Fast 3D Reconstruction of Foreground Objects using Mask R-CNN and Space Carving

Max Ferguson
Stanford University
Stanford, CA, United States
maxferg@stanford.edu

Abstract

With the commercialization of active stereo technology, it is now possible to obtain accurate and detailed 3D building models. However, the volumetric reconstruction of moving objects inside buildings is still a challenging task. In this work, we generate approximately correct volumetric representations of moving objects, in near real-time. We use a state-of-the-art image segmentation model to identify foreground objects which need to be reconstructed. We then obtain silhouettes of each object, from multiple viewpoints, using the same segmentation model. Finally, we use space carving to obtain an approximate volumetric representation of each foreground object. Hardware acceleration is used to improve the performance of the reconstruction pipeline by an order of magnitude over the CPU implementation. We show that proposed pipeline is robust to lighting conditions, to the extent that it can reconstruct a person in a darkened room.

1. Introduction

Recent advances in active stereo have made it relatively easy to create 3D building models [1]. However, tracking the location of moving objects inside these buildings is still a challenging problem [21]. In this work, we aim to use recent advances in two-dimensional image segmentation to reconstruct 3D objects in near real-time. The final goal is to generate an approximate 3D reconstruction of people moving inside a building. The reconstructed objects can then be placed inside a static model of the building geometry, to generate a detailed real-time building information model.

There are a number of factors motivating fast 3D reconstruction of building environments. Our motivation is to better facilitate the movement of mobile robots, such as autonomous wheelchairs and cleaning robots, in building environments. Through the 3D reconstruction of building environments, we hope to gain a better understanding of how people move around indoor spaces. However, fast 3D reconstruction of building environments could also be useful for other applications, ranging from building security to augmented reality [5, 21].

The fast reconstruction of foreground objects is particularly important in applications where we would like to generate a digital representation of a certain environment. Active sensing techniques, such as Light Detection and Ranging (LIDAR), can be used accurately to reconstruct the static part of the environment (we will refer to this as the background). However, using active sensing techniques to reconstruct moving (foreground) objects is still a challenging task. Most active sensors have a relatively low refresh rate, which is problematic for applications like real-time collision avoidance. Additionally, most active sensors return a raw point-cloud, with very little information about object class or segmentation. While we believe that active sensing will be a crucial component of many future robotic systems, we would like to explore a reconstruction technique that more closely resembles the human visual and inference system.

Recent advances in computer vision have made it possible to perform accurate pixel-level image segmentation in real-time. Figure 1 provides an example of image segmentation using Mask Region-based CNN (Mask R-CNN) [11]. Mask R-CNN almost perfectly locates and classifies the objects we would like to reconstruct. By applying Mask R-CNN to images from multiple viewpoints, we are able to extract rich and structured information about the scene. We explore how traditional computer vision techniques can be used reconstruct 3D objects in the scene, using this structured information.

1.1. Related Work

There is an extensive amount of literature on 3D reconstruction. However, most existing literature focuses on the reconstruction of surfaces using stereo vision [10], active stereo [8], or sensor fusion [13]. One of the more interesting approaches uses Bayesian methods to reconstruct known

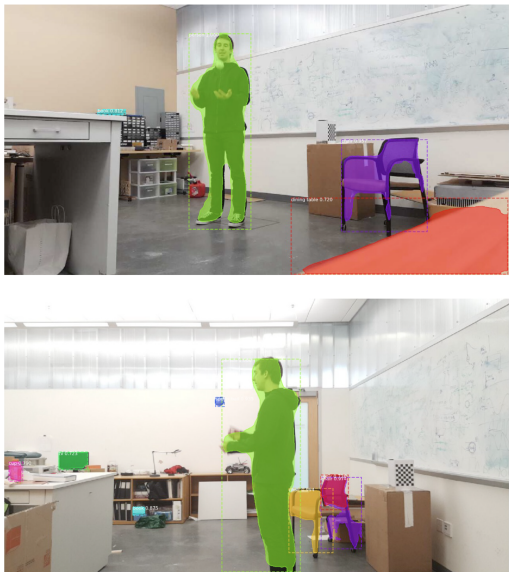

Figure 1. Image segmentation using Mask Region-based CNN. Foreground objects are well-segmented from the background.

Figure 2. Naive surface reconstruction can yield impressive results, but lacks information about volumetric segmentation. Source [2]

objects, such as humans, from a single camera view [12]. While generating 3D surfaces is useful in a range of applications, we choose to focus on reconstructing volumes. Figure 2 illustrates the surface reconstruction of a person from stereo images. While the reconstruction is geometrically detailed, the resulting surface is not particularly useful for our mobile robot navigation system.

A common approach to volumetric reconstruction is space carving [15]. Space carving is a computationally efficient method for extracting 3D geometry from multiple camera views [15]. Space carving has been used reconstruct players in a sporting environment [19]. A number of extensions have been proposed to handle noisy input data [3]. The space carving algorithm can be accelerated using a Graphical Processing Unit (GPU), as described in [7]. A review of hardware accelerated space-carving algorithms is provided in [20].

More recently, researchers have proposed a range of deep learning approaches to 3D object reconstruction [14]. Wu et al proposed an approach where complex shape distributions were learned from raw 3D data [24]. A more advanced approach that leverages generative adversarial networks has since been published [23]. In 2015, VoxNet was proposed to solve the inverse problem: classifying an object given the 3D structure [18]. MarrNet was developed to estimate the 3D structure of an object from a 2D image and a depth map [22]. While impressive, it is important to note that most of these approaches just learn representations of common objects, and do not directly address the 2D to 3D task. We believe that further progress can be made by combining data-driven image segmentation systems with more traditional 3D reconstruction algorithms.

2. Problem Statement

The goal of this project is to generate a volumetric reconstruction of a scene, using images from at least two viewpoints. We assume that the intrinsic and extrinsic properties of each camera are known. We also assume that the background is static and has known geometry. Therefore, the primary task is to reconstruct foreground objects that are moving or can be moved. The algorithm should work in near real-time; we hope to design a system that can process at least 6 frames per second.

As the system will primarily be used for collision avoidance and object tracking, the 3D reconstruction of each foreground object can be approximate. However, the predicted spatial location of the object should be correct. The foreground objects are assumed to be deformable, so we do not attempt to iteratively improve the reconstruction over multiple frames.

Additionally, the system should be able to simultaneously classify and reconstruct foreground objects. That is, the class of each volume should be specified. We do not attempt to segment multiple objects of the same class, but this could be an interesting topic for future work.

3. Technical Approach

In the proposed approach, we perform 2D image segmentation on image frames from each view to extract object silhouettes, and then use space carving to obtain a volumetric model of each foreground object. The classification of each object is performed at the same time as image segmentation. This makes it straightforward to select the correct silhouettes for each object when performing volu-

metric reconstruction. Figure 3 describes the classification and volumetric pipeline at a high level. The components of the pipeline are described in more detail throughout this section.

3.1. Image Segmentation

We use the Mask R-CNN architecture for image segmentation [11]. Mask R-CNN is a state-of-art object detection and segmentation system, based on the Faster Region-based CNN (Faster R-CNN) backbone. The Mask R-CNN architecture is composed of three modules. The first module is a CNN that proposes regions of interest (ROIs). The second module is a CNN that attempts to classify the objects in each region [11]. The third module performs image segmentation, with the goal of generating a binary mask for each region. We refer the reader to the original Mask R-CNN paper for a more detailed description of the neural network architecture [11]. The model is trained to recognize objects in the Microsoft COCO dataset, as well as common office objects, using publicly available datasets [16, 6]. We follow the same training procedure described in the original Mask R-CNN paper [11]. For the purpose of understanding the results in this work, it is sufficient to know that the trained Mask R-CNN model can classify objects with classes `chair` and `person`.

The output of Mask R-CNN is a set of bounding boxes, a set of class labels, and a corresponding set of binary object masks. Each mask encodes the spatial layout of an object inside one of the bounding boxes. We choose a mask size of 28 x 28 pixels, to be consistent with other researchers [11]. The masks are synonymous with object silhouettes, and can be directly used for space carving.

3.2. Volumetric Reconstruction

The image segmentation module provides a silhouette for each detected object. The first task is to find which silhouettes corresponds to each 3D object. As each mask is labeled by class, it straightforward to group masks by object class. In some cases a single object is recognized by the segmentation module as multiple different objects. An example of this issue is shown in Figure 4. To overcome this issue, we merge the silhouettes by class, such that there is only one silhouette per object class. This method is effective, but it means that the volumetric reconstruction pipeline is no longer able to segment multiple objects of the same class.

Space carving is performed to reconstruct objects that are detected in all of the camera views. To obtain a coarse estimate of the working volume we first perform space carving with the Mask R-CNN bounding boxes. We then perform space carving with the silhouettes to obtain a refined volumetric representation of each object [15]. The benefits of this approach are twofold: Firstly, each object is reconstructed individually using the same number of voxels, so the level of detail in each reconstruction is proportional to the object size. Secondly, the objects are reconstructed from silhouettes with known class, so the class of the reconstructed object is also known.

When rendering the reconstructed objects we would prefer to render a smooth mesh instead of a voxel grid. A fast C++ implementation of the marching cubes algorithm is used to extract a polygonal mesh from the carved voxel grid [17]. The Laplacian smoothing algorithm is then used to smooth the polygonal mesh [9]. Finally, the mesh is rendered to an image using the Vispy OpenGL wrapper [4].

4. Experiments

In this section we describe the experiments that were carried out using the proposed reconstruction pipeline. We begin be describing the laboratory setup. We then provide some qualitative and quantitative experimental results.

4.1. Dataset

The dataset was collected in the Advanced Sensing Laboratory at Stanford University, specifically for this project. Three network-connected cameras were set up in the laboratory. The cameras were positioned strategically to maximize the amount of material that could be removed in the space carving process. A video feed from each of the cameras was simultaneously streamed to a desktop computer and fed into the 3D reconstruction pipeline. The video streams were also recorded for future experiments.

Each camera was calibrated using images of a chessboard. We obtained the intrinsic parameters K_m, and the extrinsic camera parameters R_m, T_m, for each camera m. Radial distortion was initially modeled as follows:

$$x_{distorted} = x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \quad (1)$$

$$y_{distorted} = y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \quad (2)$$

where r is the radius in pixel coordinates, and k_1, k_2 and k_3 are radial distortion coefficients. However, we found that correcting for distortion during the projection step of the space carving algorithm introduced a number of artifacts in the carved object. An example these artifacts is shown in Figure 5. Fixing the distortion parameter k_3 to zero yielded much better results. Therefore, the radial distortion was modeled as follows:

$$x_{distorted} = x(1 + k_1 r^2 + k_2 r^4) \quad (3)$$

$$y_{distorted} = y(1 + k_1 r^2 + k_2 r^4) \quad (4)$$

This model was deemed acceptable, as k_3 is only necessary to correct for severe distortion, such as that introduced by

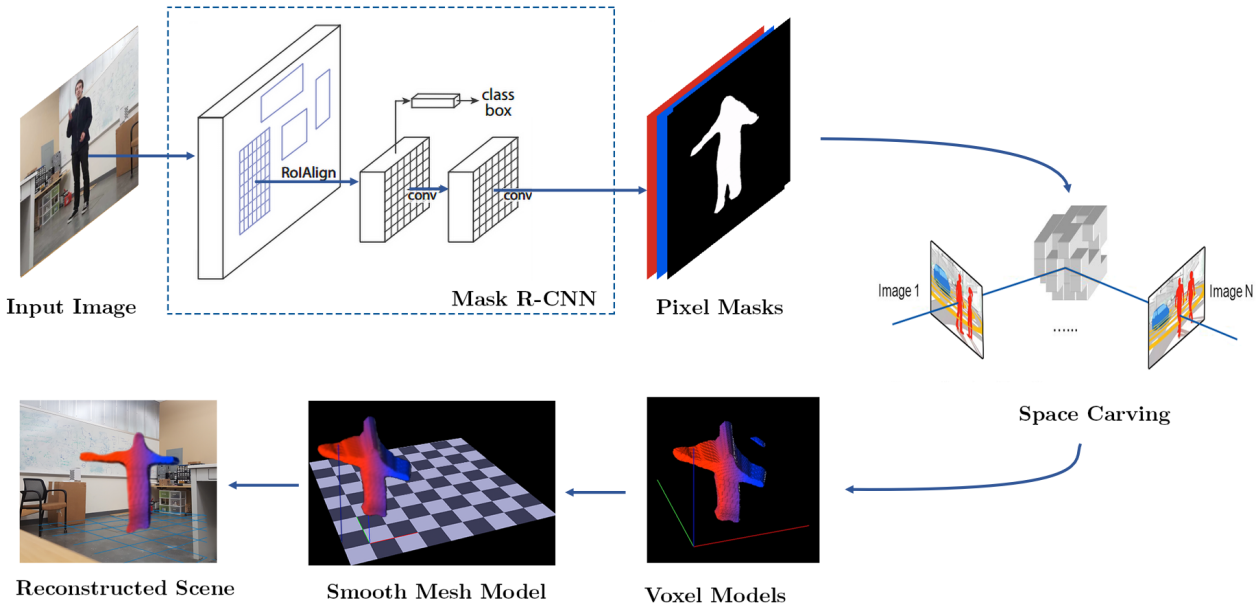

Figure 3. Proposed technical approach. Mask R-CNN is used to generate silhouettes of foreground objects from multiple viewpoints (top row). Space carving is used to obtain a volumetric representation of each foreground object (bottom row). The marching cubes algorithm is used to generate a smooth mesh surface from the voxel representation [17].

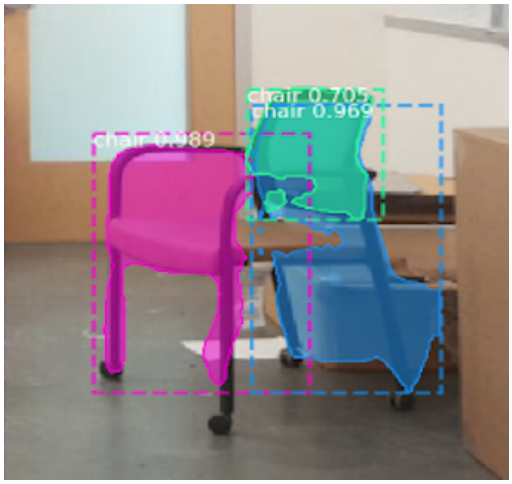

Figure 4. A single chair object, incorrectly recognized as three individual chairs. This issue can be resolved by merging all chair masks.

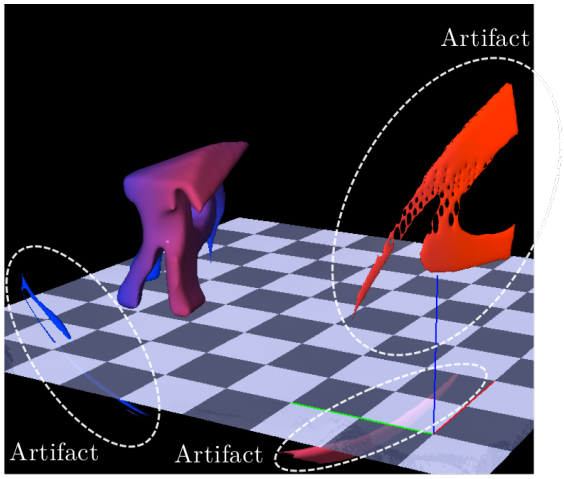

Figure 5. Artifacts introduced when silhouettes were corrected for excessive distortion. We are unsure whether this is due to an error in camera calibration or a mathematical instability in the space carving algorithm

wide-angle lenses [2]. Tangential distortion was modeled as:

$$x_{distorted} = x + [2p_1xy + p_2(r^2 + 2x^2)] \quad (5)$$

$$y_{distorted} = y + [p_1(r^2 + 2y^2) + 2p_2xy] \quad (6)$$

where x,y are pixel coordinates, and p_1, p_2 are tangential distortion coefficients. The final dataset consist of 66 chessboard images for calibration (22 for each camera) and about 10 minutes of video recorded from each camera. The video contains moving people, office chairs and common office items.

4.2. Single Object Reconstruction

In this experiment, we aim to reconstruct a single object using the proposed reconstruction pipeline. In the first trial,

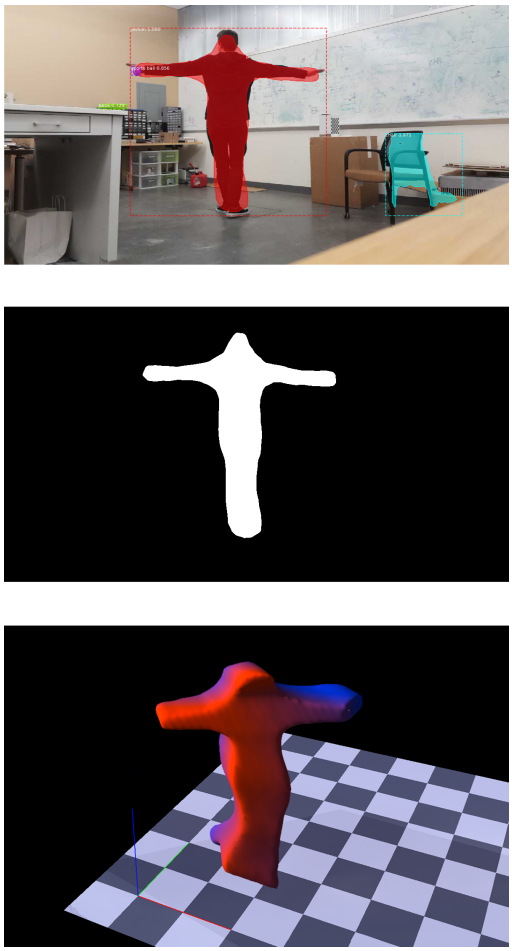

Figure 6. Object reconstruction pipeline for fast reconstruction of a single object. From the top (a) the segmented image from Mask R-CNN (b) The silhouette for the person class (b) the reconstructed person object.

we reconstruct a moving person using camera frames from three cameras. Various steps in the reconstruction pipeline are shown in Figure 6. The silhouette of the person is sufficiently accurate for use in real-time collision avoidance systems etc. However, the silhouette does not capture small details, like the narrowing of the persons neck. This is a known drawback of Mask R-CNN, and exists because silhouettes are only predicted at 28 x 28 px resolution. Increasing the resolution of the predicted silhouette is possible, but greatly increases computational requirements.

The visual hull of the person is reconstructed reasonably well. The reconstructed object appears larger than ground truth. This is expected when using the space carving procedure with very few camera views. While far from perfect, the reconstruction of this person is sufficiently accurate for the intended application, mobile robot collision avoidance.

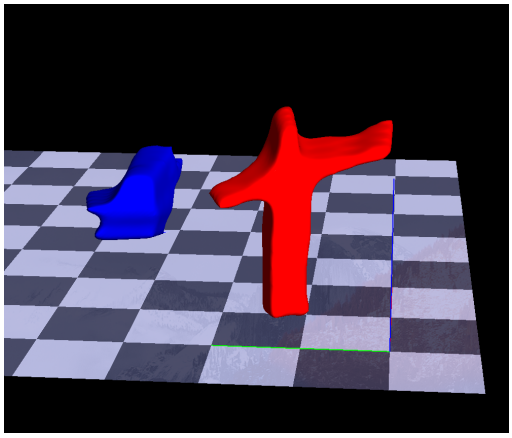

Figure 7. Reconstruction of a person and a chair. The person is shown in red and the chair is shown in blue

4.3. Multi-Object Reconstruction

In this section, we investigate whether the proposed 3D reconstruction pipeline can be used to classify and reconstruct multiple objects. We attempt to reconstruct a scene containing a person and a chair. Again, the reconstruction pipeline is applied to video frames from three network cameras. An example of the reconstructed scene is shown in Figure 7. The person and the chair appear to be reconstructed correctly.

4.4. Lighting Conditions

In mobile robot and autonomous vehicle applications, it is not alway possible to control the environmental lighting conditions. Therefore, object detection algorithms must be robust to different lighting conditions. Developing segmentation algorithms that are robust to lighting has been a difficult challenge in computer vision for many years. However, as our reconstruction pipeline uses a robust CNN approach for segmentation, we expect good performance in a range of lighting conditions. To test this theory we ran the reconstruction pipeline on images from a darkened room. Images of the room, as well as the resulting reconstruction can be seen in Figure 8. The person was correctly reconstructed in every frame of a 30 second video (900 frames). This robustness is a direct benefit of using a powerful data-driven segmentation model for generating object silhouettes.

4.5. Performance

For applications in robotics, it is important that the volumetric reconstruction process can run in real-time. In this section, we analyze the temporal performance of each component in the 3D reconstruction pipeline. The reconstruction pipeline system is evaluated on a 3.6 GHz Intel Xeon E5 desktop computer with 8 CPU cores, 32 GB RAM, and a single NVIDIA GTX 1080Ti GPU. The pipeline is eval-

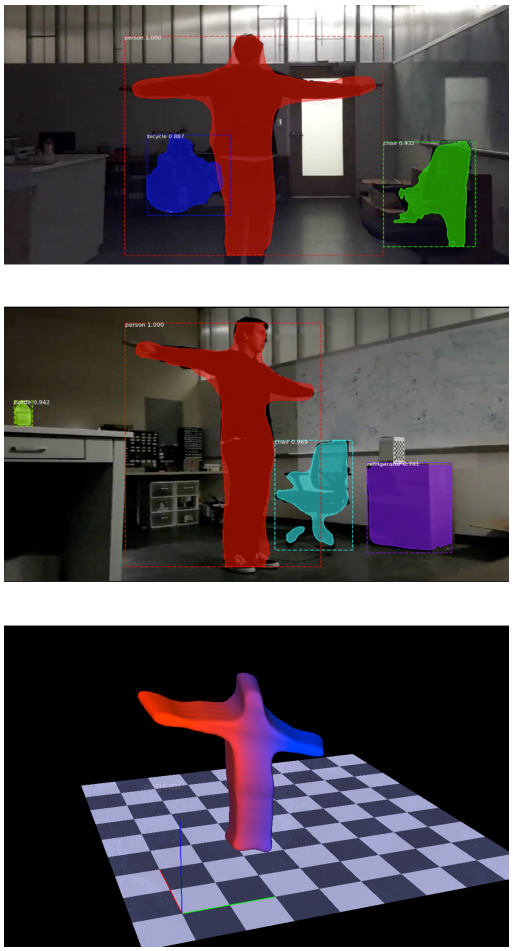

Figure 8. Reconstruction of person in a dark room. The visual hull of the person is reconstructed extremely well, given the adverse lighting conditions

uated with the GPU enabled and with the GPU disabled. The cameras are connected to the desktop computer using a wireless local-area network (LAN). In each test, we reconstruct scenes from 60 seconds of video. The average duration of each component in the pipeline is calculated. The results are shown in Table 1.

The pipeline is prohibitively slow without GPU acceleration. On average, each scene took 4.539 seconds to reconstruct, excluding network latency. With GPU acceleration, the average duration was reduced to 0.465 seconds, excluding network latency. At this speed, the pipeline can be marginally considered as near-real time. However, it is probably too slow for real-time collision avoidance in robotic systems.

There are a number of possible methods to accelerate the pipeline. Network latency is a major component of the overall scene-processing time. Computing the silhouettes at the camera location may reduce the amount of data that needs to

	Duration (ms)	
Pipeline Component	GPU disabled	GPU enabled
Network latency (video)	832	841
Image Segmentation	2643	156
Space Carving	913	205
Marching Cubes	73	70
Rendering	910	34
Total	5371	1306
Total (without network)	4539	**465**

Table 1. Time cost of the 3D reconstruction pipeline modules. Results are averaged over 1800 scene reconstructions (60 seconds of video with 30 Hz framerate)

be transmitted over the network. In the current implementation, data is moved from the CPU to the GPU and back to the CPU at every step in the pipeline. Keeping data in the GPU memory throughout the entire pipeline could lead to significant performance improvements.

5. Discussion

Volumetric reconstruction has been explored by the computer vision community for many years. In this work, we combined a state-of-the-art segmentation model with a traditional object reconstruction technique, to produce a fast and robust reconstruction pipeline. The key advantages of the proposed pipeline are:

- Only foreground objects are reconstructed
- Multiple objects can be reconstructed at once
- Reconstructed objects are automatically classified
- The pipeline is very robust to lighting conditions

However, the system is not without limitations. We found that the space carving algorithm is very sensitive to camera calibration. In some cases, the space carving algorithm generates volumes that are much smaller than ground truth. This is particularly problematic when trying to reconstruct small objects. One potential solution is to use triangulation to correct for mis-calibration. However, this approach would require correspondences to be identified across images. Another limitation of the pipeline, is that it returns the visual hull of the 3D objects, rather than the true object form. This is a known limitation of the naive space carving algorithm, and was deemed acceptable for the intended collision-avoidance application.

Future work could focus on improving the reconstruction of objects from the Mask R-CNN silhouettes. A particularly interesting application would be to use deep learning

techniques to reconstruct objects from their silhouettes, and object class. In this approach, the neural network would be able to learn priors over the 3D shape of different objects. We expect that the combination of object silhouettes and class priors could enable very accurate reconstructions. We also expect that this approach would also eliminate the need to have multiple cameras surrounding the object. One could imagine that accurate volumetric reconstruction of foreground objects could be obtained using an RGBD camera, a segmentation module like Mask-RCNN, and a reconstruction module like MarrNett [22].

6. Conclusion

We have presented a pipeline for 3D reconstruction of foreground objects using the Mask Region-based CNN for image segmentation and space carving for volumetric reconstruction. The proposed pipeline is able to reconstruct and classify multiple objects with different classes in the same scene. While we designed the pipeline with robustness in mind, we were surprised that objects were almost always well reconstructed, even in a darkened room. Through a number of experiments we demonstrated that the proposed pipeline is able to reconstruct objects with suitable accuracy for many applications in robotics. However, the pipeline relies on well-calibrated and well-positioned cameras for accurate reconstruction.

7. Source Code

The source code for this project can be found at https://github.com/maxkferg/object-reconstruction. To comply with Github file size restrictions, some of the videos are excluded from the repository. Please contact the author directly if you would like access to the video dataset.

References

[1] I. Armeni, S. Sax, A. R. Zamir, and S. Savarese. Joint 2d-3d-semantic data for indoor scene understanding. *arXiv preprint arXiv:1702.01105*, 2017.

[2] J.-Y. Bouguet. Matlab camera calibration toolbox. *Caltech Technical Report*, 2000.

[3] A. Broadhurst, T. W. Drummond, and R. Cipolla. A probabilistic framework for space carving. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 1, pages 388–393. IEEE, 2001.

[4] L. Campagnola, A. Klein, E. Larson, C. Rossant, and N. P. Rougier. Vispy: harnessing the gpu for fast, high-level visualization. In *Proceedings of the 14th Python in Science Conference*, 2015.

[5] M. Chu, J. Matthews, and P. E. Love. Integrating mobile building information modelling and augmented reality systems: An experimental study. *Automation in Construction*, 85:305–316, 2018.

[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. IEEE, 2009.

[7] M. Denkowski. Gpu accelerated 3d object reconstruction. *Procedia Computer Science*, 18:290–298, 2013.

[8] A. Dipanda, S. Woo, F. Marzani, and J.-M. Bilbault. 3-d shape reconstruction in an active stereo vision system using genetic algorithms. *Pattern Recognition*, 36(9):2143–2159, 2003.

[9] D. A. Field. Laplacian smoothing and delaunay triangulations. *International Journal for Numerical Methods in Biomedical Engineering*, 4(6):709–712, 1988.

[10] A. Geiger, J. Ziegler, and C. Stiller. Stereoscan: Dense 3d reconstruction in real-time. In *Intelligent Vehicles Symposium (IV), 2011 IEEE*, pages 963–968. Ieee, 2011.

[11] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pages 2980–2988. IEEE, 2017.

[12] N. R. Howe, M. E. Leventon, and W. T. Freeman. Bayesian reconstruction of 3d human motion from single-camera video. In *Advances in neural information processing systems*, pages 820–826, 2000.

[13] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison, et al. Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 559–568. ACM, 2011.

[14] D. Ji, J. Kwon, M. McFarland, and S. Savarese. Deep view morphing. Technical report, Technical report, 2017.

[15] K. N. Kutulakos and S. M. Seitz. A theory of shape by space carving. *International journal of computer vision*, 38(3):199–218, 2000.

[16] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[17] W. E. Lorensen and H. E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *ACM siggraph computer graphics*, volume 21, pages 163–169. ACM, 1987.

[18] D. Maturana and S. Scherer. Voxnet: A 3d convolutional neural network for real-time object recognition. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 922–928. IEEE, 2015.

[19] D. S. Monaghan, P. Kelly, and N. O'Connor. Quantifying human reconstruction accuracy for voxelcarving in a sporting environment. In *Proceedings of the 19th ACM international conference on Multimedia*, pages 1525–1528. ACM, 2011.

[20] C. Nitschke, A. Nakazawa, and H. Takemura. Real-time space carving using graphics hardware. *IEICE TRANSACTIONS on Information and Systems*, 90(8):1175–1184, 2007.

[21] D. Ramanan, D. A. Forsyth, and A. Zisserman. Tracking people by learning their appearance. *IEEE transactions on pattern analysis and machine intelligence*, 29(1):65–81, 2007.

[22] J. Wu, Y. Wang, T. Xue, X. Sun, B. Freeman, and J. Tenenbaum. Marrnet: 3d shape reconstruction via 2.5 d sketches. In *Advances In Neural Information Processing Systems*, pages 540–550, 2017.

[23] J. Wu, C. Zhang, T. Xue, B. Freeman, and J. Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In *Advances in Neural Information Processing Systems*, pages 82–90, 2016.

[24] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1912–1920, 2015.