort light and do not distribute it evenly, other acquisition approaches need to be considered. These could include shape from distortion, using calibrated light patterns and specular flow, see [IKL⁺10] for more details. However, most of the time they still have an adversity with mixed materials. This is why we set off to improve a method that can handle multiple materials, although some with drawbacks in quality. By extending it by the ability to introduce more data and multiple, arbitrary views, we hoped to minimise this lack of quality in order to produce more robust scans.

1.3 Organisation of the Thesis

As for the course of this Thesis, it is structured as following: Firstly, the Introduction 1 offers a motivation for this work as well as setting the problem scope. The Related Work 2 will further contextualize our method using other works in the field of Computer Vision. It will then provide a detailed Background section 2.1 setting the basis of our contributions. In chapter 3 we then imply our method with a thorough explanation of both theoretical and technical nature. Next, we will present some results of our method 4 which we will discuss in 5 in detail. Afterward, we will conclude our work 6, and finally take a peek into future research 7.3.

2 Related Work

Automated three-dimensional object and scene interpretation is an area of major interest in Computer Graphics. The acquisition is well-studied but far from complete, with many different approaches, each with its benefits and drawbacks. We aim to contextualize our work, and what role our contributions play in Computer Vision.

There are substantially different ways to determine the depth of scenes and objects. From soundwave-based systems like Sonar, touch-based systems, and light-interpretation methods. To keep our extent small, we will now dive deeper into only the latter. Light-based systems can be categorized into two: active and passive scanners.

Active scanners work by emitting a light signal and calculating depth based on that signal. This could include laser scanning systems and LiDar (light detection and ranging). These oftentimes refer to the Time-of-Flight (TOF) method. To gather depth information, the time difference between sending and receiving a light signal is used to determine the depth. This light signal can be either incoherent light based on a phase measurement [HSS06], or by utilizing an optical shutter technology. Yahav *et al.* used said shutter method to create a real-time depth-imaging camera [YIM07]. These systems proved to be robust to materials that reflect at least a small fraction of the sent light back to the sensor. So a scanned object must have at least some diffuse component, and can't consist of a mirror-like true specular surface.

To combat these type of materials, techniques like shape from distortion have been developed. Here a projected pattern is distorted by the specular surface, and this distortion can be interpreted as shape, with an early suggestion by Schultz [Sch94]. Similarly, other methods using calibrated light project a known light pattern on the scene of interest, and interpret the depth from there. See [RCM⁺01, SFSA12] for examples of structured light-based 3d scanners.

Passive scanners only interpret incoming light, such as ambient light and light reflected from the scene. This brings us to the field of photogrammetry, the process of obtaining 3D information of objects and scenes from photographs [MBM01]. Just like our human vision, stereo vision systems refer to two or more camera viewpoints and use their images disparity for 3d structure reconstruction, such as this lane detection stereo vision system [BB98]. The Foundation for Stereo Vision is Epipolar Geometry as discussed in [HZ04]. In general, these systems work by detecting features from different viewpoints and triangulating the image positions to gather depth information. This again creates the need for diffuse materials, as

2 Related Work

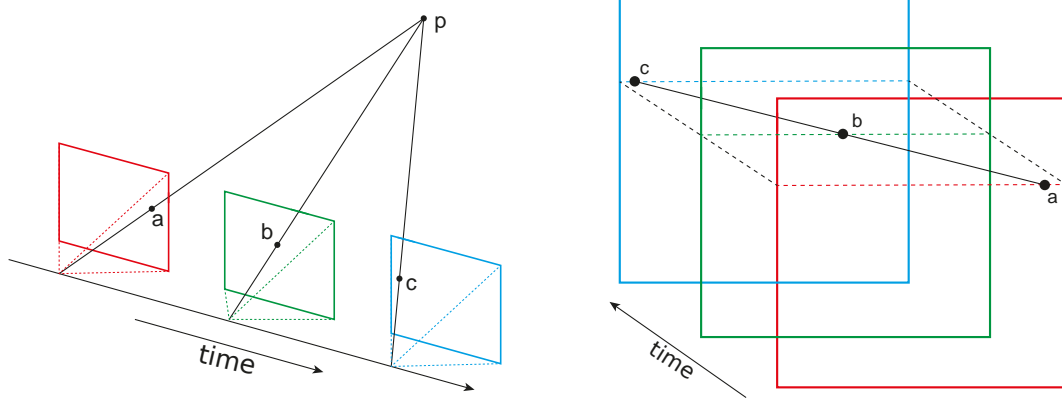


Figure 2.1: Epipolar plane Image formation:

On the left, a camera moving from left to right, the right shows how the point p appears on the epipolar image cube and the respective epipolar plane image.

specularities interfere with feature detection. We will now explore further a more feature independent approach using structure of motion, also relying on Epipolar geometry in the Background section 2.1.

2.1 Background

This section will cover the foundations of our work, which we built upon. We will investigate the necessary cornerstones, and will do so in a gradual manner.

Our method is a structure-from-motion based approach, particularly using epipolar plane image analysis to extract depth information.

2.1.1 Epipolar Plane Images

To further investigate the depth of scene points of their photogrammetric approach, Bolles *et al.* assumed the acquiring cameras motion to be known to focus on the scene reconstruction [BBM87] as oppose to camera location estimation. Specifically, they used a linear camera movement, to gather what they called epipolar plane images. Upon taking many images whilst moving the camera, an image block can be formed by concatenating these in a third dimension. One horizontal slice of such image block forms an epipolar plane image (EPI), as illustrated in figure 2.1. These movements create lines with particular slopes in the epipolar plane images, which can be interpreted as depth of scene points. However, it is not well defined what angle of a slope corresponds to a certain depth, but the relations between the slopes remain the same. This allows for the creation of depth disparity maps.

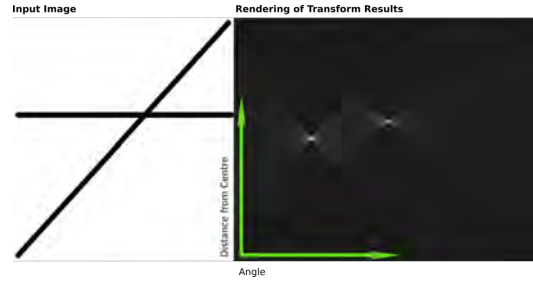


Figure 2.2: An example of a Hough transformation for line detection. On the left is the edge map, and on the right the corresponding parameter space or Hough space. By [Daf06].

To detect these lines and slopes in a robust and versatile manner, Vianello *et al.* introduced the use of a Hough transformation to create disparity maps of the scenes depth [VMDJ17].

2.1.2 Hough Transformation

A Hough transformation is a method for detecting lines, circles, or any other parametrizable geometric figures in images, patented in 1962 by Paul V. C. Hough [Hou62]. To accomplish this, a parameter space or Hough space is created, in which every point in the Hough space corresponds to a geometric figure in the image. This can be slope and offset for lines or center and radius for circles. For each point of the image's edge map, all possible parameters can be registered in the Hough Space, and in the end, the parameter space can be post-processed to search for local extrema that correspond to the existing geometric figures in the image.

An example of a Hough Transformation for line Detection can be seen in figure 2.2.

2.1.3 Circular Epipolar Plane Image Analysis

Usually, epipolar plane images refer to a 2D image with a horizontal and a time axis. They are acquired with a camera that takes multiple images while moving along a straight line. However, camera movement is of course not limited to linear paths. This is why Feldmann *et al.* extended the epipolar plane image analysis to circular camera movements, such as a camera moving around an object in an equidistant radius. Or, to simplify, a fixed camera and a rotating object. Instead of lines with particular slopes, they found the scene points to form sinusoidal curves in the epipolar plane images.

To illustrate, we will briefly go over the circular epipolar image cubes captured by orthographic cameras. The Image Cube Trajectory (ICT) for a scene point P can

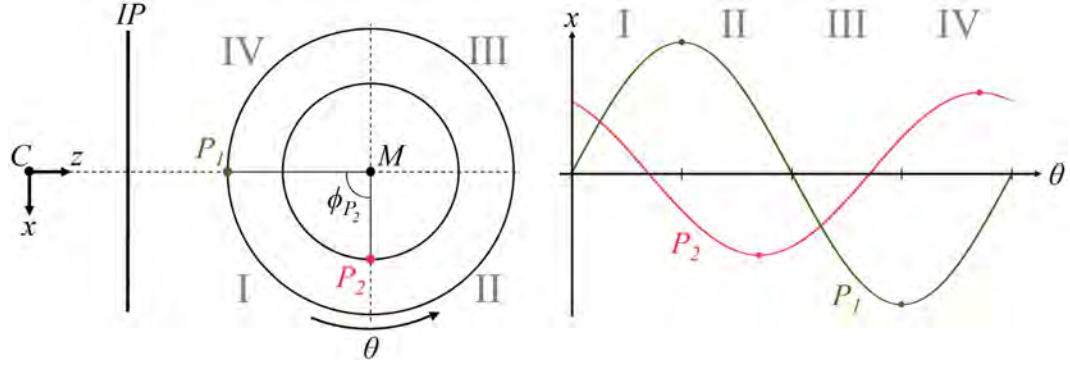


Figure 2.3: Left: Points in 3D space parametrized by radius and angle around the center of rotation M .
 Right: The respective trajectories of the points and their representation as sinusoidal waves in an epipolar plane image.
 Figure from [VADJ18].

be described as the sinusoidal function $X_P = R_P \cdot \sin(\phi + \Delta\phi_P)$, as can be seen in figure 2.3. Each of the N images in the image cube corresponds to an initial offset $\phi = 2\pi/N_i$. The scene point P has the radius R_P around the center axis of rotation, and a phase of $\Delta\phi_P$. Thus, it draws a curve with the amplitude R_P and the phase of $\Delta\phi_P$ in the epipolar plane image. Feldmann *et al.* split the path of trajectories into four quadrants, whereas points in the first and fourth quadrants would occlude points in the second and third. This in turn enables the localization of a point using its corresponding sine wave on an epipolar plane image.

In the case of perspective projections, these trajectories get distorted, and introduce an up-and-down movement rather than only side-to-side. Rather than focusing on these introduced complications, this work and its basis focused on the fact that a great part of a point's trajectory still lies on the easily parametrizable curve from the orthographic case. This is done with full knowledge of introducing flaws in the methods.

2.1.4 Robust Hough Transform Based 3D Reconstruction from Circular Light Fields

Building on their previous work, Vianello *et al.* used the extension to circular Light fields, to develop an according Hough transformation to extract depth information [VADJ18]. Their findings and approaches are the basis of our work which is why we will now explain the Robust Hough Transformation of Vianello *et al.* in greater detail, starting off with the rough idea and afterwards the technicalities of implementation.

Based upon the circular epipolar plane image analysis Feldmann *et al.* proposed as described in 2.1.3, a modified Hough transformation was used to automate

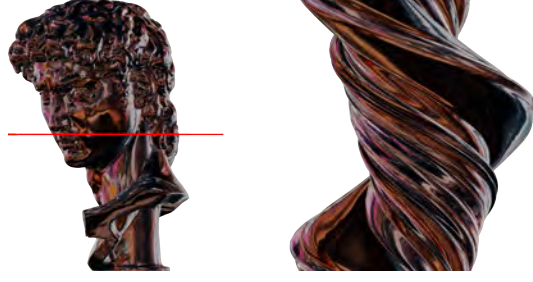


Figure 2.4: Circular epipolar plane image (EPI) construction.

On the right is an EPI of the horizontal red line on the left, with the axes (y, time), while the object rotates over time.

three-dimensional depth reconstruction. For the input, images of a full 360° rotation need to be acquired of an object or scene on a turntable, as viewed from the front. Each point of the object is now moving in a circle around the central axis of rotation. When viewed from the front this creates a non-linear movement from left to right. When interpreting the position of any point on an image, the possible trajectories one such point can lie on are now finite and can be parametrized using circular epipolar plane image analysis. Therefore, they can be translated into a parameter or Hough space, to find accumulations for determining the true trajectories and thus the initial angle and radius of the objects points.

The pixels in the circular epipolar plane images, the slices of the image block portraying the full rotation, are of the coordinates (x, θ) corresponding to the x-axis and the current rotation of the object. The Hough space has the form amplitude and phase offset (A, ϕ) to parameterize the sine waves drawn by the scene points. Vianello *et al.* split their Hough spaces into two, to account for occlusions. The first Hough space corresponds to quadrants I and IV and the second to II and III. As Feldmann *et al.* have shown, points in quadrants I and IV would occlude points in II and III.

Using a camera with a resolution of $N_x \times N_y$ pixels, and N images, both of these Hough spaces are of the shape $A \in [0, 1, \dots, N_x/2]$ and the phases $\phi \in [0, 2\pi/N, \dots, 2\pi]$ for each of the N_y horizontal slices. Conveniently the slices can be treated independently from each other, each with their two-dimensional EPI's and Hough spaces.

To populate these Hough spaces with data, they created a binary edge map for the epipolar plane Image, and checked every non-zero pixel. Every one of these pixels with their coordinates (x_i, θ_i) lies on a trajectory with the amplitude being bigger or as big as the distance from x_i to the images center x_c . So for every possible Amplitude A_i the corresponding phase offset ϕ can be calculated by:

$$\phi = \begin{cases} \arcsin(\frac{x_i - x_c}{A_i}) - \theta_i, & \text{if } \frac{\delta x}{\delta \theta} > 0 \\ \arccos(\frac{x_i - x_c}{A_i}) - \theta_i + \frac{\pi}{2}, & \text{otherwise.} \end{cases} \quad [\text{VADJ18}]$$

Where the first cell index refers to the first Hough-space, and the second the other.

2 Related Work

This determines the cells to be incremented for every possible amplitude with the Hough space coordinates (A_i, ϕ_i) . To further determine the parameters formed by trajectories of scene points, the Hough spaces have to be post-processed. The consideration of amplitudes bigger or as big as a points offset to the center, leads to higher cell values of greater amplitudes. Before correcting for this, low frequencies of the Hough spaces must be eliminated by subtracting its low-pass filtered version from it. Then, the result is re-weighted by a weighting matrix $W = e^{-0.001} \cdot [1, 2, \dots, A_{max}]^T$ to account for the bigger cell values in higher amplitude values. Finally, they are globally thresholded by using Otsu's method.

The resulting local maxima can then be derived to depth maps, using the occlusion rules proposed by Feldmann *et al.* After generating such a depth map for every horizontal epipolar plane image the object or scene can be reconstructed.

2.2 Aruco Markers

Garrido-Jurado *et al.* proposed a fiducial marker system in 2014 called ArUco markers (Augmented Reality University of Cordoba) [GJMSMCMJ14]. Its intended use was camera pose estimation for augmented reality and robotic vision, using a generative system of fiducial markers. Their algorithm allows for iterative generation of the smallest set of markers needed, while focusing on a high inter-marker distance in order to minimize inter-marker confusion. Also implied was a technique to detect such markers in Images.

2.3 Otsu's Method

In means to separate background and foreground in images, finding the appropriate thresholding value is of major importance in image Processing. Otsu introduced a method for automating this process using an images grayscale histogram back in 1979 [Ots79]. Optimally, an image's histogram would have two major peaks and a valley where the threshold would lie in the valley. However, most real images do not have such distinct value distribution. Otsu's method offers to maximize the separation of values nonetheless, setting apart background from foreground.

3 Ellipsoid based Trajectory determination

The Robust Hough Transform proposed by Vianello *et al.* is designed to consider viewing angles perpendicular to the rotational axis only. In particular it is unable to detect horizontal planes, as well as horizontal crevices obstructed by other parts of the object. The object reconstruction is limited to the parts the camera can represent. In order to overcome this issue, we propose a modified method to detect ellipses in epipolar plane image stacks to allow for imaging from arbitrary viewing angles. This on the one hand handles said issue, but on top enables us to combine multiple viewing angles to introduce more data into our Hough spaces and set the basis for increasing the accuracy of our reconstructions.

When an object is placed on a turntable, each point on the object draws a perfect horizontal circle in our scene with the passage of time. These scene points all rotate with different radii and different heights, but conveniently all with the same center, the rotational axis. It is well known that a circle in 3D space corresponds to an ellipse in image space after a camera projection [HZ04]. We can therefore detect these ellipses to determine the radius, the height and angle of the corresponding scene points. Since each point of an image could lie on a trajectory with the radius of its distance from the axis of rotation or greater, we have several but finite possible scene locations for each image point. In order to find the correct scene points we try to determine them using a Hough transformation.

We can parameterize an ellipse as the following:

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$$

which gives us the unit circle stretched by a on the x-axis and b on the y-axis. Here, we assumed the ellipses not to be tilted. We would either account for a tilted camera before processing or align the camera with the turn table.

In terms of orthographic projections, all circles with the same orientation are displayed as ellipses with the same major- to minor-axis ratio, the ellipse ratio $\epsilon = \frac{b}{a}$. For a frontal view of the turntable and the object, the ellipse ratio would be 0, as all the movement would remain on the horizontal line, moving only from left to right in image space. For a full top-down view, the ellipse ratio would be 1, as each

3 Ellipsoid based Trajectory determination

trajectory would draw a perfect circle.

As a way to reduce the complexity drastically, we focused on the orthographic projection and did not account for perspective projections, although we did run the risk of errors when acquiring images with perspective cameras. For further elaboration see 7.1.

3.1 Creation of the Hough Spaces

Each 3D point corresponds to a point (θ, r, z) in the parameter space coordinates. We therefore decided to create the Hough spaces in the shape of θ, a, z , where θ is the angle around the rotation axis, from zero to the length of our image set. r is the radius from each point to the central axis, and z is the height of the point, with a maximum being the images height. This creates a 3-dimensional Hough space.

Contrary to the original paper from Vianello *et al.* [VADJ18] we consider an image from the image set as a whole and do not split it up into epipolar plane images, since one slice of the image no longer corresponds to a horizontal slice of the scene. In the original case, each slice of a given height could be treated separately, but our approach is much more interdependent, and the entire 3D Hough space must be considered at all times. They also split the Hough spaces into two, one for the quadrants I and IV and one for II and III as seen in figure 2.3. We have chosen not to do this, as we interpret our data as polar coordinates and thus have no use for occlusion knowledge, as opposed to working with depth maps.

To populate the Hough space we create a binary edge map of every image of our image set, and process each edge pixel individually. For each edge pixel (u, v) we loop over every possible radius $a \in \{|a_{max} - u|, |a_{max} - u| + 1, \dots, a_{max}\}$ and with the ellipse ratio we can calculate the semi-minor axis $b = a \cdot \epsilon$. From this we get the x-coordinate on the ellipse $x = u - a_{max}$ and the y-coordinate by $y = \frac{b \cdot \sqrt{a^2 - x^2}}{a}$ after solving the ellipse parametrization for y. This gives us z_0 and z_1 , the center height of the two possible ellipses when subtracting or adding the y-coordinate from the v-coordinate. For the angle θ we need a y_{unit} value, which is the y value of the unit circle scaled by a for the given u value. That is, we calculate the value of θ if our ellipse were a perfect circle. This way, we don't have to stretch our ellipse afterwards. This is done simply by $y_{unit} = \sqrt{a^2 - x^2}$ and then we can take the arctan2 of x and y_{unit} to get the angle. Since each image of our set corresponds to a different angle ϕ of the object, we take it into account and subtract it from the calculated angle. And because we have a discrete natural Hough space, θ, a and z must all be rounded to correspond to natural numbers. For θ in particular, we have to stretch 360° to the number of images, which is the full extent of our hough space axis. Once we have the two sets of θ_0, a, z_0 and θ_1, a, z_1 we can increment the Hough space at both locations.

Since we don't know which direction a point is traveling on its trajectory, i.e. whether it is in Quadrants I and IV or in II and III, we decided to assume both cases. This means that we increment the cells as if it hit both trajectories. For a visually aided explanation, visit Figure 3.1. We also went into greater detail about this complication in the discussion part 5.3.

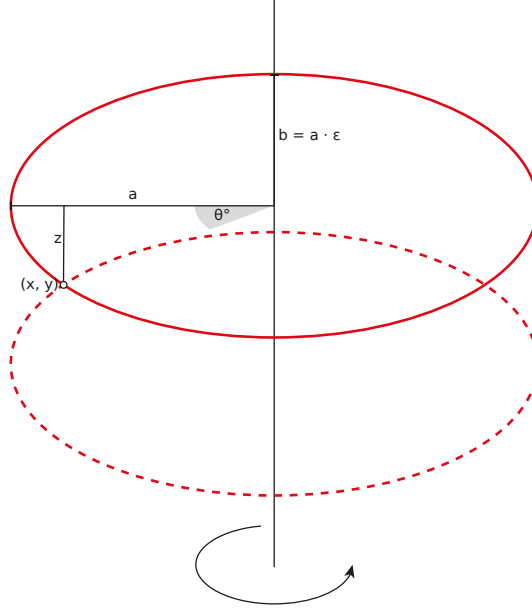


Figure 3.1: The two possible ellipses hitting the point (x,y) with the radius a . The ellipses respect the ellipse ratio with the minor-axis being $b = a \cdot \epsilon$ and meet the point at the angle θ . z is the distance in height from (x,y) to the center of the ellipse. This shows all three of our parameters, the coordinate system of our Hough spaces, angle, radius, and height.

An important side note is the consideration of negative and positive angles. Upon inspection of figure 3.2 it becomes clear, that a point's rotation trajectory in 3D space changes its elliptical direction in image space between viewing from the top or from the bottom. When implementing our method, we focused on top-view angles. To further account for the change of trajectory direction, simply reversing the order of our images was a sufficient solution to simulate the same direction of movement.

3.2 Post processing and interpretation

To further interpret the data, we need to post-process the Hough space. We decided to post-process all layers of the Hough space at once, to account for the dependencies between the points. As in Vianello *et al.* [VADJ18] we remove the low frequencies, but using a three-dimensional Fast Fourier transform to create a true low-pass

3 Ellipsoid based Trajectory determination



Figure 3.2: This figure shows the difference between top-down and bottom-up views. Although both views show a point in 3D space rotating in the same direction, the point moves anti-clockwise in the left and clockwise in the right view on their respective ellipsoidal trajectories.

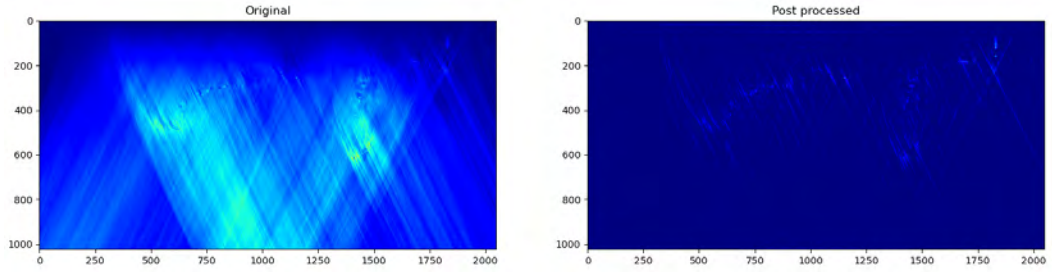


Figure 3.3: A Hough space layer before and after the post-processing step. This Hough space was created using the 120° view of the marble orthographic DAVID Dataset 4.8.

filter we can subtract from our hough space. Then again, we are performing a wheighting over the radius axis r to reduce higher amplitudes. The exponential function $e^{-0.002 \cdot [1, 2, \dots, r_{max}]}$ has been chosen. We do this because points always consider radii of their amplitude and higher, thus higher amplitudes get reasoned for more often. We then thresholded the Hough space using Otsu's Method to get a clearer distinction between object Points and non-object points. Finally, we found local maxima with a small minimum distance. These extrema of θ, a, z are polar coordinates that can directly be interpreted as 3d points of a point cloud. We then used the values of the extrema to use as certainty values. This allows further possibilities down the line such as another manual threshold. Figure 3.3 shows the post-processing of a Hough space layer.

3.3 Scaling and combination of Hough Spaces

The way we chose the shape of our Hough spaces to be, our scaling is purely based on the image size and distortion our particular camera position and angle views the object. For a 2000x800 pixel camera image, this would result in a reconstruction with a maximum height of 800, as well as a maximum radius of 1000. In addition, the different camera angles may distort the reconstruction by squashing, stretching and translating. To overcome this, we convert the Hough spaces and their coordinates into world coordinates. This allows us to get an accurate representation of our object, but moreover, this makes combining Hough spaces taken from different angles trivial.

In order to get the correct transformation parameters from Hough space coordinates to world coordinates, we need to determine the possible factors: The height 0 is not fixed for a Hough space, as the location of the object in the images captured determines the location on the rotation axis. This means we have to take a height-translation value into account. The height is also in pixels, so a height-scale factor is needed as well. Similarly, the radius is also in pixels, which is why we need a radius scale too. A radius translation value is not feasible, as the points have a fixed relationship to the center turn axis, and can not move away from it, nor can they move closer. For θ , the number of images taken determines their maximum value and the step size can thus not be changed. but of course, the camera might view the object from a different angle than which we would want to consider, so an angle translation is needed at last.

The advantage of combining multiple Hough spaces from different angles is, of course, more data. Depending on the scanned object itself and its material properties, the camera setup, and the number of images and their resolution, the Hough spaces may lack accuracy. Combining different angled views can potentially fill gaps, and better the certainty for uncertain areas, such as crevices. Of course, the processed point clouds could be merged together, but this would result in the same artifact as the Hough spaces have on their own, so the goal is to combine them before the post-processing stage. This way they can be processed together and will lead to better, more accurate results, by reinforcing true peaks and weakening artifacts.

To merge the hough spaces, we considered several approaches:

Iterative closest point is an algorithm, that takes two point clouds and merges the source point cloud into the target point cloud [BM92]. This is done by taking the closest point from the source to the target, for each source point, and then calculating the transformation to minimize the distances between these points. Then the closest points are recomputed, and the source point cloud iteratively moves closer to the target. Although this is a well-known algorithm for this task, a few problems arose for our scope. Firstly, scaling factors are not part of the default ICP, but there are

3 Ellipsoid based Trajectory determination

versions of the algorithm that do include scaling. On the other hand, we have nailed down only four specific transformation factors, so a custom ICP algorithm would be needed nonetheless. The main problem, however, is that it is highly dependent on the quality of our point clouds. Noise can easily mess up the transformation, and reducing this noise would be a reason to combine multiple Hough spaces in the first place. And in the end, even if we had the perfect transformation values, our combined hough spaces would only be the size of our possibly already distorted target point cloud and not the world coordinates.

We also considered using optimization of a loss function which is essentially minimizing the squared distances between the points as changing our four parameters. But again, this is highly dependent on the quality of the point clouds and still not in the world coordinate system. Also, both of them have the problem of having to post-process the Hough spaces, get the transformation values, and then fuse the non-processed Hough spaces to post-process them together.

So we decided to take a different approach, which is quality independent, and the transformation values can get calculated before the Hough spaces even exist. By knowing the exact Hough space coordinates of two known points in an Image taken from which we know its theta base value, we can compute the transformation values. Of course, this is not trivial, as we increase multiple cells in the Hough space for each point. But if we know of either the radius of the trajectory that this particular point draws in image coordinates or the θ value at which this point would end up, we could determine the exact location our point would be in Hough space coordinates. Looking at figure 3.1 we can solve for two of the three parameters if one of them is known. If we know the θ value, there is only one possible radius and this ellipses height this point could be on, thus being able to solve the equation from the hough space generation for the radius and the height. The other option is to use the radius in pixels of the point's trajectory. If the point is not the outermost point of the ellipse, there are actually two possible ellipses. But once we assume that we know if the center of the ellipse lies below or above the point, this gives us only one location. For example, we can assume that the point we see is facing toward the camera, and we look at it from slightly above, so in turn, the center of this ellipse lies above the point. From there we can use the same calculation from our Hough space generation to calculate the Hough space coordinates. Regarding that this would only need to be done for two points for each camera angle, this could also be done manually.

3.3.1 ARuco Marker Attachment

In order to determine the real-world scale of our scanned objects, we created an attachment for the turntable we were using. This attachment is a 3D-printed extension

3.3 *Scaling and combination of Hough Spaces*

of the turntable with ARuco markers on the side as can be seen in our real-world dataset 4.1. Using these readily available image features, we can pinpoint their locations with pixel accuracy. We have designed the turntable attachment in such a way, that we would know the exact locations of these markers so that we enabled accurate scaling to the real-world coordinate scope.

Furthermore, we were able to use the ARUco edge points, to fit an initial ellipse, which gave us three things: The ellipse ratio ϵ , the center point of the ellipses and therefore the location of the central axis of rotation, and the radius of the ellipse which hits a point (u, v) , an ARuco edge pixel. Using this known radius, and the same calculation method from our Hough space generation, we can determine exactly, where this one point corresponds to in the Hough space. Once we have two different points with different heights, each with its true coordinates and the Hough space coordinates, we can calculate the scaling and translation factors. From there, we can scale our object directly to its true size. In addition, this allows us to easily combine multiple angle views and Hough spaces. By using multiple distinct markers, it allowed us to acquire the true coordinates with radius, angle, and height for any of the markers. This way, multiple scans can be merged together, without them initially seeing the same side of the object.

As a sidenote, this marker attachment also allowed us to calculate the initial tilt of the cameras, so that we could pre-process the images to account for that.

4 Results

We will now present the resulting scans using our method. For demonstration purposes, we have used our algorithm on both real-world data, as well as a synthetic dataset.

4.1 Real-World Dataset

For our real-life dataset, we provide three scanned objects of varying materials. All of them have been acquired using a lens of 100mm focal length, the resolution of the images being cut to the region of interest ranging between 2600x1100 pixels and 4096x2600 pixels. We placed them on our turntable extension with ARuco markers, as previously presented in 3.3.1 to enable the acquisition of ellipse ratio, the center of rotation, a potential tilt of the images as well as the necessary parameters for scaling of the results. For each of the object's camera angles, we took 1440 images, one for each quarter of a degree of the 360° rotation. These were then pre-processed in order to rotate the images to account for the tilt we introduced during setup. The tilt angle was again determined using the same method of fitting an ellipse to the ARuco markers on our attachment. The setup can be seen in figure 4.1.

Our three Objects are a toad, a glittering shoe and a figurine of a cartoonish bird. The bird is made of ceramics and it has a matte, diffuse appearance. It does not have many hard edges and has a very homogenous surface. Figure 4.2. The toad is made of glazed ceramics and is thus highly specular, reflecting the many



Figure 4.1: The setup for our real world data aquisition. Here with our glazed ceramic toad, sitting on our ARuco marker attachment of the turntable below.

4 Results

lights of our scene harshly. Figure 4.3.

And lastly, the glittering shoe is partly made of very sparkly materials, partly smooth specular and partly very homogenous areas, which makes it a highly complicated mix of materials. Figure 4.4.

4.2 Synthetic Dataset - DAVID

For our computer generated dataset, we used a bust of Michelangelo’s David, created by Thomas Flynn [noab], and rendered all images using Blender [Fou].

All size units in this Dataset are measured in meters. The bust is 1.99985 meters tall and was captured using three different materials. A diffuse marble texture, a highly specular chrome texture as well as a transparent glass-like texture, although the glass texture is not a true volumetric render.

For each material, we rendered images from three different angles: 60°, 90° and 120°, from slightly above, horizontally and from slightly below the object. We did this for both an orthographic camera, as well as a camera with a focal length of 100mm, the same focal length we were able to use for our real-world data. We settled on taking 1440 images for each full rotation cycle of our object, which works out to one frame per quarter of a degree.

On top, we rendered the 60° angle with shorter focal lengths to show its effect in 4.13 using the marble texture.

We then computed point clouds using our ellipsoid-based method for all angles, as well as all combinations of angles.

For comparison, we also computed point clouds using the original method of Vianello *et al.*, each labelled as EPI with quadrant separation and without quadrant separation. As they split their Hough spaces into two, one for quadrants I and IV and one for II and III, we post-processed them separately from each other once and merged them afterward, and because they should convey the same data, also once without the quadrant separation, post-processing them together.

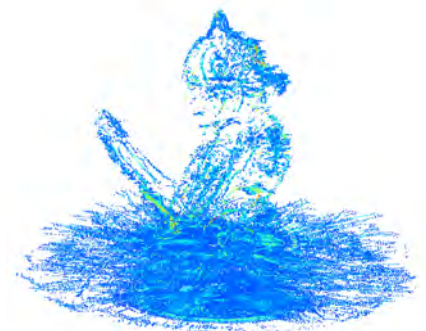
Both the Hough spaces created by our method and the ones using EPI’s were post-processed using the same function with a 3D Fourier transformation to keep comparison consistent. All were scaled to the true scale using our method, which was slightly modified for the EPI-scans to fit their requirements. For plotting, we used Cloudcompare [noaa] to measure the distances from each point to the ground truth mesh. The point clouds were colorized using a normalized colormap from blue through green and yellow to red, where red is at the 0.2m mark, or 20cm (10% of the model height).



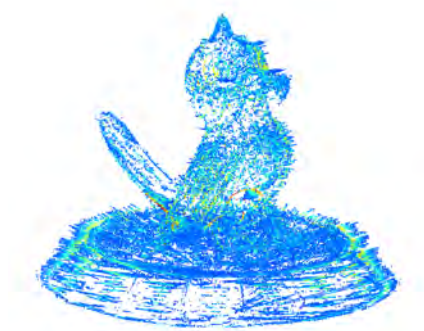
Origin Image Angle 1



Origin Image angle 2



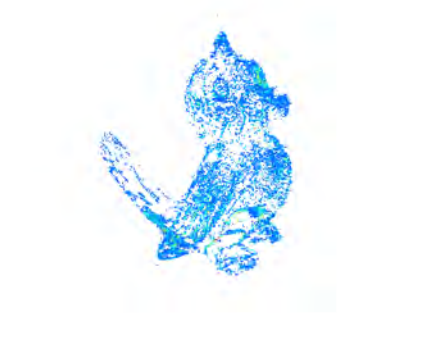
Computed using angle 1



Computed using angle 2



Computed using both angles



Computed using both angles,
only points above marker attachment.



Low Certainty

High Certainty

Figure 4.2: **Real-World Dataset - Bird**

Different computational stages of our bird model, colored according to their normalized certainty values using the color map shown.

4 Results



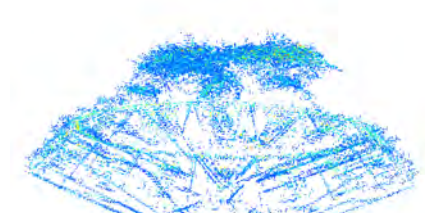
Origin Image Angle 1



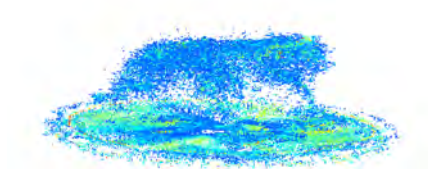
Origin Image angle 2



Computed using angle 1



Computed using angle 2



Computed using both angles



Computed using both angles,
only points above marker attachment.



Low Certainty

High Certainty

Figure 4.3: **Real-World Dataset - Toad**

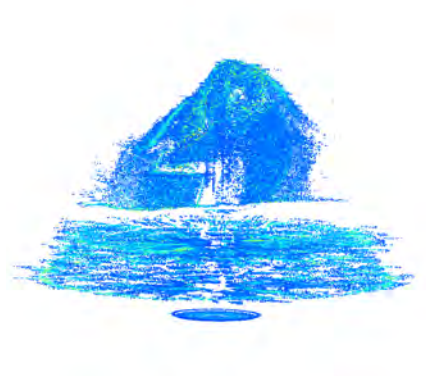
Different computational stages of our toad model, colored according to their normalized certainty values using the color map shown.



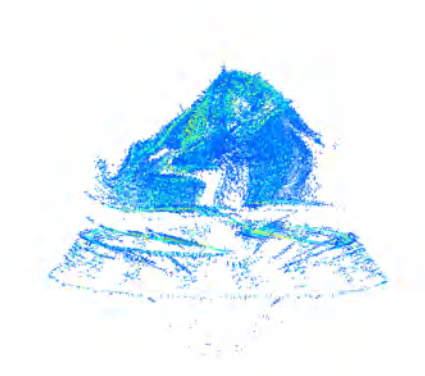
Origin Image Angle 1



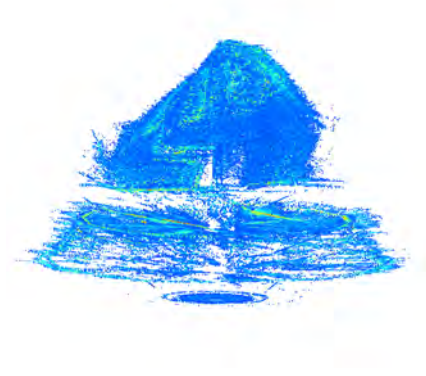
Origin Image angle 2



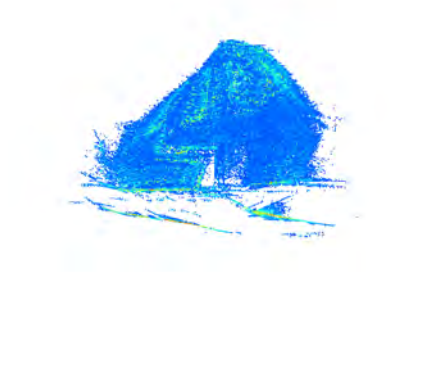
Computed using angle 1



Computed using angle 2



Computed using both angles



Computed using both angles,
only points above marker attachment.



Low Certainty

High Certainty

Figure 4.4: **Real-World Dataset - Glitter Shoe**

Different computational stages of our glitter shoe model, colored according to their normalized certainty values using the color map shown.

4 Results

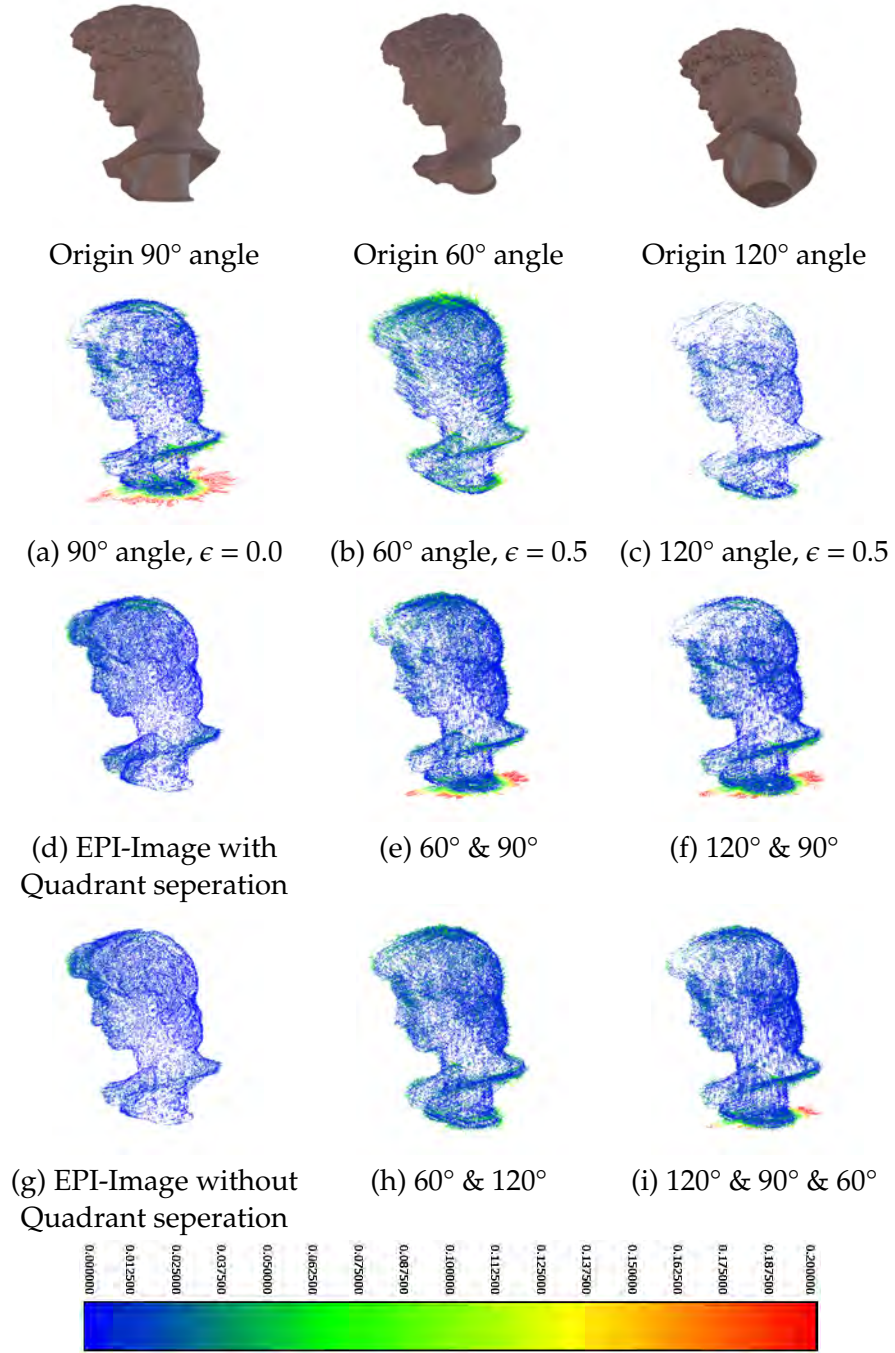


Figure 4.5: **MARBLE - 100mm focal length**

(a), (b), (c) are point clouds computed with our method using only one respective angle.

(e), (f), (h), (i), are generated using our method with multiple combined angles before post-processing the Hough spaces.

(d) and (g) are created using the original method by Vianello *et al.* [VADJ18]. (d) with post-processing the Hough spaces for each pair of quadrants separately, and (g) with post-processing them together.

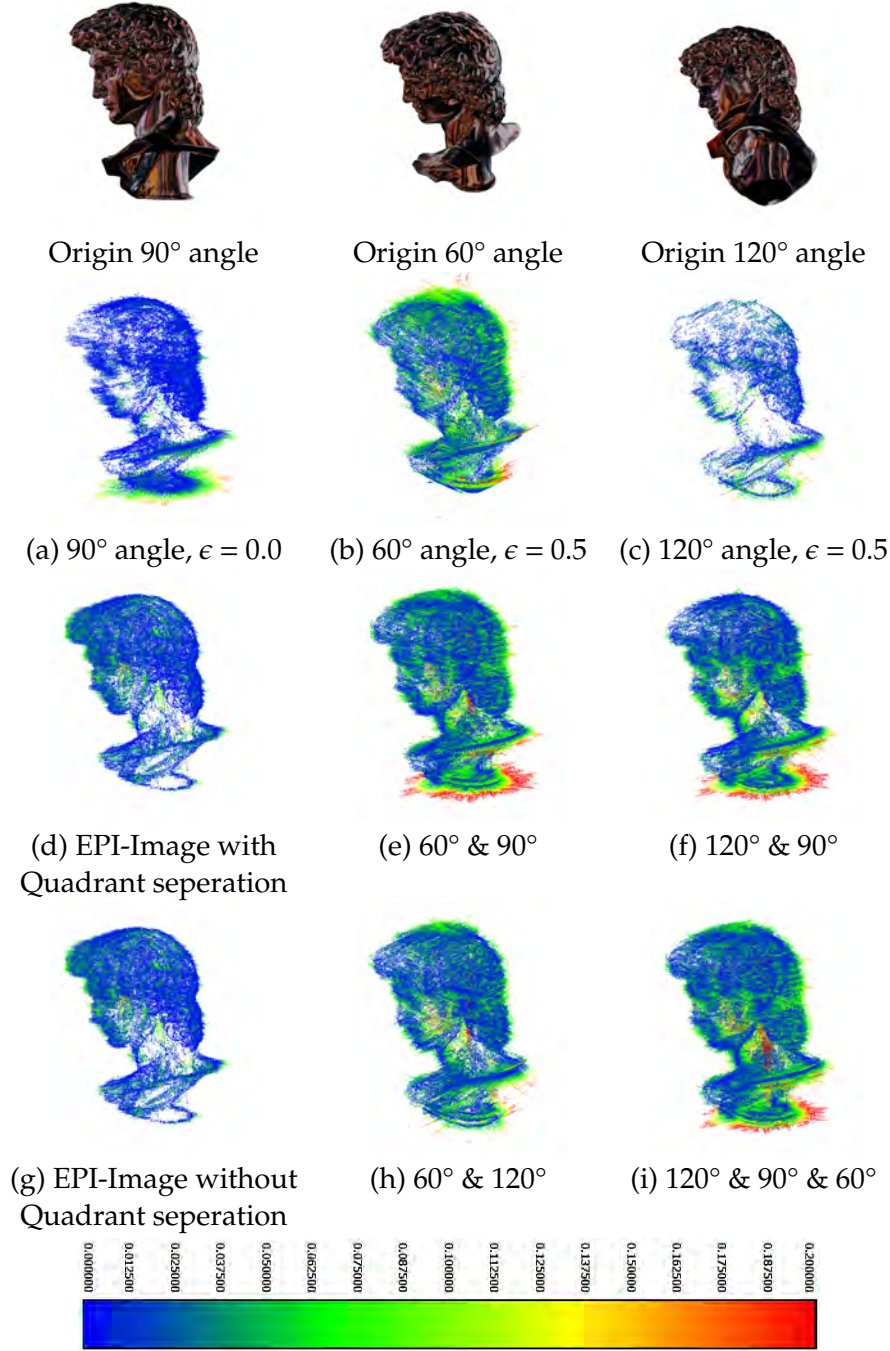


Figure 4.6: **CHROME - 100mm focal length**

(a), (b), (c) are point clouds computed with our method using only one respective angle.

(e), (f), (h), (i), are generated using our method with multiple combined angles before post-processing the Hough spaces.

(d) and (g) are created using the original method by Vianello *et al.* [VADJ18]. (d) with post-processing the Hough spaces for each pair of quadrants separately, and (g) with post-processing them together.

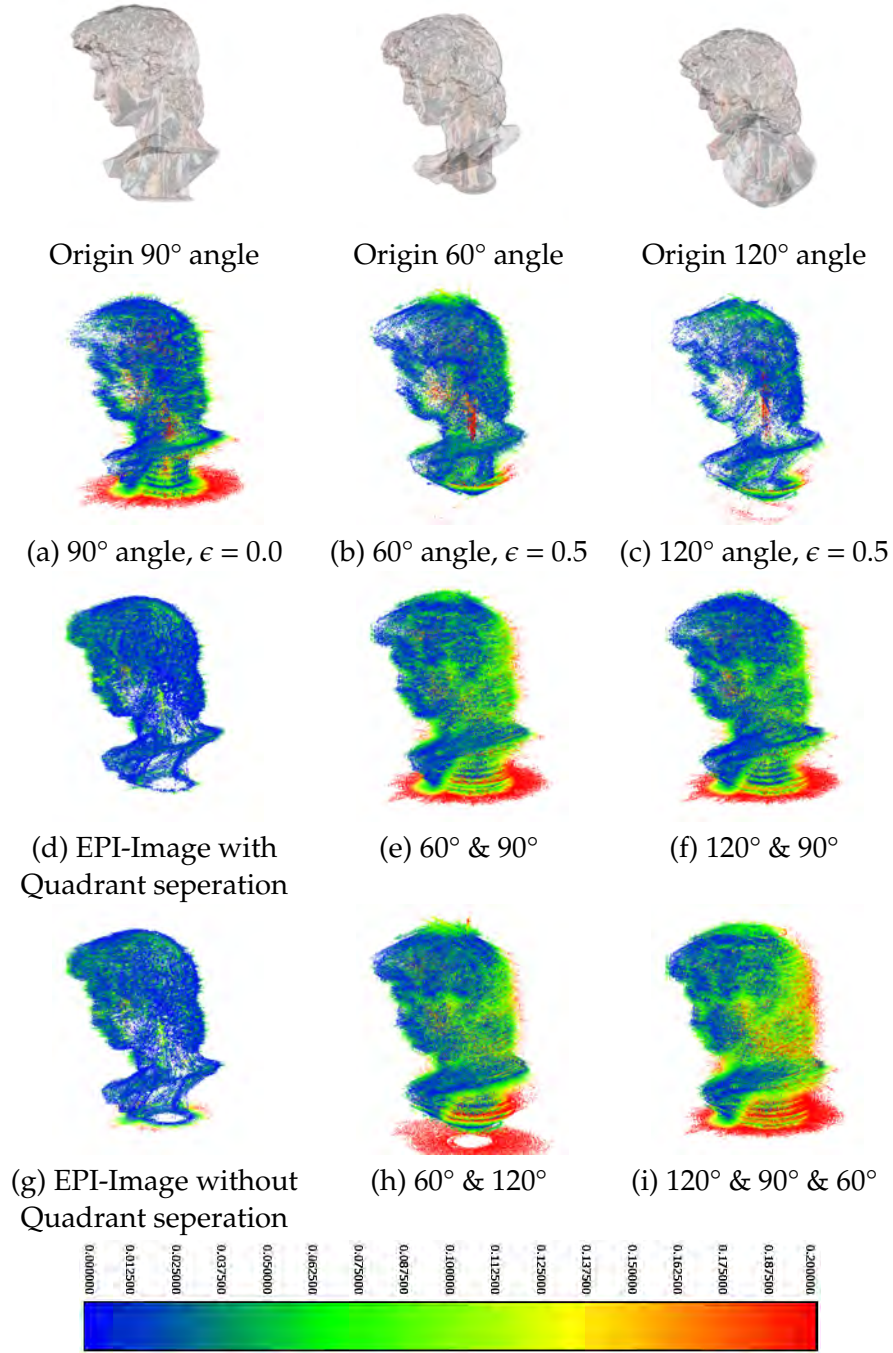


Figure 4.7: **GLASS - 100mm focal length**

(a), (b), (c) are point clouds computed with our method using only one respective angle.

(e), (f), (h), (i), are generated using our method with multiple combined angles before post-processing the Hough spaces.

(d) and (g) are created using the original method by Vianello *et al.* [VADJ18]. (d) with post-processing the Hough spaces for each pair of quadrants separately, and (g) with post-processing them together.

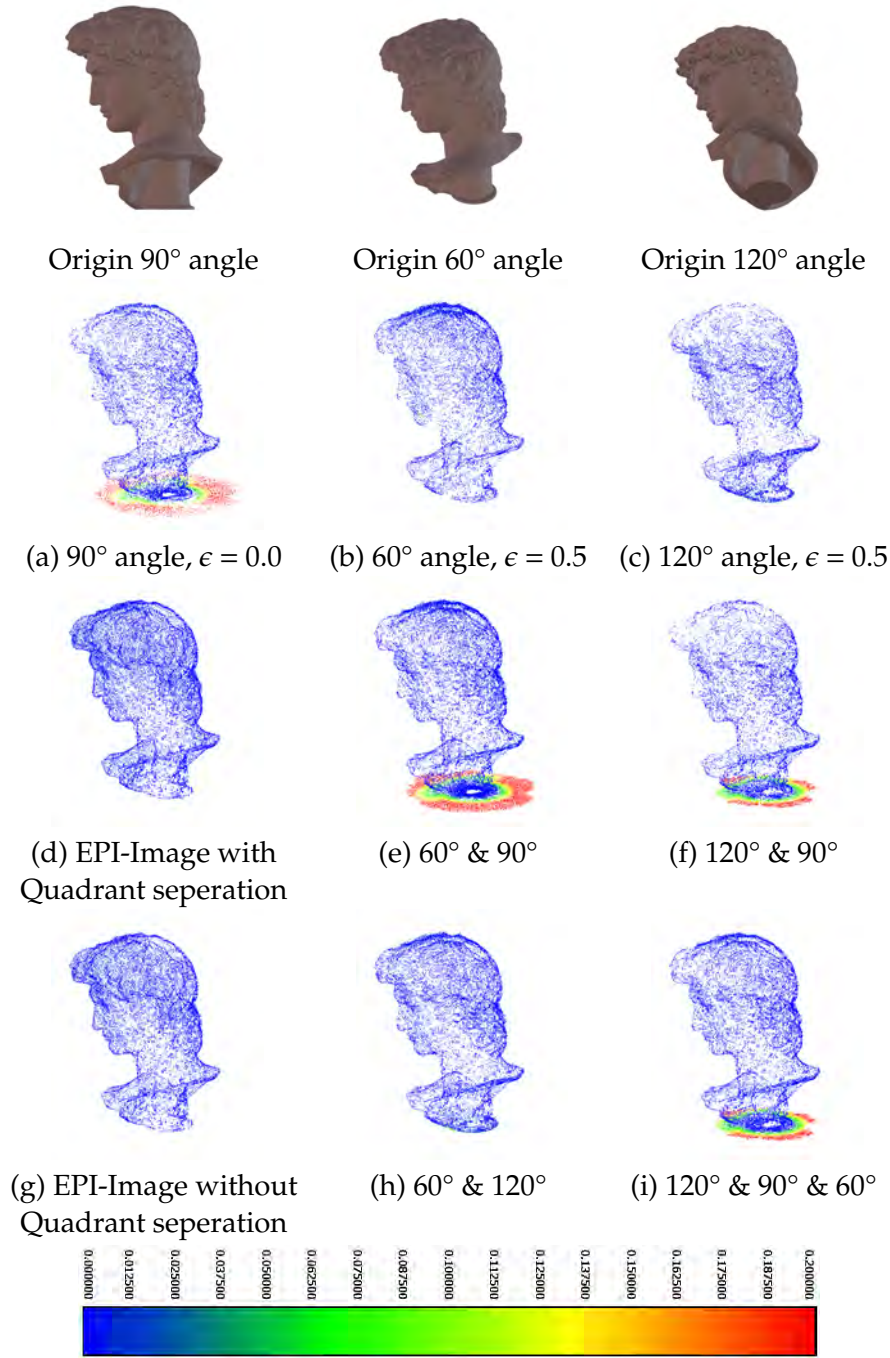


Figure 4.8: **MARBLE - ORTHOGRAPHIC**

(a), (b), (c) are point clouds computed with our method using only one respective angle.

(e), (f), (h), (i), are generated using our method with multiple combined angles before post-processing the Hough spaces.

(d) and (g) are created using the original method by Vianello *et al.* [VADJ18]. (d) with post-processing the Hough spaces for each pair of quadrants separately, and (g) with post-processing them together.

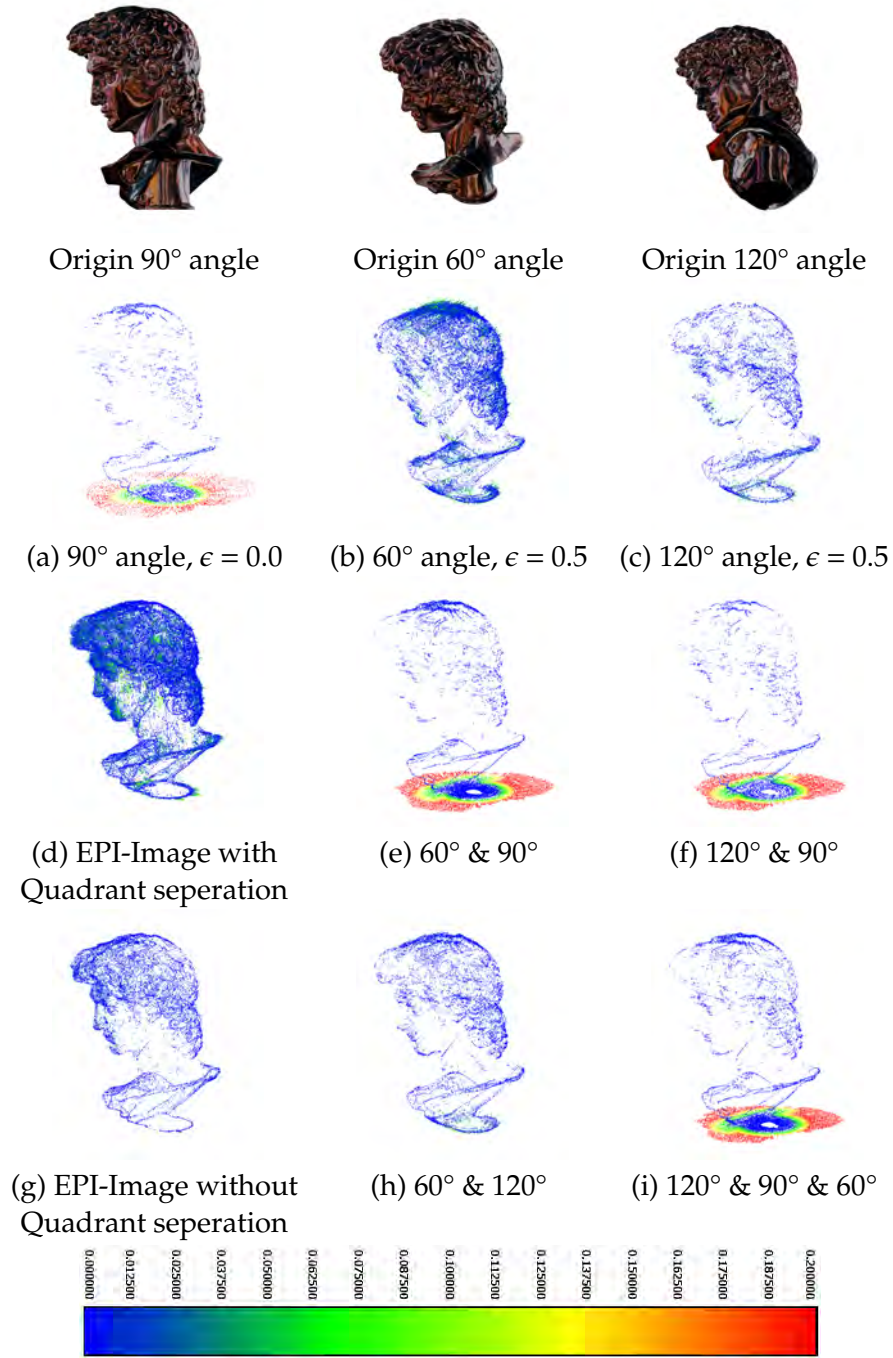


Figure 4.9: **CHROME - ORTHOGRAPHIC**

(a), (b), (c) are point clouds computed with our method using only one respective angle.

(e), (f), (h), (i), are generated using our method with multiple combined angles before post-processing the Hough spaces.

(d) and (g) are created using the original method by Vianello *et al.* [VADJ18]. (d) with post-processing the Hough spaces for each pair of quadrants separately, and (g) with post-processing them together.

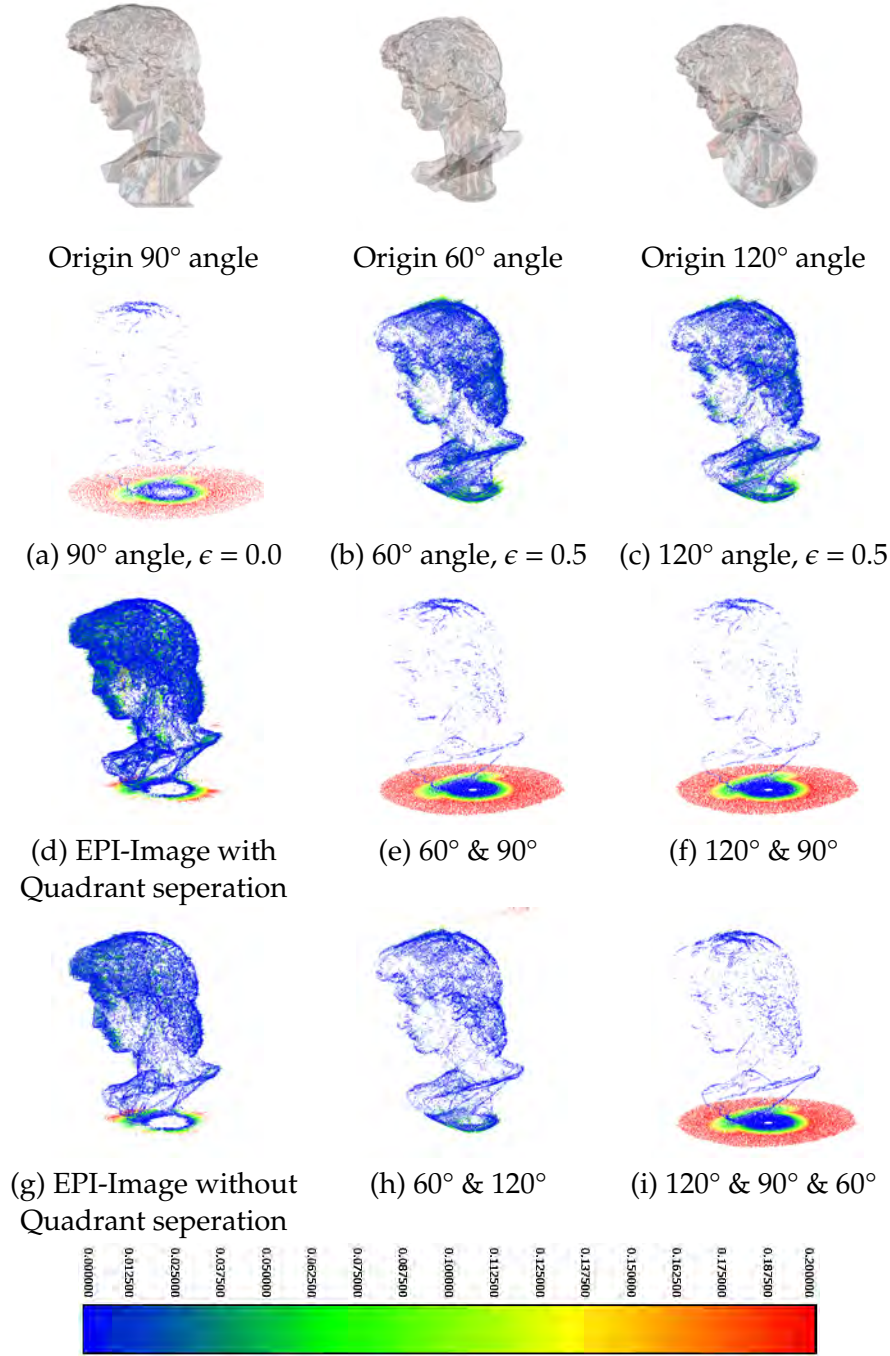


Figure 4.10: **GLASS - ORTHOGRAPHIC**

(a), (b), (c) are point clouds computed with our method using only one respective angle.

(e), (f), (h), (i), are generated using our method with multiple combined angles before post-processing the Hough spaces.

(d) and (g) are created using the original method by Vianello *et al.* [VADJ18]. (d) with post-processing the Hough spaces for each pair of quadrants separately, and (g) with post-processing them together.

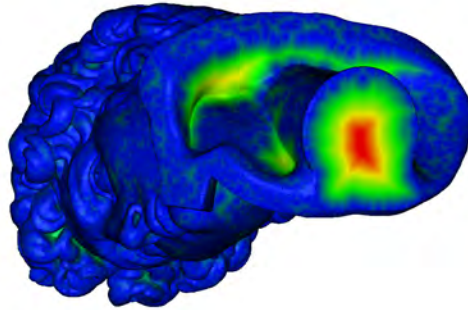
4 Results

0	90°
1	60°
2	120°
3	90° & 60°
4	90° & 120°
5	60° & 120°
6	90° & 60° & 120°
7	EPI with quadrant seperation
8	EPI without quadrant seperation

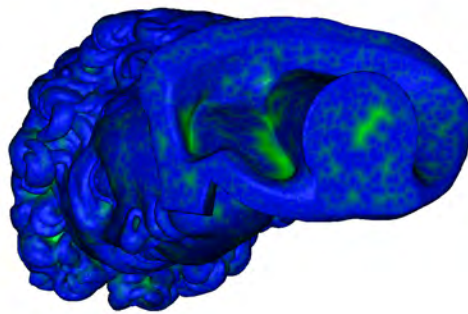
Means of Distance to Ground Truth

	Marble Orthographic	Chrome Orthographic	Glass Orthographic	Marble Perspective	Chrome Perspective	Glass Perspective
0	0.0338728	0.088926	0.176357	0.0248498	0.0305519	0.0725089
1	0.00261992	0.0119075	0.013894	0.023295	0.0366079	0.0464326
2	0.00217203	0.00783132	0.0139366	0.0147775	0.0240503	0.0368101
3	0.039316	0.0865578	0.167261	0.0189082	0.0443866	0.0874267
4	0.0236415	0.10398	0.164498	0.0186545	0.0401465	0.0873772
5	0.00230234	0.00523249	0.010343	0.0178176	0.0344586	0.0857175
6	0.0193411	0.0708787	0.137615	0.0160169	0.0575308	0.0922838
7	0.00261252	0.0141432	0.0169215	0.0115637	0.0206778	0.0239088
8	0.00248278	0.00516361	0.0147971	0.00982329	0.0200945	0.0267051

Figure 4.11: This table shows the mean distances from the point clouds to the ground truth from our computer generated DAVID Dataset. The bold printed values are the minima for each material scan.



Mesh to cloud distance for Marble Orthographic EPI scan with full front-view



Mesh to cloud distance for Marble Orthographic Ellipse based scan of 60° and 120° angle combined. Bottom and Top View.

Figure 4.12: This figure shows the coverage of the point clouds on the ground truth mesh using mesh to cloud distances. As the upper figure was generated using images taken from the front, the crevices under the shoulders and the bottom of the base were never visible on the processed images. The figure below was created using images taken from above and below, thus being able to fill in spots otherwise not visible. The original point clouds can be seen in figure 4.8

4 Results

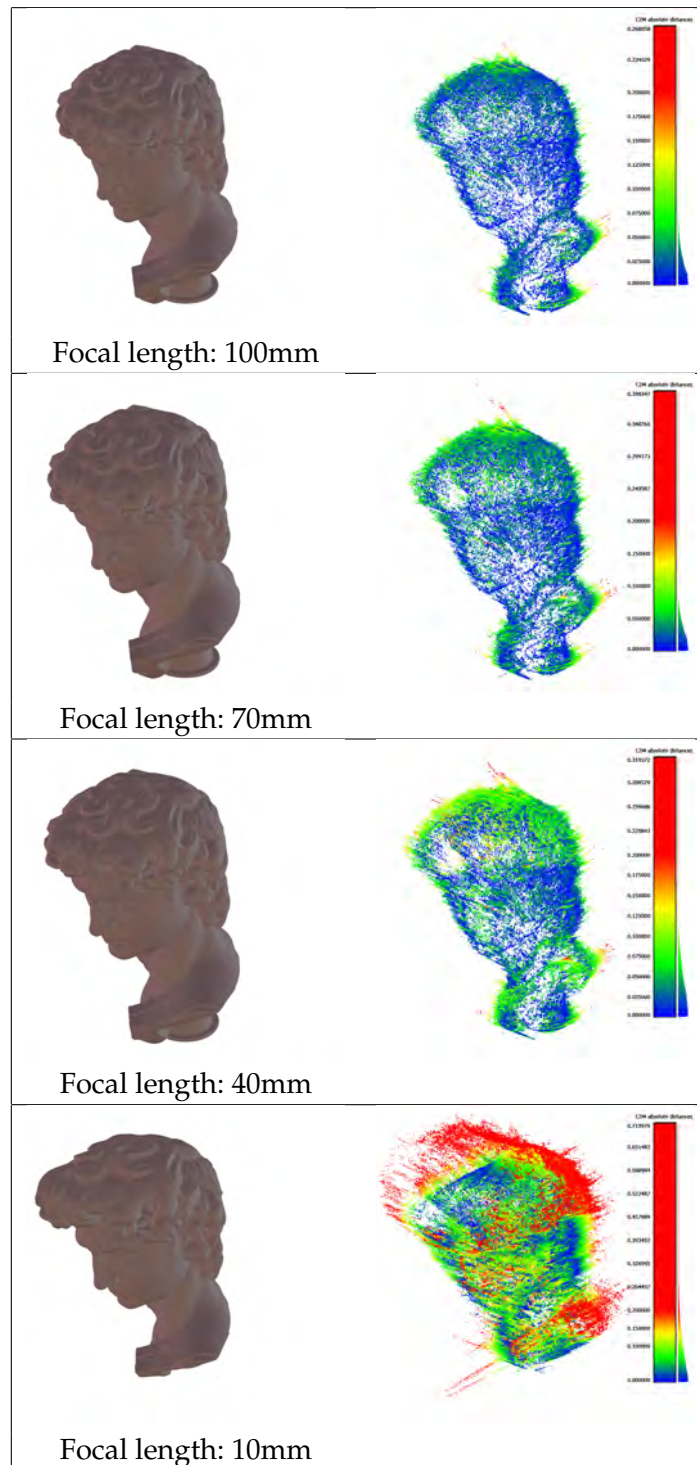


Figure 4.13: This shows the results of using different focal lengths.

5 Discussion

In order to create a flexible adaptation of the robust Hough transform based 3D reconstruction algorithm from Vianello *et al.*, we have modified the method to introduce arbitrary camera angle image acquisition. Our goal was to take the basis of their method and enable an omnidirectional scan that fuses different views of an object or scene. On top of that, our method allows more versatile setups, and we introduce techniques to keep calibration to a minimum.

5.1 Comparison to the original method

To put the results of our work into context, we will now compare our results to the original method by Vianello *et al.*. We will refer to the synthetic dataset DAVID 4.2, where we plotted the results of both methods together.

Upon general inspection of the error plots, our method seems to perform similarly to the original method, but more often than not, the original method outperforms our model in terms of overall quality and homogeneous distribution of points. We will later elaborate on why that might be the case. However, both are able to handle multiple varying material properties to a some extent.

As expected, the diffuse marble texture yields the best results, but the chrome and even the glass texture can be captured with close to acceptable error rates. As we assume orthographic projection for both orthographic and perspective images, it is only sensible that the orthographic scans are of better quality.

To emphasize our findings, we computed the mean distances from points to the ground truth for each of our Point clouds in the DAVID dataset 4.11. In 4 out of 6 cases, the original method has a smaller mean than any of our scans of multiple and mixed-angled views. Marble orthographic seems to be better represented by the 120° view, although being more sparsely sampled. In the orthographic glass case, the combined 120° and 60° scan is a more robust scan, both in terms of the mean value and to the naked eye.

One particular area where our method produces better results is of course anything that can not be seen from the front. This can be either flat, horizontal regions, or crevices in which the camera would have no visual. In figure 4.12 we show the

consequences with the point cloud coverage of the mesh. Of course, the original method struggles with said regions, whilst ours manages to fill in these gaps.

Very noticeable is also the importance of a longer focal length in figure 4.13. Although expected, it becomes very clear that a lower focal length, and thus further away from an orthographic projection yields weaker results. The shape is still clearly visible for a 40mm focal length and one could argue even for the 10mm. Nonetheless, the longer the focal length, the better the quality of the point cloud.

5.2 Interpretation of Real-World Dataset

For a real application test, we set out to create scans of three different objects of varying materials of increasing difficulty.

5.2.1 ARuco-marker Attachment

Although our ARuco-marker attachment was far from being perfectly calibrated, we managed to make it work by using multiple images ARuco-marker locations for our ellipse fitting. This enabled a fully automated pipeline without any further user input or parameter adjustments. By using this attachment, we were not only able to find the premature tilt of our cameras, the center of rotation, and the ellipse ratio we needed for the Hough space generation, but also enabled us to match these points with the real-life coordinates to find the transformation vectors necessary for scaling to the world coordinate system. Furthermore, this set the basis for combining multiple views.

Due to this attachment, our image acquisition process required no further attention. We were able to set up the camera anywhere, any angle, just the object and the attachment being in frame, the rest is taken care of by the automatic processing. This made for a really simple acquisition process.

5.2.2 Bird 4.2

Although the bird is of diffuse material, it poses a unique challenge: its surface is very smooth and homogenous, resulting in few hard edges and points. As all data is gained from edge images of the original photographs, this surface property hardens the scanning process. Nevertheless, credible results have been obtained using both angles on their own and in combination.

5.2.3 Toad 4.3

The toad which made up of glazed ceramic produces slightly noisier scans, but still very good results. It lacks detail especially on the underside, this is due to the high reflectance on the top and with it a high frequency part of the image, which creates the need for a edge-map threshold that isn't able to account for both levels of detail. This however is much more pronounced in the case of the glitter shoe 5.2.4.

5.2.4 Glitter Shoe 4.4

The glitter shoe is the most challenging of our three objects. Not only do its sides sparkle at the smallest movement, but there is also a big shiny part and large homogeneous areas on it. In regards of these factors, our system did surprisingly well. Not to oversee however, is that only the middle part of the shoe is represented by the point cloud. Because the glitter creates so many points in the edge map, the front and back of the shoe, both being fairly homogeneous areas, are not able to keep up with the frequency of cell increases in the Hough space. This leads to a filtering out of the lower frequencies in the front and back. Coupled with the lower weighting of higher radii, this effect is amplified even more.

5.3 Quadrant decision Problem

A major advantage of the original method from Vianello *et al.* [VADJ18] over our method is, that the direction of an inspected point is known, whether it is on the way from left to right or vice versa. Specifically, as we went over in 2.1.3, if the point lies in the quadrants I and IV or in the quadrants II and III. This allows the incrementation of a cell for only one angle for each possible radius.

Since we are working with ellipses, it was not possible for us to determine the direction in a similar matter using the epipolar plane images gradients, so we increment the two possible cells for each points radius as can be seen in figure 3.1. When weighing the matrix, we take this fact into account by multiplying the higher amplitudes by a matrix of shape $e^{-0.002} \cdot [1, 2, \dots, r_{max}]$ instead of $e^{-0.001} \cdot [1, 2, \dots, r_{max}]$. However, incrementing by 2 cells for every points radius still introduces more noise into our Hough spaces.

To further compensate for this, we would need to pinpoint in which of the quadrants an inspected point lies. Using frontal Epipolar plane images, this can be determined by examining their gradients and thus the slope directions of the trajectories. For our ellipse-based approach, however, this becomes more complicated. Figure 5.1 shows an Ellipse travelling through space and time, just as our ellipsoid trajectories would. In order to determine which quadrant a point on the trajectory is, it is necessary to measure the slope at the current point. As can also be seen in Figure 5.1, quadrants

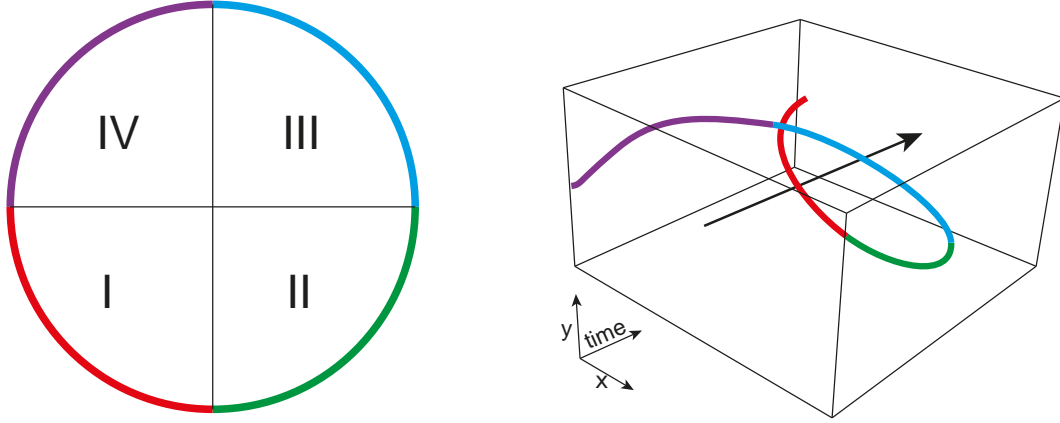


Figure 5.1: This figure shows an ellipsoid trajectory traveling through an epipolar image block. The different quadrants of the ellipse are color coded for distinction.

IV and II have positive slopes, and II and III have negative slopes. Since our method requires knowledge of the rotational axis position, and thus the horizontal center of the ellipse, we would be able to distinguish each quadrant upon acquisition of the slope sign.

In theory, our findings show that this obstacle presents the biggest distinction between the two methods in terms of quality. This in turn means, that overcoming this obstacle would allow scans of a similar quality to the original method using front-view EPI's while using images taken from any angle. On top of this, our method of fusing multiple views would create an environment that could only be better from there.

Despite having a solution proposal, we faced major difficulties regarding slope determination, due to our multi-dimensional scope with limited resolution. We were not able to solve this issue at this point in time. This led us to postpone this particular problem to a future work, and to deal with an increased noise factor in this work.

5.3.1 Platforms out of thin air and strands of Hair

When interpreting figure 5.2 it is noticeable, that the points that are considered and their respective cells get incremented in our Hough spaces always form lines pointing towards the camera. Because this is the case, the point clouds often show patterns resembling strands of hair, where the points and lines are strong, but only ever pointing towards the camera angle. This can be seen in the results chapter 4 as well.

We came across a particularly interesting case of this phenomenon in our research. If a horizontal circle around the central rotation axis is present, and our viewing angle

5.3 Quadrant decision Problem

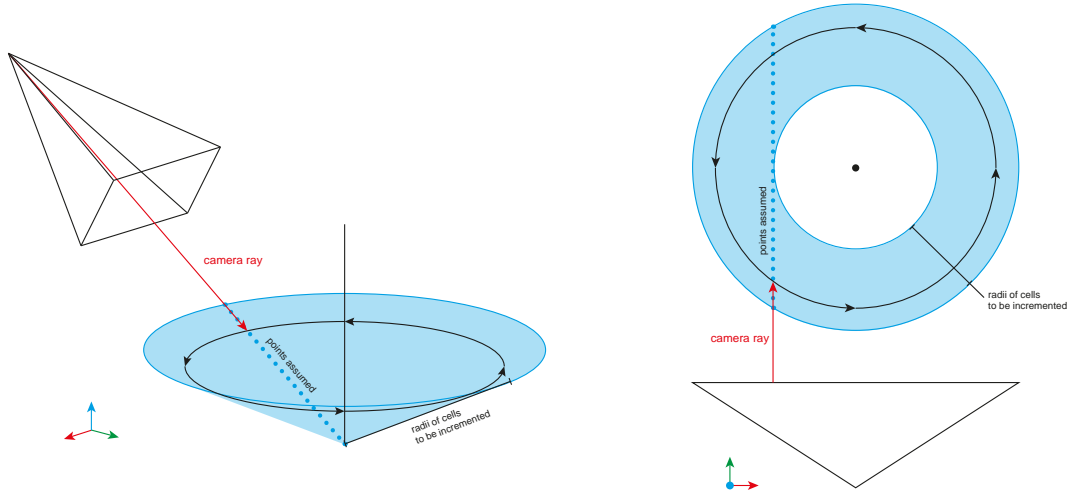


Figure 5.2: This figure shows two examples of how a point on a circle is processed. On the left, a camera with an angle of 60 degree hits a point and increments cells lying on a conical shape. On the right a horizontal camera with its camera ray on the same plane as the circle, the incremented cells also lie on said plane.

is also perfectly horizontal, our camera sees a straight line at every time step. For each point on said line, our method would increment all cells of points with radius of the distance to the center or greater, again, see figure 5.2, right, for a visual aid. Since we are in the scope of a horizontal view, and thus have an ellipse ratio of 0, all of the points will end up in the same plane, thus creating a homogeneous circular plane for just one image. If we go on like this, for all images of our 360° rotation, we will accumulate a plane that is so distinct, that it can't be filtered out anymore, nor is it affected by our weighting matrix. Subsequently, platforms will appear seemingly out of thin air.

In our computer-generated DAVID dataset which can be examined in Chapter 4, we placed the axis of rotation at the center of the base, which approximates a circle. Conical shapes around the base can be noticed in some of the scans of 60° or 120° angles. However, the platforms on the base of the 90° angle scans and each scan partially composed of 90° image scans are quite obvious. In some extreme cases such as 4.10 the platforms are accumulated to such an extent that it messes with the global threshold using Otsu's method. Here the "platform" had such strong values in comparison to everything else that the scans are sampled noticeably sparsely. Although this is a general problem with our method, being able to decide in which of the quadrants a point is would limit the number of points incremented to one quadrant instead of two. This would effectively cut this error in half and would enable a better handling of the platforms.

5.4 Further Complications

5.4.1 Usage of Global Thresholding

After removing the low frequencies and weighing the Hough spaces, the next step in the post-processing pipeline is to compute a global threshold using Otsu's method. Although this worked for the most part, we stumbled upon the need for local thresholding several times during our research. There are a multitude of reasons for data imbalance in the Hough spaces. Some parts of the scanned object may have significantly more detail resulting in more populated edge maps, therefore more entries in the Hough space for these parts. This can also be due to mixed material objects, as we were able to observe in our glitter shoe 5.2.4. Or because some parts are not visible as well in as many frames for them to have a lower impact on the Hough space.

The resulting uneven nature of density distribution proved to be a difficult task, so we did not find a feasible, noise-free solution to the problem. This is why we kept the Otsu Threshold for the scope of this bachelor's thesis, but will refer to future work to solve this issue.

5.4.2 Data imbalance upon Hough space combination

A problem that we only encountered in theory, and have yet to see in practice is regarding combining multiple views before post-processing the data. Although combining scans prior to post-processing is sensible as they would increase the robustness of each other, it can also introduce complications. It can be thought of as a Venn diagram: if two or more angled views of a particular part of the object, let's say the middle of the object, this data will have double the density in the Hough space as for perhaps the top or the bottom of the object only one view has visuals on. This only accumulates as more angles are introduced, creating an imbalanced Hough space. This again could be accounted for by local thresholding as seen in subsection 5.4.1.

5.4.3 Perspective consideration

So far, we have assumed an orthographic projection for both orthographic and perspective image acquisition, while acknowledging the errors caused by this incorrect assumption. As the consideration of perspective distortion is an impactful topic for this method, we will discuss it further in the Future Work section 7.1 to emphasize the need to work on it.

6 Conclusion

To conclude this work, we have successfully introduced a novel extension to circular epipolar plane image based three-dimensional object reconstruction.

Using Vianellos *et al.*'s method of a robust Hough transform, we proposed an algorithm to extend image and view acquisition independent of the camera angle. Using an ellipse-based approach, we were able to parameterize circular trajectories of object points as a way to populate Hough spaces to capture depth information. We extract data from a full 360° view of our object for thorough reconstructions, focusing on accessibility by creating a fully automated pipeline with minimal calibration needs.

Although there is still work to be done, our proposal offers the ability to create 3D-Scans robust to multiple material properties and calibration errors. Furthermore, we introduced simple methods to enable scaling of the results to the world coordinate system, as well as combining parameter spaces of different camera angles to allow for more robust scans in terms of quality and coverage.

7 Future work

After concluding this work, it now leads us outside of our researched terrain. Unfortunately, the scope of this bachelor's thesis did not allow us to look further into the subject. Nonetheless, we can take a glimpse into what could be done proceeding with this work.

7.1 Consideration of Perspective distortion

To drastically reduce our algorithm's complexity, we assumed the acquiring cameras to be orthographic.

This was done in full knowledge of accumulating errors, be that as it may, it still produced credible results.

This is because even if the actual ellipsoid trajectory differs from the assumed ellipse, this variation is minimal enough that a reasonable portion of the point's trajectory stays on an ellipse of our fixed ratio. In addition, we have chosen our cameras to be the closest to orthographic representation available to us, to further minimize the error. But to create more accurate scans, perspective distortion should be considered. The results of our disregards can be seen in figure 4.13.

We assumed a fixed ellipse ratio within an image from top to bottom, since in the orthographic case every horizontal circle would be depicted as an ellipse of the same ratio. However, this does not hold for the perspective case, where the ellipse ratio would be different from top to bottom, as can be seen in figure 7.1. This in turn presents us with the problem of either brute forcing or characterizing the co-dependency of height and ellipse ratio. Both of these approaches exceeded our limitations at this point in time.

Another assumption we made was the height equivalence of the circle center and the center of the fitted ellipse in the images. Also easily noticeable in figure 7.1, this is not the case for perspective cameras. But again, we chose to disregard this fact due to complexity issues.

This in turn means that there is room for improvement in the future by calibrating the acquiring cameras and taking the perspective distortion into account.

7 Future work

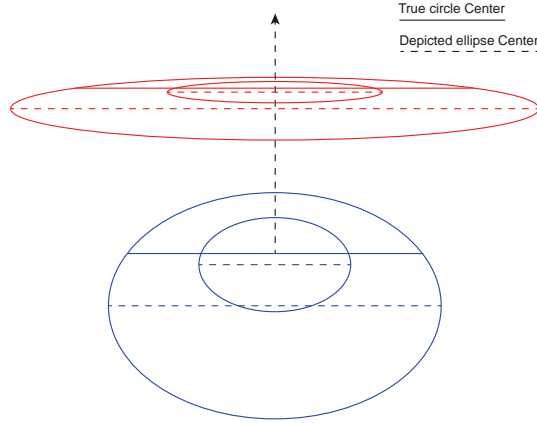


Figure 7.1: Example of the distortion of circles rendered with a perspective camera. The solid horizontal lines are of the circles center, and the dotted lines are the center of the depicted ellipses after perspective transformation. The circles are at two different heights in the scene, grouped by color. This was chosen to be an extreme example with a focal length of only 10mm.

7.2 Availability and Parrallelization

In the scope of this work, we set off to create a proof of concept of the method. Implementing this in Python, our program was not optimized for speed. On the bright side, our algorithm is embarrassingly parallelizable in most of the pipeline steps.

On our setup we found that the image acquisition takes quite a while, which is why simultaneously processing the acquired images is sensible provided the necessary parameters such as ellipse ratio and rotational center.

Furthermore, each image can be processed independently of the others, their results just need to end up in the same Hough space. This allows for parallel computing as well.

But processing a singular image is not the end of the story, every single edge pixel can also be processed in parallel, resulting in a list of Hough space coordinates to increase the parameter spaces cells for.

This allows for significant parallel generation of the Hough space, the most computationally expensive part of our method.

The further processing of data requires a three-dimensional Fast Fourier Transformation, which can also be calculated in parallel. This leaves only a few minor parts of the pipeline and enables a noteworthy increase in speed.

7.3 Mesh, Color and Material

As far as the scope of this thesis is concerned, we set our focus on reconstructing the depth of objects in scenes by creating 3D point clouds with our method for proof-of-concept. However, in order to work with the scans further, mesh reconstruction would be a useful next step, as well as normal estimation, color definition, and perhaps even a material analysis using intensity variations across point trajectories.

Bibliography

- [BB98] M. Bertozzi and A. Broggi. GOLD: a parallel real-time stereo vision system for generic obstacle and lane detection. *IEEE Transactions on Image Processing*, 7(1):62–81, January 1998. Conference Name: IEEE Transactions on Image Processing.
- [BBM87] Robert C. Bolles, H. Harlyn Baker, and David H. Marimont. Epipolar-plane image analysis: An approach to determining structure from motion. *International Journal of Computer Vision*, 1(1):7–55, March 1987.
- [BM92] Paul J. Besl and Neil D. McKay. Method for registration of 3-D shapes. In *Sensor Fusion IV: Control Paradigms and Data Structures*, volume 1611, pages 586–606. SPIE, April 1992.
- [Daf06] Daf-de. Hough Transform example results, August 2006.
- [Fou] Blender Foundation. blender.org - Home of the Blender project - Free and Open 3D Creation Software.
- [GJMSMCMJ14] 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*, 47(6):2280–2292, June 2014.
- [Hou62] Paul V. C. Hough. Method and means for recognizing complex patterns, December 1962.
- [HSS06] B. J. Hosticka, P. Seitz, and A. Simoni. Optical time-of-flight sensors for solid-state 3D-vision. 2006.
- [HZ04] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, Cambridge, 2 edition, 2004.
- [IKL⁺10] Ivo Ihrke, Kiriakos N. Kutulakos, Hendrik P. A. Lensch, Marcus Magnor, and Wolfgang Heidrich. Transparent and Specular Object Reconstruction. *Computer Graphics Forum*, 29(8):2400–2426, December 2010.

Bibliography

- [MBM01] Edward M. Mikhail, James S. Bethel, and J. Chris McGlone. *Introduction to Modern Photogrammetry*. John Wiley & Sons, March 2001. Google-Books-ID: D4h8EAAAQBAJ.
- [noaa] CloudCompare - Open Source project.
- [noab] Head Of Michelangelo's David, Optimised - Download Free 3D model by Thomas Flynn (@nebulousflynn) [d29af50].
- [Ots79] Nobuyuki Otsu. A Threshold Selection Method from Gray-Level Histograms. 1979.
- [RCM⁺01] C. Rocchini, P. Cignoni, C. Montani, P. Pingi, and R. Scopigno. A low cost 3D scanner based on structured light. *Computer Graphics Forum*, 20(3):299–308, 2001. _eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/1467-8659.00522>.
- [Sch94] H. Schultz. Retrieving shape information from multiple images of a specular surface. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(2):195–201, February 1994. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.
- [SFSA12] Christoph Schmalz, Frank Forster, Anton Schick, and Elli Angelopoulou. An endoscopic 3D scanner based on structured light. *Medical Image Analysis*, 16(5):1063–1072, July 2012.
- [VADJ18] Alessandro Vianello, Jens Ackermann, Maximilian Diebold, and Bernd Jähne. Robust Hough Transform Based 3D Reconstruction From Circular Light Fields. pages 7327–7335, 2018.
- [VMDJ17] Alessandro Vianello, Giulio Manfredi, Maximilian Diebold, and Bernd Jähne. 3D reconstruction by a combined structure tensor and Hough transform light field approach. *tm - Technisches Messen*, 84, January 2017.
- [YIM07] G. Yahav, G. J. Iddan, and D. Mandelboum. 3D Imaging Camera for Gaming Application. In *2007 Digest of Technical Papers International Conference on Consumer Electronics*, pages 1–2, January 2007. ISSN: 2158-4001.