PhysQSR: Improving Reasoning in Three Dimensions and Time With Image Processing and Physics

Nathan W. Eloe[1] and Jennifer L. Leopold[2]

[1]Department of Computer Science and Information Systems, Northwest Missouri State University,
nathan.eloe@gmail.com
[2]Department of Computer Science, Missouri University of Science and Technology,
leopoldj@mst.edu

Abstract

Qualitative Spatial Reasoning (QSR) is an exceptionally powerful tool in the fields of computer cognition and automated computer reasoning. Recent results have shown the potential feasibility of pairing image processing techniques with basic principles of physics that humans inherently understand in order to allow the computer to extrapolate additional information about the environment in which it exists. Initial results showed that, while using the tenets of conservation of mass, conservation of energy, and inertia allowed the computer to gain more information than was initially apparent, noise in the perceived input data resulted in the software erroneously reasoning about the state of the system. Hence improving the image processing techniques used in analyzing the data should ameliorate the errors in reasoning. In this paper, the authors investigate this claim, and present a system that allows a more precise and correct computational view of the environment.

Keywords Qualitative Spatial Reasoning, Image Processing, Object Segmentation

1 Introduction

Human perception of an environment is incredibly complex to replicate in a computational system. Alone, the fact that the perception of the environment is not necessarily consistent between two different observers leads to the conclusion that the results reported by a computer may be verified by one person and invalidated by another. The best that can be done is to have the computer deduce all objectively correct information: that which can be mathematically proven to be true.

Previous results [9] have shown that, in the absence of full three-dimensional knowledge of an environment, it is feasible to use stereoscopic images to obtain information about relative depths and shapes of objects from the computer's observation point. Augmenting this data with laws that govern the physical world allows the computer to learn more about the world. These physical laws were chosen based on their relation to how humans perceive the world. For example, the laws of Conservation of Mass and Energy can directly be seen as a mathematical way to describe Object Permanence [2].

In the authors' previous work, these results showed that noise in the input data led directly to incorrect or even impossible output. The observations of objects when fully visible were used to extrapolate positions of the objects when obscured; noise in the observed positions had a deleterious effect on the extrapolated positions, resulting in potential incorrect assumptions about the system. Herein we discuss the effects a more robust object segmentation algorithm has on the quality of data used in computational reasoning.

2 Background and Related Work

2.1 Image Processing and Disparity

Image processing is an important field in computer and robotic vision. A significant amount of research in this area has been devoted to finding computationally efficient algorithms; images are inherently two-dimensional, which implies that most naive algorithms are at best $O(m \times n)$ in their

DOI reference number: 10.18293/DMS2015-008

computational complexity for an $m \times n$ image. The persistence of high resolution images (full high definition already common and 4k resolution is beginning to emerge) means that these algorithms will be computationally expensive. Many image formats are 4-channel (giving an $m \times n \times 4$ data structure size to hold RGBA or HSVA (Hue Saturation Value Alpha) information, two popular information formats), which only serves to increase the amount of computation needed for a single image.

Disparity [3, 10, 7] and the parallax effect are two concepts exploited in image processing to mimic human perception of depth; objects closer to the observer appear larger than more distant objects. Thus, by determining the parallax between occurrences of an object in each of a pair of stereo images, the relative distance from the cameras to the object can be determined. Disparity also has been used to estimate the motion of objects [7]. It is an invaluable tool in determining spatial information from multiple observations of the same scene.

Object segmentation in image processing is a task that has a large number of established approaches. These methods range from thresholding mechanisms like Otsu's method [11], to clustering algorithms, to region-growing methods. For the purposes of this research, a combination of an edge detection mechanism and a heavy modification of a watershed style algorithm is used to segment objects when all objects are visible. A more explicit description of this object segmentation method is included in Section 3.

2.2 Qualitative Spatial Reasoning (QSR)

Qualitative Spatial Reasoning (QSR) has varying applications in Geographic Information Systems (GIS), visual programming language semantics, and digital image analysis [13, 6, 12, 15]. Systems for spatial reasoning over a set of objects have evolved in both expressive power and complexity. The design of each system focuses on certain criteria, including efficiency of computation, ease of human comprehension, and expressive power.

The spatial reasoning system chosen for this investigation is VRCC-3D+ [16], an expansion and implementation of the RCC-3D [1] system designed by Albath et al. As opposed to other RCC systems (most of which have no implementation), the relations in VRCC-3D+ express both connectivity (in 3D) and obscuration. Obscuration will change from viewpoint to viewpoint, but connectivity is a global property that can be used to discern new information at every perspective in the system.

For this work, the authors focus on the obscuration element of the VRCC-3D+ relation. The connectivity portion of the relation will become important as the system is expanded to handle an arbitrary number of cameras and vantage points. VRCC-3D+ identifies four basic kinds of obscuration: no obscuration (*nObs*), partial obscuration (*pObs*), complete obscuration (*cObs*), and equal obscuration (*eObs*). The system further breaks each base obscuration into four different classes: regular obscuration (object A obscures object B), converse obscuration (object A is obscured by object B), equal obscuration (object A and object B obscure each other equally), and mutual obscuration (objects A and B obscure each other). At this point in the investigation, this further classification is unimportant; it only matters if obscuration is present between two objects, not which object is being obscured.

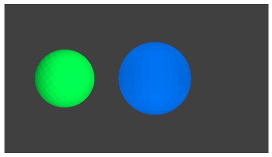
2.3 PhysQSR: QSR with Image Processing and Physics

The system described in [9] has been enhanced to improve the ease with which the system can be used and extended. A thorough explanation of the enhancements made can be found in [8]. Briefly, the implementation of PhysQSR has moved to using a detector system on each set of frame pairs. These detectors are hot-pluggable based on what kind of information they require to run. Initially, only a motion detector and a collision detector had been implemented. This paper describes the construction of an object detector that is used to more accurately track objects as they move through the environment.

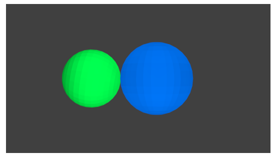
The basic process of analyzing the video footage remains mostly unchanged; while becoming more modular through the use of detectors, each frame pair goes through image analysis (e.g. disparity calculations), obscuration analysis, and object analysis (object position, either calculated or estimated). Previously, this process was very procedural; the shift to detectors allows this to become a more modular process. A pair of frames is grabbed (from each of the left and right cameras), and passed through detectors. These detectors perform tasks such as object detection, obscuration detection, collision detection, and motion detection.

3 Object Segmentation in PhysQSR

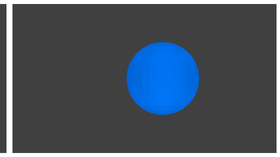
The initial exploration of augmenting QSR with Physics and Image Processing used a very simplistic object segmentation mechanism. The initial testing video input (see Figure 1) were generated in Blender. Because the objects were known to be spheres of two well defined colors, a naive HSV (Hue Saturation Value) filtering/thresholding method was used to generate object masks. This method was satisfactory for one video pair, but when analyzing a more complex scene in which the objects collided, changes in lighting caused a large amount of noise in the calculated and estimated positions of the objects (Figure 2). While the general observed motion of the objects is correct, the noise frequently creates problems where, in some cases, the



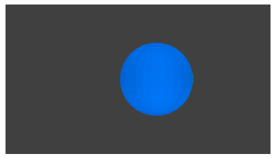
(a) Frame 33: nObs



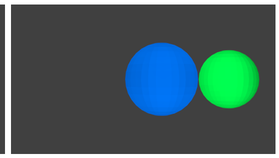
(b) Frame 50: nObs to pObs



(c) Frame 91: pObs to cObs



(d) Frame 101: cObs to pObs



(e) Frame 145: pObs to nObs

Figure 1: Images from analyzed video: as seen from the left camera. The green sphere is further from the cameras than the blue sphere, and as such appears smaller.

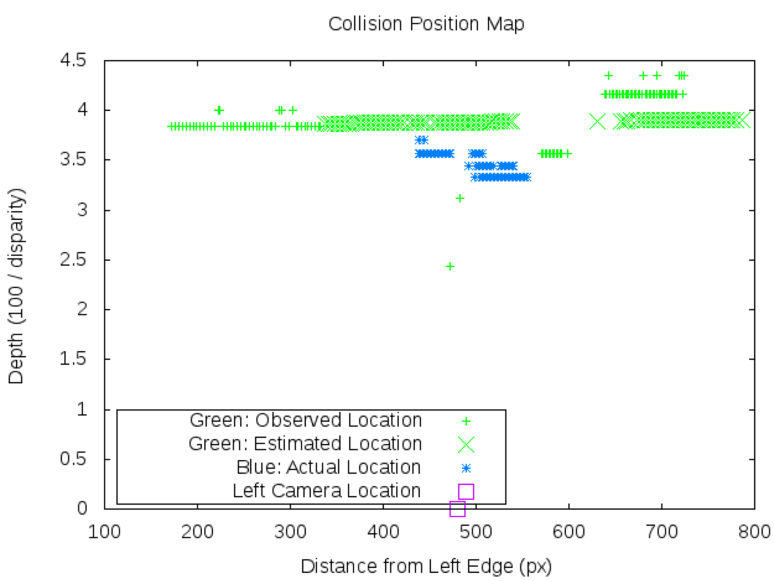


Figure 2: Observed and estimated object positions when collision is present in input.

green ball (which, in the video, passes behind the blue ball) is determined to be in front of the blue ball, a physical impossibility.

A significant portion of the noise in this data can be directly attributed to the effects of lighting in the scene. Figure 3 clearly demonstrates how lighting in the scene can artificially introduce noise into the data when using HSV segmentation. Note that the shape of the mask between the two frames shown changes with the shape of the shadows cast by the blue object. It is interesting to note that these particular images were taken when the lower bound of the saturation value for the mask was set to 20. When set to 100 (the lower bound used in experimentation), both the leading and trailing edge of the masks were practically unrecognizable, which in turn would lead to an unpredictable lateral position in the image. As the position of the trailing edge along the horizontal is used to calculate both the distance from the left edge of the image and the disparity, the oscillating behavior in calculated positions that were initially observed in [9] is understandable.

As such, a more powerful mechanism of object segmentation was deemed necessary. Instead of using HSV masks solely, a combination of edge detection and a watershed like method was used to segment the objects. First, a Canny edge detector [5] was applied to the frame. The thresholds for the hysteresis procedure were set at 150 and 200, and all other parameters were left as the default values provided by the OpenCV [4] Python bindings. The edges resulting from the detector were used to identify closed contours. These contours were then used with a modified Watershed Transform algorithm [14]. In this method the areas enclosed by contours were flooded, designating individual objects. In this way, the objects were segmented. All that remains is to determine which object is which; in this case, the color of the objects is the defining feature, so the HSV masks are still used to identify objects uniquely.

The identified centers of these objects are then used as the locations of the objects for the calculation of disparity and distance from image edge. By using edge detection before the object is filtered based on color, the effects of lighting can be ameliorated. Figure 4 shows how this method provides a much more precise view of where objects are positioned in the image.

4 Results

First, consider the effect this has on the observed position of the objects. Using the new object detection mechanism results in the positions observed in Figure 5. The observed positions demonstrate significantly less noise than the original, as shown in Figure 6. The lateral shift in the position (distance from image edge) is easy to explain. In the previous object tracking method, the trailing edge of the object was used to determine where it existed. Using the new watershed/edge tracking method, the center of mass of the observed object is used. As such, the lateral shift is only a change in the point of interest on the object; the fact that the magnitude of the shift is roughly constant throughout the calculated positions corroborates this.

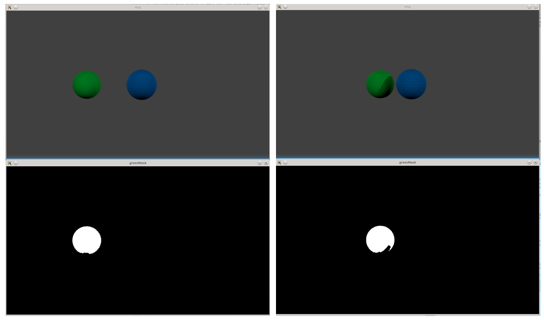
Applying this new object segmentation algorithm to the video originally analyzed in [9] yields some interesting results. Figure 7 shows the calculated positions of the objects in the secene when applying the legacy HSV masking object segmentation algorithm, modified only to take into account the transition of the code to detectors instead of the original procedural mechanism for video analysis, and identical experimentation parameters to the video representing collision. Transitioning to the edge detection segmentation algorithm yielded the positions seen in Figure 8.

It is fascinating that this transition exhibits both improvement and regression. The most marked improvement can be seen in the portion of the scene where the blue ball obscures the green ball. There was significant noise in the data such that the horizontal position of the green ball was calculated to be inhabiting the blue ball's space. Not only that, but at times, the green ball was calculated to be in front of the blue ball, a physical impossibility. However, the noise in the calculated depth at the locations where the green ball was fully visible exhibits less noise overall when using the HSV segmentation than when using the Watershed method. In both cases, however, the noise in the calculated depth cause the fit polynomial to trend closer to the camera. This leads the investigators to believe that perhaps using a polynomial fit line that accounts for all previously known information for the location of the object may not be an optimal way to extrapolate the position of the object.

5 Conclusions

Earlier work showed the feasibility of using image processing techniques on limited visual data (stereoscopic images of a scene) in conjunction with physical properties to learn more about the environment. The identified weakness in the system was primarily the overly simplistic object segmentation mechanism, as it led to noise in the data that caused inconsistencies in the computer system's conclusions. Before expanding the system to work on real-life video input (in real time), the effects of this limitation were deemed critical to examine before continuing with the implementation of this system.

As theorized, using a more robust image tracking system immediately provides better data for the computer to reason with. The system is more robust with respect to variations in lighting and color shifts, and the result is immediately cleaner data. Any regressions noted in the transition did not lead to reasoning that was more incorrect. Indeed, the



(a) Frame 50 (b) Frame 75

Figure 3: The effect of lighting with HSV object segmentation. Only the segmentation of the green object is shown. The white area in each bottom image represents the area marked by the segmentation as belonging to the green object.

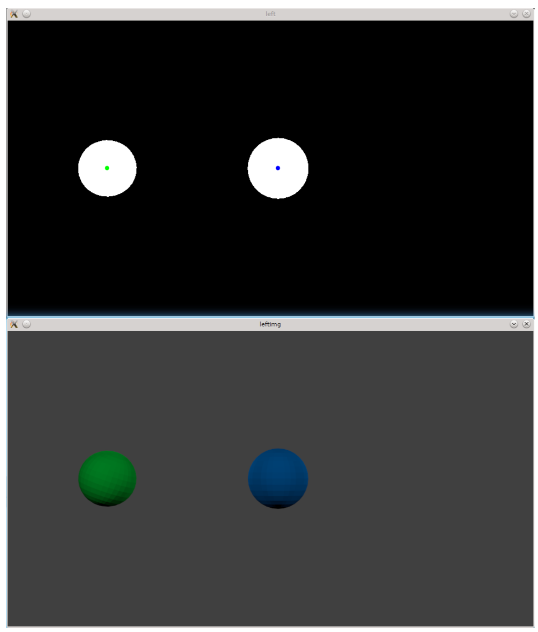


Figure 4: Tracking objects with edge detection and modified Watershed. Each white area is a segmented object in the original image, which are identified with a colored dot corresponding to the original object's color.

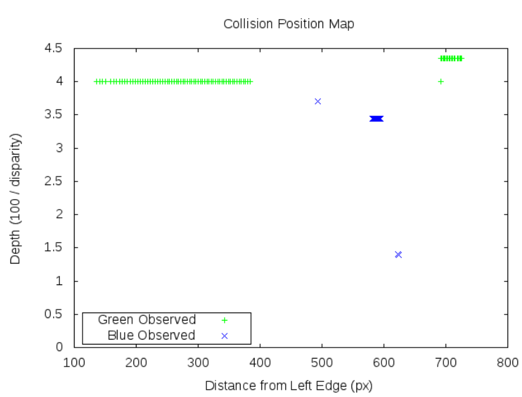


Figure 5: Observed and estimated object positions when collision present in input, new object segmentation.

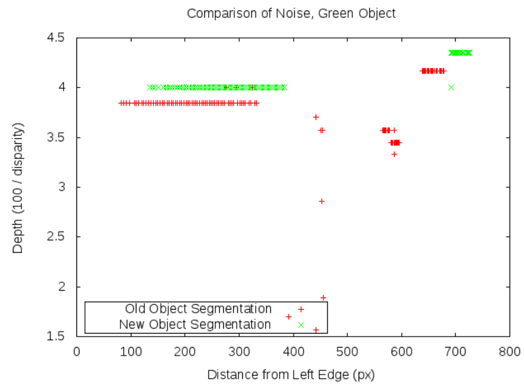


Figure 6: Comparison of noise with new and old object segmentation methods.

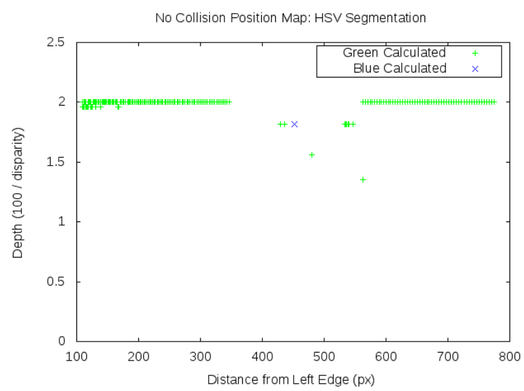


Figure 7: Comparison of noise with new and old object segmentation methods.

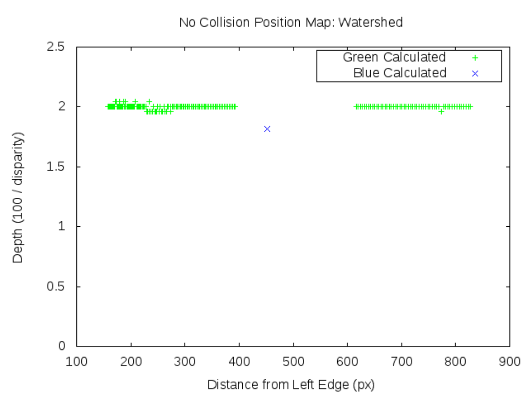


Figure 8: Comparison of noise with new and old object segmentation methods.

additional noise observed in the scenario with no collision resulted in very little to no change in the estimated behavior of the object when it could not be fully observed. This suggests that the system is more sensitive to the magnitude of the noise in the data instead of the amount of the noise.

6 Future Work

Immediate future work on this project will focus on two areas: using the system to analyze real world video data, and further suppression of noise in the input data. Analyzing any real world video data will introduce more noise into the system as imperfect synchronization, mismatched camera sensors, and data transmission over data buses causes degradation. As such, noise suppression will be important to the future success of this project.

Furthermore, investigation of the mechanism used to extrapolate the position of occluded objects will be required. The presence of any noise in the data causes the estimated positions to exhibit erroneous behavior. As it may be impossible to completely eliminate all noise from input data (or even be able to identify noise), more robust object position estimation algorithms will provide a more powerful system that produces reliable results.

References

[1] J. Albath, J. L. Leopold, C. L. Sabharwal, and A. M. Maglia. RCC-3D: Qualitative Spatial Reasoning in 3D. In *Proceedings of the 23rd International Conference on Computer Applications in Industry and Engineering*, pages 74–79, Nov. 2010.

[2] R. Baillargeon. Representing the existence and the location of hidden objects: Object permanence in 6- and 8-month-old infants. *Cognition*, 23(1):21–41, 1986.

[3] S. T. Barnard. Disparity analysis of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2(4):333–340, July 1980.

[4] G. Bradski. *Dr. Dobb's Journal of Software Tools*, 2000.

[5] J. Canny. A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (6):679–698, 1986.

[6] A. G. Cohn, B. Bennett, J. Gooday, and M. M. Gotts. Qualitative Spatial Representation and Reasoning with the Region Connection Calculus. *Geoinformatica*, 1(3):275–316, Oct. 1997.

[7] D. Demirdjian and T. Darrell. Motion Estimation from Disaprity Images. In *Proceedings of the Eighth IEEE International Conference on Computer Vision ICCV*, volume 1, pages 213–218, 2001.

[8] N. Eloe, J. L. Leopold, and C. L. Sabharwal. Spatial Temporal Reasoning Using Image Processing, Physics, and Qualitative Spatial Reasoning. *International Journal of Software Engineering and Knowledge Engineering*. In Review.

[9] N. Eloe, J. L. Leopold, C. L. Sabharwal, and Z. Yin. Spatial Temporal Reasoning Using QSR, Physics, and Image Processing. In *Distributed Multimedia Systems '13*, pages 14–19, Aug 2013.

[10] K. Mühlmann, D. Maier, J. Hesser, and R. Männer. Calculating Dense Disparity Maps from Color Stereo Images, an Efficient Implementation. *International Journal of Computer Vision*, 47(1–3):79–88, Apr. 2002.

[11] N. Otsu. A threshold selection method from gray-level histograms. *Automatica*, 11(285-296):23–27, 1975.

[12] D. A. Randell and M. Witkowski. Using Occlusion Calculi to Interpret Digital Images. In *Proceedings of the 17th European Conference on Artificial Intelligence*, pages 432–436, Aug. 2006.

[13] J. Renz. *Qualitative Spatial Reasoning with Topological Information*. Springer-Verlag, 2002.

[14] J. B. Roerdink and A. Meijster. The watershed transform: Definitions, algorithms and parallelization strategies. *Fundamenta informaticae*, 41(1):187–228, 2000.

[15] P. Rost, L. Hotz, and S. von Riegen. Supporting Mobile Robot's Tasks through Qualitative Spatial Reasoning. In *Procceedings of the Ninth International Conference on Informatics in Control, Automation and Robotics (ICINCO 2012)*, volume 2, pages 394–399, 2012.

[16] C. L. Sabharwal, J. L. Leopold, and N. Eloe. A More Expressive 3D Region Connection Calculus. In *Proceedings of the 2011 International Workshop on Visual Languages and Computing (in conjunction with the 17th International Conference on Distributed Multimedia Sysems (DMS 11))*, pages 307–311, Aug. 2011.