Depth Image Region Segmentation

Jackson Spell and **Micah Brown**
{jaspell,msbrown}@davidson.edu
Davidson College
Davidson, NC 28035
U.S.A.

Abstract

Segmentation algorithms separate an image into unique regions determined by characteristics of the pixels within those regions. Image segmentation has applications in Artificial Intelligence, including computer vision, 3D modeling, and robotics. We use two different segmentation algorithms to separate a depth image, which has distance information encoded into each pixel rather than conventional RGB values. Whereas color images are commonly segmented into regions of similar color or luminosity, we segment the depth image into cohesive surfaces. We segment depth images using gradient difference and Laplacian edge detection. When combined together, these two methods offer a chance to reconstruct three dimensional objects from depth data.

1 Introduction

Image processing is a vital component of computer vision and artificial intelligence. Within image processing, image segmentation continues to be a focus of research as machines are expected to be able to have similar visual abilities as humans. Image segmentation divides an image into regions of similar characteristics, which ultimately can lead to object recognition.

Depth images contain information about the distance a point in space is to the camera eye. Whereas traditional RGB images contain a matrix of pixels, each containing a red, green, and blue value, a depth image only contains a matrix of depth values. Depth images can be visually rendered as black and white images, where a point of maximum distance is represented as white and a point of minimum distance is represented as black. Accurate depth images were hard to construct until the invention of modern technologies. Notably, the Xbox Kinect sensor has the ability to simultaneously record depth images and color images, providing researchers easy access to rich depth data.

There are two main approaches for segmenting an image: edge-based segmentation and region-based segmentation (Russell and Norvig 2003). Edge-based segmentation locates major discontinuities in the image, indicating separate distinguishable objects, whereas region-based segmentation identifies different surfaces within those objects by calculating the normal vectors at each point. The remainder of this paper will go over background information pertaining to the two methods of segmentation, describe our specific experimental design, and share our results.

2 Background

Laplacian Edge Detection

Edge detection attempts to distinguish solid objects in an image. This method of segmentation relies on the fact that edges of an image will have large discontinuities in the data. In color images, two different objects often have very different luminosities, colors, or brightnesses; edge detection locates the areas where the pixels experience a rapid change in those values. While pixels in RGB images have these various characteristics, depth image pixels only have one piece of data (the depth). Thus, edge detection in depth images locates objects in an image by finding the areas in which the depth data rapidly changes.

To find edges in depth images, the Marr-Hildreth algorithm can be applied (Marr and Hildreth 1980). To do so, a Gaussian blur convolution is first applied to the depth image. Then, the blurred image, $image_{blur}$, is subtracted from the original image, $image_{original}$, resulting in the "edged" image, $image_{edged}$. Areas with similar values are relatively unchanged after the blur, and result in low values in $image_{edged}$. However, the areas of a depth image with large differences in values have a larger difference represented in $image_{edged}$. When rendered, the $image_{edged}$ is dark in continuous areas of the depth image. However, in places with significant discontinuities, the $image_{edged}$ will have bright edges on those boundaries.

Gradient Surface Detection

Background information on G.S.D. goes here.

$\varphi or \phi = \tan^{-1}\left(\frac{d_1 - d_2}{2}\right)$
$\vartheta or \theta = ...$
$\psi = \cos^{-1}\left(\cos\theta_1 \cos\theta_2 + \sin\theta_1 \sin\theta_2 \cos\left(\theta_1 - \theta_2\right)\right)$

3 Experiments

In this section, you should describe your experimental setup. What were the questions you were trying to answer? What was the experimental setup (number of trials, parameter settings, etc.)? What were you measuring? You should justify these choices when necessary. The accepted wisdom is that there should be enough detail in this section that I could reproduce your work *exactly* if I were so motivated.

4 Results

Present the results of your experiments. Simply presenting the data is insufficient! You need to analyze your results. What did you discover? What is interesting about your results? Were the results what you expected? Use appropriate visualizations. Prefer graphs and charts to tables as they are easier to read (though tables are often more compact, and can be a better choice if you're squeezed for space). **Always** include information that conveys the uncertainty in your measurements: mean statistics should be plotted with error bars, or reported in tables with a ± range. The 95%-confidence interval is a commonly reported statistic.

5 Conclusions

In this section, briefly summarize your paper — what problem did you start out to study, and what did you find? What is the key result / take-away message? It's also traditional to suggest one or two avenues for further work, but this is optional.

6 Acknowledgements

Thanks to Dr. Raghu Ramanujan for guidance throughout this project and Dr. Tabitha Peck for algorithmic inspiration.

References

Marr, D., and Hildreth, E. 1980. Theory of edge detection. *Proceedings of the Royal Society of London. Series B, Biological Sciences* 207(1167):pp. 187–217.

Russell, S. J., and Norvig, P. 2003. *Artificial Intelligence: A Modern Approach*. Pearson Education.