3D Object Reconstruction using Stationary RGB Camera

José G. dos S. Júnior[1][a], Gustavo C. R. Lima[1][b], Adam H. M. Pinto[1][c], João Paulo S. do M. Lima[2][d], Veronica Teichrieb[1][e], Jonysberg P. Quintino[3], Fabio Q. B. da Silva[4], Andre L. M. Santos[4] and Helder Pinho[5]

[1] *Voxar Labs, Centro de Informática, Universidade Federal de Pernambuco, Recife, Brazil*

[2] *Departamento de Computação, Universidade Federal Rural de Pernambuco, Recife, Brazil*

[3] *Projeto de P&D CIn/Samsung, Universidade Federal de Pernambuco, Recife, Brazil*

[4] *Centro de Informática, Universidade Federal de Pernambuco, Recife, Brazil*

[5] *SiDi, Campinas, Brazil*

Keywords: 3D Reconstruction, Background Segmentation, Stationary Camera.

Abstract: 3D objects mapping is an important field of computer vision, being applied in games, tracking, and virtual and augmented reality applications. Several techniques implement 3D reconstruction from images obtained by mobile cameras. However, there are situations where it is not possible or convenient to move the acquisition device around the target object, such as when using laptop cameras. Moreover, some techniques do not achieve a good 3D reconstruction when capturing with a stationary camera due to movement differences between the target object and its background. This work proposes two 3D object mapping pipelines from stationary camera images based on COLMAP to solve this type of problem. For that, we modify two background segmentation techniques and motion recognition algorithms to detect foreground without manual intervention or prior knowledge of the target object. Both proposed pipelines were tested with a dataset obtained by a laptop's simple low-resolution stationary RGB camera. The results were evaluated concerning background segmentation and 3D reconstruction of the target object. As a result, the proposed techniques achieve 3D reconstruction results superior to COLMAP, especially in environments with cluttered backgrounds.

1 INTRODUCTION

The creation of 3D assets is one of the challenges concerning virtual and augmented Reality, mainly when turning a real-world object or scenario into a virtual reference. One of the main techniques applied to 3D reconstruction consists of mapping the desired target using different images from various points of view, know as photogrammetry (Thompson et al., 1966). However, to fully make a 3D reconstruction, one of the leading technologies used is Structure from Motion (SfM) (Ullman, 1979) combined with Multi-View Stereo (MVS) (Goesele et al., 2006). Those together are responsible for getting camera pose parameters and matching features to create a dense point cloud representation of the desired object or scenario. After that, meshing and texturing algorithms do the final work of modeling the reconstruction.

Reconstruction pipelines such as COLMAP (Schonberger and Frahm, 2016) provide all the necessary steps for an excellent 3D object mapping from image sets with different points of view. However, although this technique works well in scenarios where the camera moves around the target object, there are situations where its easier to move the object itself keeping a stationary camera. Much pipelines do not handle this situation well. This causes SfM and MVS to not work as expected, generating wrong camera poses, which seriously harms the final 3D reconstruction results. One solution would be to extract the background from the scene, segmenting the valuable part of the image for reconstruction. However, current segmentation techniques needs prior information about the target object (Rother et al., 2004) (Maninis et al., 2018).

[a] https://orcid.org/0000-0001-5808-0371
[b] https://orcid.org/0000-0002-5843-742X
[c] https://orcid.org/0000-0001-9302-3575
[d] https://orcid.org/0000-0002-1834-5221
[e] https://orcid.org/0000-0003-4685-3634

S. Júnior, J., Lima, G., Pinto, A., Lima, J., Teichrieb, V., Quintino, J., B. da Silva, F., Santos, A. and Pinho, H.
3D Object Reconstruction using Stationary RGB Camera.
DOI: 10.5220/0010807000003124
In *Proceedings of the 17th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2022) - Volume 5: VISAPP*, pages 793-800
ISBN: 978-989-758-555-5; ISSN: 2184-4321

This paper presents a new 3D reconstruction pipeline COLMAP-based that uses automatic target object segmentation without requiring camera calibration, manual intervention, prior knowledge or scene manipulation. The main contributions of this work are (1) the improvement of the GrabCut and Deep Extreme Cut techniques, making background segmentation possible in batches of images without the need for manual interventions; (2) Creation of a COLMAP-based 3D object mapping pipeline to allow the reconstruction of objects from images obtained by a low-resolution stationary camera and; (3) set of test cases with quantitative evaluation of the background segmentation.

2 RELATED WORKS

Acquiring 3D information of an object is an important field of research in computer vision and graphics. Moreover, with the advent of virtual and augmented reality, creating 3D models of objects is fundamental. However, despite a large amount of research in the area, differently from large scene reconstruction, extracting small object information is more challenging, and the current state of the art uses more advanced sensors for that.

In this work, like the ProForma (Pan et al., 2009) technique, we propose a batch solution using one single stationary RGB camera, but do this by combining a modified version of COLMAP (Lyra et al., 2020) pipeline for extensive scene reconstructions with modified segmentation algorithms to enable object reconstruction.

The interest in the reconstruction of non-rigid objects has also grown with works such as (Newcombe et al., 2015; Yu et al., 2015; Bozic et al., 2020) and those are interested in using RGB-D cameras for better results. As for the rigid objects, reconstruction works such as (Locher et al., 2016; Pokale et al., 2020) are focusing more on moving cameras and single image pose estimation, respectively, for applications in mobile phones and robotics. But in our solution, we focus on image sequences for using the SfM technique. Both (Shunli et al., 2018; Zhang et al., 2019) use SfM for mapping the scene.

For object reconstruction, the background segmentation from the object is fundamental. The approach of Kuo et al. (Kuo et al., 2014) is a mobile solution for object reconstruction that makes a study about three different segmentation algorithms. Among them, GrabCut (Talbot and Xu, 2006) is a user-guided solution where it takes as input an image and a bounding box around the desired object. The algorithm then selects the pixels outside that box as known background and the inside ones as unknown. After that, a sequence of Gaussian Mixture Models (GMMs) (Reynolds, 2009) are created until all pixels converge for the final segmentation. The GrabCut does an excellent job on a clean background, but it struggles to make a pleasing contour around the object when it comes to complex ones with different textures.

Another more robust algorithm, in this case, is Deep Extreme Cut (Maninis et al., 2018), which is also a user-guided technique. It takes as input an image and the four extreme points of the desired object to be segmented, but it uses deep learning to improve the segmentation, making it more reliable in textured backgrounds.

3 PROPOSED METHOD

3.1 Overview

As already mentioned in Section 2, the primary task on 3D object reconstruction is the background segmentation for proper mapping. That step is even more relevant when using a stationary RGB camera, where the object's movement will provide the necessary data for SfM. To achieve that, we used a modified version of (Lyra et al., 2020), made for large-scale scene reconstruction. Furthermore, we applied a combination of the contour retrieval method available on the OpenCV library (Bradski and Kaehler, 2000) and both GrabCut (Talbot and Xu, 2006) and Deep Extreme Cut (Maninis et al., 2018) for the automatic segmentation procedure. In the end, an updated version of the Poisson Surface Reconstruction (PSR) method (Kazhdan and Hoppe, 2013) was employed for mesh generation and texturing.

3.1.1 BGSLibrary

Background subtraction consists of comparing an observed image with another one that represents the background. In general, the results tended to show a partial and unrepresentative image of the object. As a work centered on the typical user, the specific background settings were out of the question, so we used the solution to automatically clear the background (Sobral and Bouwmans, 2014) proposed the BGSLibrary to facilitate the use of these algorithms. Currently, the library is open-source, written in C++, based on OpenCV, and has 43 algorithms available for video background separation.

3.1.2 GrabCut

Considering that the techniques used had already been proven with good reconstructions in previous works using a mobile camera, the results were still below expectations even when using the background removal techniques. However, the resulting images still had a lot of noise, with information that was not part of the main object. To solve this problem, we first use GrabCut (Rother et al., 2004), which is available on OpenCV.

In the GrabCut implementation, the monochromatic image is replaced by a colored one using a GMM. This first segmentation is followed by border matting, computing a narrow band around these segmentation limits. One of the important points of Grabcut, compared to other techniques, is little need for interaction with the user. However, for our solution, there should not be any user interaction, even more considering that multiple frames are used to create a model.

To solve this problem we initially use the BGSLibrary that detects movement in the scene and identifies the target object's location in the image, returning a binary mask that specifies these regions. Then, through a morphological dilation transformation, the algorithm calculates the most significant contour in this mask, having the greatest possibility of being the object of interest. An example of the complete pipeline can be seen in Figure 1.

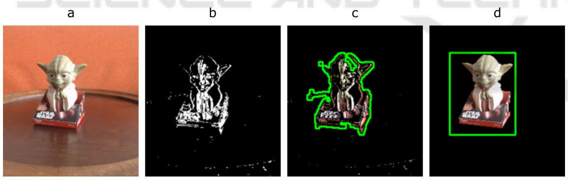

Figure 1: GrabCut pipeline showing the input image (a), the background filter found (b), the most important contour found (c), and the final result (d).

3.1.3 Deep Extreme Cut

Background segmentation using BGSLibrary and GrabCut presents good results in environments with little background texture information. However, combining these two techniques gives segmentation problems with some frames, especially when trying to segment objects with many concavities acquired in environments with low light or with a cluttered background. One alternative is the Deep Extreme Cut (Maninis et al., 2018).Deep Extreme Cut is an algorithm that uses convolutional neural networks (CNN) to segment images based on RGB information and a set of four extreme points of the target object silhouette (left-most, right-most, top, bottom pixels). In this technique, the CNN receives as input an image of the scene and a heat map containing information about the extreme points and returns a probability map that specifies if each pixel is part of the target object or not. Then from the probability in each pixel, it is possible to infer a binary mask used to segment the input image.

The CNN can segment background images with good accuracy and robustness results, even in critical situations. However, the original technique requires manual intervention by the user to locate the extreme points in the image. To solve this problem, we propose to perform the detection of the extreme point automatically. We also used the BGSLibrary to extract the object contours. Thus, from the coordinates of the points on this contour, it is possible to extract the extreme points necessary for the operation of Deep Extreme Cut. A representation of this process can be seen in Fig. 2.

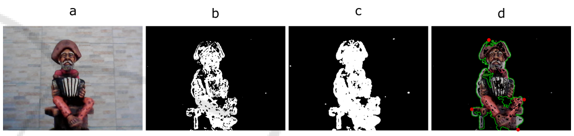

Figure 2: Automatic extreme points selection for Deep Extreme Cut. Input image (a), binary mask with motion regions (b), dilated mask binary (c), and target object contour and extremes points (d).

4 EXPERIMENTS

For the experiments, we created a dataset using a stationary camera composed of 12 test cases, in which three objects (*car*, *oldman*, *geisha*) with different textures and formats were shot on a turntable in 4 different environments with background texture and lighting variations. To acquire the dataset images we positioned the target object at a distance of 30 to 40 cm from the capture device. We used OBS Studio and a USB2.0 VGA UVC WebCam with a maximum resolution of 640x480 (0.307 MP) integrated to the laptop ASUS VivoBook X510U. We are not concerned with camera calibration during image acquisition, since the intrinsic camera parameters estimated by COLMAP during reconstruction proved to be sufficient for providing good results. Thus, we were able to obtain a dataset with the following characteristics: little or no movement in the background; objects with different physical characteristics, symmetrical or non-symmetrical, with little or a lot of texture information; different levels of disorder present in the background; and different levels of lighting (outdoor/indoor).

In Fig. 3 shown some frames of the different test cases: “oldman-01” (Fig. 3a), with poorly textured background; “oldman-02” (Fig. 3b), with untextured background; “oldman-03” (Fig.3c), with other objects present in the background; and ”oldman-04” (Fig. 3d), with different lighting conditions for untextured background. Fig. 3e, f, g shown the three objects used to compose the dataset.

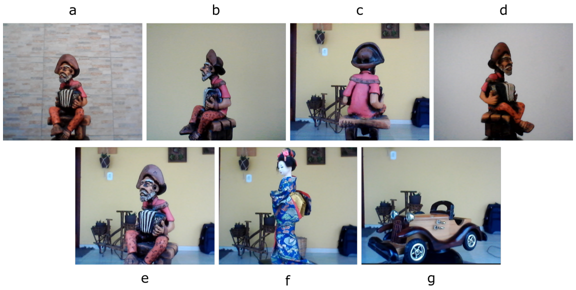

Figure 3: Examples of test cases of the dataset with variation in the type of background and the target object.

In this work, we divided the experiments into two steps: the first was used to analyze changes in the background segmentation methods proposed. The main objective of this step is not to assess the best segmentation technique but to survey the main features in our results to understand the impact of each of them on the 3D reconstruction process. In the second step, we evaluate the 3D object reconstruction with a stationary camera from the results of these segmentation methods.

For a qualitative assessment of the segmentation step, we observed how close the segmentation mask was to the real silhouette of the target object and counted the number of technique failures. For this, we determined that the technique fails when it confuses the background with the foreground during segmentation. We consider segmentation failure cases when parts of the background were not segmented or when elements of the object were cropped along with the background.

To evaluate the reconstruction technique, we compared the results of 3D reconstruction from images obtained with a stationary camera using the original and modified COLMAP pipeline and the reconstruction pipeline proposed in this work. It means that we added pre-processing with two segmentation techniques, BGSLibrary with GrabCut or BGSLibrary with Deep Extreme Cut. To make this comparison possible, we used the number of points, the deformation, the noise, and the completeness of the reconstructed model.

In performing the experiments, we used a notebook with an Intel Core i7-7700HQ @ 2.80 GHz 2.81 GHz processor, 16 GB of RAM, and NVIDIA GeForce GTX 1060 6GB graphics card. The system was implemented in C++, with the support of some libraries, such as OpenCV, COLMAP, BGSLibrary, and PyTorch C++ API.

5 RESULTS AND DISCUSSION

5.1 Segmentation Results

The first experiments were carried out to evaluate the results of background segmentation using the improvements of segmentation techniques proposed. To make this work easier to read, we will name the two approaches as **BGS-Grab** and **BGS-Deep** to refer to GrapCut and Deep Extreme Cut techniques respectively.

Due to the high cost in the technique processing time, we used a sampling of the test cases frames in all experiments. For this, it was selected 1 in every 20 frames. We named this parameter “skipped frames”. This results in an average of 70-100 images used for the reconstruction. We achieved good reconstruction with 30-40 images. However, using fewer images generates inconstant results more frequently. In relation to the BGSLibrary algorithms was used *DPTexture* to BGS-Grab and *frameDifference* to BGS-Deep in all experiments.

We evaluate the robustness of the techniques according to the number of poorly segmented background frames concerning the total frames used in each test case. For this, we consider two failures types: the first, called here as “With bg”, occurs when the distance between one of the segmentation binary mask’s edge points and the target object’s edge is greater than 30 pixels, causing part of the background to appear in the final segmentation result. The second, called “Cut object”, occurs when the technique cuts parts of the target object as if it were background.

Tab. 1 presents the segmentation results using the test cases with *oldman*. Analyzing the results, concerning wrong segmentation of the “With bg” error, we observed that the BGS-Deep technique obtained the best results, except in the test case “oldman-01”. Regarding the “Cut object” error, we can observe that the BGS-Grab technique obtained the best results in most test cases except for “oldman-03”. The results also showed that considering the “With bg” failure type, both techniques performed better in the test cases in environments with low-textured backgrounds. However, the results still showed that the BGS-Deep technique was little affected by the change of ambient lighting in the “oldman-04” test case. On the other hand, BGS-Grab had the number of faults "With bg" considerably increased in this scenery.

Fig. 4 shows a sample of the background segmentation results in the "oldman-02" case. The results were purposely selected in a sequence where well-segmented and failure cases occurred in both techniques. In general, BGS-Grab obtains better accuracy than the BGS-Deep. That means, even in the best BGS-Deep cases (4g and 4h), the segmentation mask's lack of accuracy concerning the actual object silhouette is noticeable. It is also possible to observe that when the "With bg" error occurs (4c, 4d, 4e, and 4f), the background area in the final result is more clear in BGS-Grab than in BGS-Deep.

Figure 4: Segmentation results in oldman-02 test case. a, b, c and d uses BGS-Grab; e, f, g and h uses BGS-Deep.

As we are using a modified version with an automatic selection of extreme points and bounding rectangles, the accuracy of these processes is not better than the originals that use manual selection. However, we allow the segmentation of objects without previous knowledge or manual intervention from stationary camera images. The average time for concluding the segmentation in these tests cases was about 15-20 minutes.

5.2 Reconstruction Experiments

The experiments with 3D reconstruction were carried out to compare the results using four different reconstruction pipelines: BGS-Grab (Proposed-01) and BGS-Deep (Proposed-02), the original COLMAP and the modified COLMAP. In all experiments, we used the "SIMPLE RADIAL" camera model. In both versions of COLMAP, we use the "exhaustive matching" mode in the feature matching step. For all other configuration parameters of the technique, the library default values were kept. The average time for our reconstruction was about 1 hour and 20 minutes.

In Tab. 2 we can see a summary of the results in *oldman* reconstruction experiments using four analyzed pipelines. Among the four pipelines evaluated, the one that presented the best regularity concerning the number of vertices of the resulting meshes was "Proposed-01". However, it is possible to note that this pipeline presented problems in reconstructing the target object in the test case with cluttered background ("oldman-03"), with the resulting point cloud being deformed and incomplete. Also, the "Proposed-01" pipeline did not show a good reconstruction using the test case with low lighting "oldman-04".

Regarding the "Proposed-02" pipeline, none of the experiments resulted in deformed or noisy clouds. However, in the "oldman-01" and "oldman-04" test cases, the generated point clouds were incomplete and with fewer points compared to the average of the others. This was the only technique that enabled a non-deformed, complete, low-noise reconstruction using the "oldman-03" test case. While the "COLMAP modified" got good reconstruction results in "oldman-2" and "oldman-4" many artifacts were generated in the mesh, and had a low number of "Mesh vertices" compared to the others.

The COLMAP modified version could not reconstruct both "oldman-01" and "oldman-03". The experiments using the original COLMAP pipeline showed the worst results compared to the others. However, the original COLMAP pipeline was the only one that allowed a complete reconstruction of the target object in "oldman-04", although the resulting cloud was quite noisy.

In Fig. 5, we expose the experimental results with the test case oldman-01, where the Proposed-01 pipeline obtained better reconstruction compared to Proposed-02. The Proposed 2 pipeline was not able to rebuild the object's back, probably due to the number of "Cut object" type failures that the technique obtained in this test case (see Tab. 1). In Fig. 5 there are noise near the bench, but it is not possible to say that this did harm the construction of the mesh in the tests with Proposed-01.

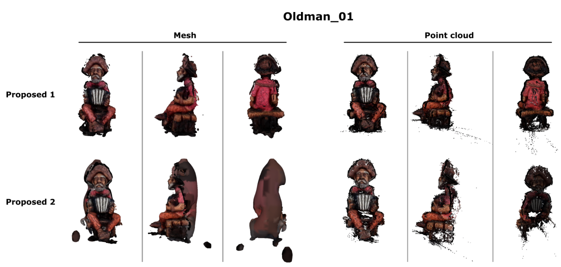

Figure 5: Views of mesh and point cloud of 3D reconstruction results using "oldman-01" test case.

In Fig. 6 the results of the 3D reconstruction from test case "oldman-02" are presented. The number of vertices of each mesh obtained by the "Proposed-01" and "Proposed-02" pipelines are very close. How-

Table 1: Comparison of segmentation techniques in the oldman test cases.

Test case	Algorithm	#Images	With bg	Ratio (%)	Cut object	Ratio (%)
oldman-01	BGS-Grab	100	**0**	**0.0**	**11**	**11.0**
	BGS-Deep	100	5	5.0	75	75.0
oldman-02	BGS-Grab	68	15	22.1	**4**	**5.9**
	BGS-Deep	68	**7**	**10.3**	24	35.3
oldman-03	BGS-Grab	83	72	86.7	25	30.1
	BGS-Deep	83	**25**	**30.1**	**22**	**26.5**
oldman-04	BGS-Grab	74	26	35.1	**8**	**10.8**
	BGS-Deep	74	**6**	**8.1**	43	58.1

Table 2: Comparison of 3D reconstruction pipelines in the oldman test cases.

Reconstruction pipeline	Test case	# Mesh vertices	Deformed mesh	Completeness	Noise
Proposed 1	oldman-01	105,092	no	yes	few
	oldman-02	108,964	no	yes	few
	oldman-03	110,336	yes	no	a lot
	oldman-04	103,478	no	no	few
Proposed 2	oldman-01	76,846	no	no	few
	oldman-02	108,785	no	yes	few
	oldman-03	111,305	no	yes	few
	oldman-04	76,209	no	no	few
COLMAP modified (Lyra et al., 2020)	oldman-01	-	-	-	-
	oldman-02	42,223	no	yes	a lot
	oldman-03	-	-	-	-
	oldman-04	33,737	no	yes	a lot
COLMAP original (Schonberger and Frahm, 2016)	oldman-01	-	-	-	-
	oldman-02	602,946	no	no	a lot
	oldman-03	-	-	-	-
	oldman-04	1,038,721	no	yes	a lot

ever, it is possible to verify that both the mesh and the point cloud obtained by "Proposed-01" presents a better quality than "Proposed-02", mainly in the back region.

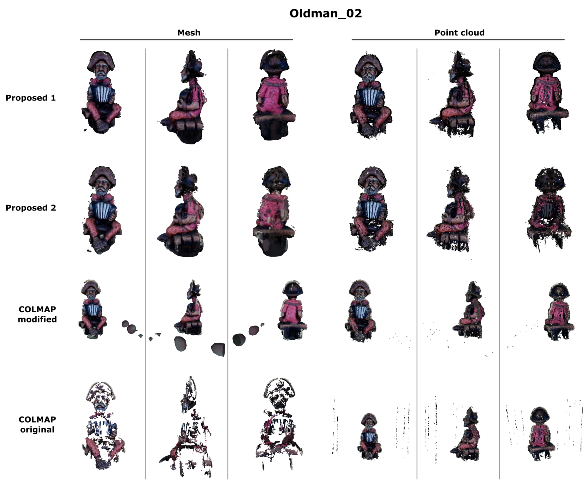

Figure 6: Views of mesh and point cloud of 3D reconstruction results using "oldman-02" test case.

The results of the experiments with "oldman-03" are shown in Fig. 7. So far, "oldman-03" was the one with the more considerable reconstruction difficulty due to the amount of texture information present in the background. Here, it is possible to see that this had a negative impact on the reconstruction performed by "Proposed-01", as the reconstruction presents an incomplete and deformed point cloud and a mesh with only some information from the target object. In this case, the "Proposal-01" obtained the worst segmentation results concerning the two types of failures accounted for in Tab. 1. The results of "Proposed-02" pipeline did not present significant problems, except for some flaws in a small region of the object.

In Fig. 8 it is possible to see that the lack of luminosity had a negative impact on the "Proposed-02" pipeline in "oldman-04" test case, which presented an incomplete point cloud and mesh. In this test, the "Proposed-01" was the one that presented the best reconstruction results, with a complete mesh of the target object, containing only some noise and flaws in the point cloud.

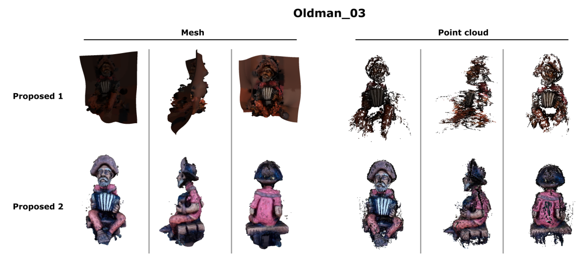

Figure 7: Views of mesh and point cloud of 3D reconstruction results using ”oldman-03” test case.

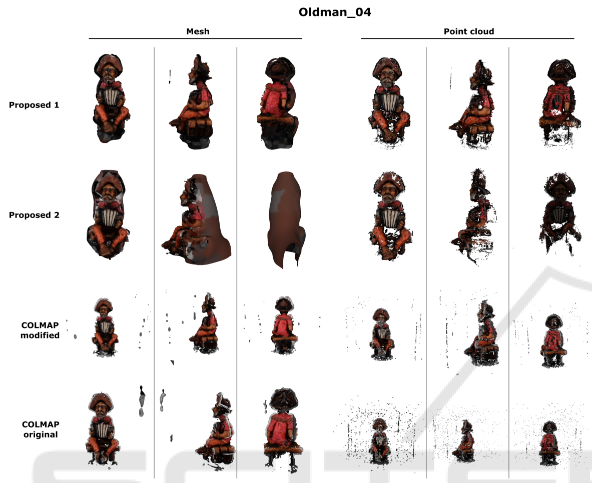

Figure 8: Views of mesh and point cloud of 3D reconstruction results using ”oldman-04” test case.

As “Proposed-01” had the best results in poorly textured environments, this technique was selected to test objects’ physical nature using other test cases created by us. Fig. 9 presents some of the results obtained during these experiments, showing the results of the target object reconstruction from three points of view from the “geisha-02” test case (Fig. 9a) and from the “car-4” test case (Fig. 9b). From the geisha images, the pipeline allowed a significant reconstruction while preserving the geometric and texture details of the object. However, there is a relevant problem in the reconstruction technique in the car test case, in which the symmetry confuses the algorithm, generating a partial reconstruction.

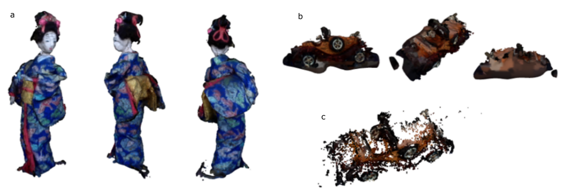

Figure 9: Geisha and car statue reconstruction. Three points of view from geisha mesh (a) and car mesh (b), and one point of view from the car point cloud (c).

5.3 Discussion

In the first stage of the experiments, it was possible to observe that the proposed techniques in this work enabled automatic segmentation without prior knowledge of the target object. However, the results showed some problems of lack of accuracy and robustness in segmentation when calculating the binary mask. In these results, parts of the background appeared in several frames or, in others, parts of the target object were cut off. These experiments also showed that BGS-Grab enabled better segmentation than BGS-Deep in environments with untextured backgrounds. At the same time, BGS-Deep presented better results in the test case with cluttered backgrounds. The COLMAP pipelines, which does not have a segmentation step, was the one that obtained the worst results, not even being able to complete the reconstruction in two of the scenarios used.

Once it was observed that the quality of the 3D reconstruction depends on the quality of the results in the segmentation stage, the segmentation challenges started to be interpreted as problems of the all process using a stationary camera. Experiments with good sets of segmented images in the preprocessing stage, with reasonable accuracy and robustness, showed that the target object’s physical characteristics could negatively influence the final reconstruction result. It is also possible to highlight that the quality of the results is quite sensitive to the choice of the motion detection algorithms of BGSLibrary; the number of skipped frames used in sampling each test case; and the number of features in the COLMAP feature extraction step.

6 CONCLUSIONS

In this work, we propose changes in background segmentation techniques to compose two new 3D object mapping pipelines based on COLMAP (Lyra et al., 2020), thus making it possible to obtain good 3D object reconstructions from stationary camera images. We adapted the results of motion detection algorithms from BGSLibrary (Sobral and Bouwmans, 2014) as input to segmentation techniques proposed in (Talbot and Xu, 2006; Maninis et al., 2018), enabling background extraction without manual intervention and a priori information about the target object. A set of experiments were carried out and showed that the proposed 3D mapping pipelines presented better results than the original COLMAP (Schonberger and Frahm, 2016) and modified COLMAP (Lyra et al., 2020), with the segmentation steps corresponding to an additional 15% of the total processing time of the full reconstruction technique and improving the results significantly.

As future works, we believe that a study that seeks to improve the accuracy of detecting the bounding rectangle is needed for BGS-Grab. The detection of the extreme points of BGS-Deep can improve the accuracy and robustness of the background segmentation in both pipelines and, consequently, the quality of the final 3D reconstruction. We also believe that a study in which partial occlusion and sudden movements are considered in the target object shooting could also enable the reconstruction from test cases in which the objects are moved manually. Such a study would provide an alternative to use turntables during capture, making the scenario of obtaining the datasets closer to realistic scenarios. Complementarily, GPU processing in parts of the 3D mapping pipeline still processed in the CPU can significantly decrease the technique processing time.

REFERENCES

Bozic, A., Zollhofer, M., Theobalt, C., and Nießner, M. (2020). Deepdeform: Learning non-rigid rgb-d reconstruction with semi-supervised data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7002–7012.

Bradski, G. and Kaehler, A. (2000). Opencv. *Dr. Dobb's journal of software tools*, 3.

Goesele, M., Curless, B., and Seitz, S. M. (2006). Multi-view stereo revisited. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 2402–2409. IEEE.

Kazhdan, M. and Hoppe, H. (2013). Screened poisson surface reconstruction. *ACM Transactions on Graphics (ToG)*, 32(3):1–13.

Kuo, P.-C., Chen, C.-A., Chang, H.-C., Su, T.-F., and Lai, S.-H. (2014). 3d reconstruction with automatic foreground segmentation from multi-view images acquired from a mobile device. In *Asian Conference on Computer Vision*, pages 352–365. Springer.

Locher, A., Perdoch, M., Riemenschneider, H., and Van Gool, L. (2016). Mobile phone and cloud—a dream team for 3d reconstruction. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–8. IEEE.

Lyra, V. G. d. M., Pinto, A. H., Lima, G. C., Lima, J. P., Teichrieb, V., Quintino, J. P., da Silva, F. Q., Santos, A. L., and Pinho, H. (2020). Development of an efficient 3d reconstruction solution from permissive open-source code. In *2020 22nd Symposium on Virtual and Augmented Reality (SVR)*, pages 232–241. IEEE.

Maninis, K.-K., Caelles, S., Pont-Tuset, J., and Van Gool, L. (2018). Deep extreme cut: From extreme points to object segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 616–625.

Newcombe, R. A., Fox, D., and Seitz, S. M. (2015). Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 343–352.

Pan, Q., Reitmayr, G., and Drummond, T. (2009). Proforma: Probabilistic feature-based on-line rapid model acquisition. In *BMVC*, volume 2, page 6. Citeseer.

Pokale, A., Aggarwal, A., Jatavallabhula, K. M., and Krishna, M. (2020). Reconstruct, rasterize and backprop: Dense shape and pose estimation from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 40–41.

Reynolds, D. A. (2009). Gaussian mixture models. *Encyclopedia of biometrics*, 741:659–663.

Rother, C., Kolmogorov, V., and Blake, A. (2004). " grabcut" interactive foreground extraction using iterated graph cuts. *ACM transactions on graphics (TOG)*, 23(3):309–314.

Schonberger, J. L. and Frahm, J.-M. (2016). Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113.

Shunli, W., Qingwu, H., Shaohua, W., Pengcheng, Z., and Mingyao, A. (2018). A 3d reconstruction and visualization app using monocular vision service. In *2018 26th International Conference on Geoinformatics*, pages 1–5. IEEE.

Sobral, A. and Bouwmans, T. (2014). Bgs library: A library framework for algorithm's evaluation in foreground/background segmentation. In *Background Modeling and Foreground Detection for Video Surveillance*. CRC Press, Taylor and Francis Group.

Talbot, J. F. and Xu, X. (2006). Implementing grabcut. *Brigham Young University*, 3.

Thompson, M. M., Eller, R. C., Radlinski, W. A., and Speert, J. L. (1966). *Manual of photogrammetry*, volume 1. American Society of Photogrammetry Falls Church, VA.

Ullman, S. (1979). The interpretation of structure from motion. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 203(1153):405–426.

Yu, R., Russell, C., Campbell, N. D., and Agapito, L. (2015). Direct, dense, and deformable: Template-based non-rigid 3d reconstruction from rgb video. In *Proceedings of the IEEE international conference on computer vision*, pages 918–926.

Zhang, Z., Feng, X., Liu, N., Geng, N., Hu, S., and Wang, Z. (2019). From image sequence to 3d reconstructed model. In *2019 Nicograph International (NicoInt)*, pages 25–28. IEEE.