Image Depth Estimation Using Stereo Vision

Mrinall Umasudhan

April 25, 2022

Candidate ID: 943289432.

Contents

§1 Introduction

One of the most explored problems in the field of computer vision is the process of accruately estimating the real-world depth of a pixel within a two dimensional image. The inference of three dimensional information is done by using multiple two dimensional views of a scene, the process being demmed the name stereo vision.

§1.1 Applications

A common counter-argument to the practicality of stereo vision algorithms are the presence of other sensors that do not make use of visual data such as ultrasonic or time of flight distance sensors. While these sensors are not not impacted by factors that would be detrimental to the accuracy of stereo vision algorithms such as the lack of adequete lighting, "stereo vision has the advantage that it achieves the 3-D acquisition without energy emission or moving parts" (https://research.csiro.au/qi/stereo-vision/). Moreover, whereas traditional distance sensors focus on a singular point in space, stereo vision algorithms are only limited by the camera's field of view, making the depth analysis large area far more simple and cost effective. Finally, stereo vision algorithms are able to easily work in conjunction with other computer visision techniques such as machine leanrning based object detection models when compared to the previous depth estimation approaches as it already tracks depth on the same image plane that a object detection model may be implemented on. These factors allow for a far greater analysis of the various shapes and angles in an image leading to its usage in various fields.

A common applications for stereo vision algorithms is in the quality control process of industrial factories. Factories must analyze each finshed product for deformities in order to maintian a standard of quality in their products. However, many factories output a high volume of product every day meaning that the human anlysis of such product would be far too expensive and inefficeint when considering the large amounts of workers needed to manually inspect each products as well as the time it takes for the inspection of a product. The installation of multiple distance sensors in order to analyze each sqaure inch of a product would also be far too expensive. However, becuase the factory is in a controlled environment with uniform lighting and the object is one of known shape, the usage of a stereo vision algorithm would be ideal for the situation as stereo cameras are able to analyze objects with their large field of view and can easily detect deformities as the object being analyzed is of known geometry, meaning the algorithm can compare each depth point of the current object to the depth of a model product, reporting any deformities both accuratly and efficiently. and can easily detect deformities as the object being analyzed is of known geometry, meaning the algorithm can compare each depth point of the current object to the depth of a model product, reporting any deformities both accuratly and efficiently.

§2 Triangulation

Core of the algorithm

§2.1 Forward Projection Model

> **Remark 2.1.** Notes

1. First to know how to map 2d to 3d, the traditional Forward Imaging model must be understood.
2. Go over the pinhole model, projection matrix (world to camera and camera to pixel) simplified to the projection formula
3. We can use this simplified formula to derive the distance of the pixel in an image by using two cameras
4. This new formula is ... , take note of the (xl-xr) which indicates the difference in pixel position from the left and right camera this is known as **disparity**
5. However, in order to attain accurate disparity values from the two cameras, they must be calibrated

§2.2 Derivation of Backwards Projection Model

§3 Camera Calibration

§3.1 Intrinsic matrix

Remark 3.1. Notes

1. External Parameters: Position and Orientation of the camera with respect to the world coordinate frame...
2. Internal Parameters: How the camera maps world points into the image coordinate frame.
3. For the case of this program becuase of camera model was constructed to match an ideal model, we only need to make use of two variables the focal length and the distnce between the two cameras d.
4. Use this as a good guide for the process: https://medium.com/swlh/i-see-you-computer-vision-fundamentals-64cc662d0b05. Essentianlly use an mage of known geometry and rearange the original projection eqations to find both the fx anf fy
5. In many modern cameras, the lenses are contructed in such a manner that pixel quality is maximized, however the shape of these lense can lead to singificant distortion in the image taken when compared to the true real world position.
6. There consists of two types of distortion **radial and tangential**
7. In the paper you can describe the correction equation as well as what they algo:ref but in the actual Implementation you would have to use the built in function since a custom distorition function is way to complex , have a custom intrinsic and extrinsic matrix computation tool though.

§3.2 Extrinsic Parameters

§3.3 Lens Distortion

§3.3.1 Radial Distortion

§3.3.2 Tangential Distortion

§4 Stereo Rectification

§4.1 Window Based SSD Disparity Estimation

§4.2 Adaptible Window Optimizations

> **Remark 4.1.** Notes
>
> 1. Now that the two cameras have been calibrated such that they are on a identical image plane wiht one another you can now find the disparity between pixels of the two images in order to find depth using the triangulation model.
> 2. One way this may be done is through a window based method, where we search for the object selected in one image the seocnd by creating a window and linearly searching for identical pixels on the second image. This method may be done efficiently due to camera calibration as the pixel range is on the same scan line.
> 3. We can explore other methods of stereo rectification if we have time as the computation is pretty simple. You would have the use a minimmum squared difference algorithm on the average intensity of the window.

§4.3 Dynamic Programming Based Disparity Estimation

§5 Implementation

§5.1 Hardware

§5.2 Program

§6 Testing

§7 Conclusion

§8 Works Cited

§9 Appendix