Active 3D Shape Acquisition Using Smartphones

Jae Hyun Won won1425@gmail.com
Man Hee Lee maninara@gmail.com
In Kyu Park pik@inha.ac.kr

School of Information and Communication Engineering,
Inha University, Incheon 402-751, Korea

Abstract

In this paper, we propose an active 3D shape acquisition method based on photometric stereo using smartphone's camera and flash. A pair of smartphones collaborates as the master and slave, in which the slave projects illumination from different locations while the master captures the images and processes photometric stereo algorithm to reconstruct 3D shape. In order to reduce the error, the smartphone's camera is calibrated to overcome the effect of lens distortion and nonlinear camera sensor response. We apply SURF feature matching and five-point algorithm to estimate the relative pose between the master and slave smartphones. Then the lighting direction is estimated to run photometric stereo algorithm. All procedures are implemented on an off-the-shelf smartphone. Experimental result shows that the proposed system enables us to use smartphone as a 3D shape capturing device with low cost and reasonable quality.

1. Introduction

Recent years have witnessed the explosive growth of 3D multimedia industry which leverage contemporary 3D imaging technology. Common users can easily access immersive 3D visual media through 3D cinema and 3D TV. In computer vision and computer graphics, numerous techniques have been developed to create 3D multimedia contents by capturing 3D shape and motion of existing object. However, it is still hard for common users to create 3D contents without using expensive devices.

Another trend in IT industry is the wide availability of high-performance smartphones, *e.g.* iPhone and Galaxy S. Modern smartphone is a visual computing powerhouse. It has a high speed CPU, high resolution camera, high quality color display, 3D graphic processor, DSP for image and video processing, and several sensors including GPS, compass, and acceleration. In addition to hardware performance, a variety of multimedia applications have been developed on iOS and Android OS, and the demand for new ones is continuously increasing.

It is expected that multimedia applications based on 3D imaging technology is the next mainstream in the near future. In this paper, the core technology of 3D imaging on a mobile device is tackled using smartphone as a platform. Conventional methods of 3D shape acquisition on a smartphone are mostly based on the passive techniques. Lee *et al.* proposed a 3D shape acquisition method based on shape from silhouette [8]. However, this method needs accurate foreground/background segmentation. Hartl *et al.* reconstructed 3D shape of small object on mobile phone using voxel carving technique [6]. Note that both shape from silhouette and voxel carving have difficulty in reconstructing concave shape. Arth *et al.* proposed a localization algorithm using mobile device to reconstruct 3D environment [1]. Pan *et al.* achieved scene reconstruction on a mobile phone from panoramic images using multi-view stereo [10]. However, these methods do not reconstruct dense 3D surface.

Unlike passive techniques, there has been few works on 3D shape acquisition on a mobile device using active techniques. Higo *et al.* proposed a hand-held 3D camera system based on photometric stereo and multi-view stereo [7]. Since it involves a lot of image acquisition and huge amount of computation, it cannot be easily implemented on a mobile device. Schindler obtained 3D facial shape using computer screen's lighting which is extended to a mobile phone using smartphone's screen [13]. Even though this method can obtain 3D facial shape rapidly, it often generates inaccurate result due to low intensity of screen illumination. In general, active techniques provide more accurate and dense shape than passive techniques. However, they need additional devices such as beam projector (structured lighting technique) or illumination (shape from photometric stereo).

In this paper, we propose an active 3D shape acquisition system on mobile smartphone. By observing the functionality of smartphone's camera and LED flash, photometric stereo is employed in our approach. Note that photometric stereo technique is known to produce 3D shape with high

978-1-4673-1612-5/12/$31.00 ©2012 IEEE

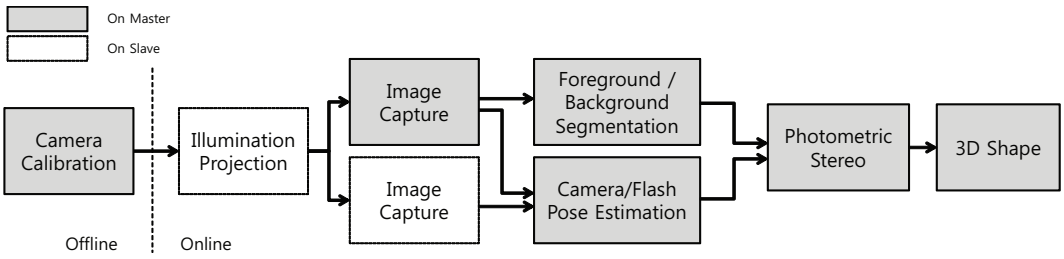


Figure 1. Block diagram of the proposed system.

accuracy in a controlled environment. In order to implement the idea on off-the-shelf smartphone, we use a pair of smartphones which collaborate in imaging and illuminating functionalities. Slave smartphone projects illumination from different position and takes images simultaneously, while master smartphone captures the input images. Given an illumination, SURF features in the images captured by master and slave are matched simultaneously to find the relative pose between them. Then, using conventional procedure of photometric stereo, we reconstruct surface normal map and consequently 3D surface by solving the Poisson equation. To the best of our knowledge, the proposed system is the first one which uses smartphone's camera and flash cooperatively to capture the 3D geometry of the scene. Figure 1 shows the overall block diagram of the proposed system.

This paper is organized as follows. In Section 2, we briefly introduce photometric stereo technique. Section 3 describes the camera calibration and segmentation for image preprocessing. In Section 4, we describe the proposed method used to estimate relative pose between smartphones. Experimental results are shown in Section 5. Finally, we give a conclusive remark in Section 6.

2. Photometric Stereo

2.1. Basic Formulation

Photometric stereo technique observes illuminated 3D scene from different lighting condition (at least 3) and estimates the surface normal based on surface reflectance model [14]. It is assumed that the scene has Lambertian surface without specular reflection. The image pixel intensity I is computed as follows.

$$I = \rho \mathbf{n} \cdot \mathbf{s} \tag{1}$$

where ρ, $\mathbf{n}$, and $\mathbf{s}$ denote albedo, surface normal vector, (known) lighting direction vector, respectively.

In order to make Eq. (1) a well-posed problem, multiple images with different lighting direction are commonly used. Then Eq. (1) becomes an overdetermined system of linear equation as follows.

$$\begin{pmatrix} I_1 \\ \vdots \\ I_N \end{pmatrix} = \rho \mathbf{n} \cdot \begin{pmatrix} \mathbf{S_1} \\ \vdots \\ \mathbf{S}_N \end{pmatrix} \tag{2}$$

Pixel's albedo ρ and surface normal $\mathbf{n}$ can be computed by solving Eq. (2) in least square sense. This procedure is repeated for every pixel position.

2.2. Photometric Stereo Using Smartphone

In our approach, we utilize smartphone's camera and flash as the image acquisition and the light source. A pair of smartphones, *i.e.* master and slave, collaborates together to capture images at fixed location (of the master) and to illuminate the scene from different locations (of the slave). The master performs the main computation of photometric stereo.

In photometric stereo, each lighting direction should be known in advance. In order to achieve this in the proposed system, the slave also captures the image while projecting the illumination. Robust feature (SURF [2]) detection is performed on both images, which are matched and then refined by RANSAC [5]. Given the initial matching of SURF feature, five-point algorithm [9] is employed to estimate the relative pose between master and slave. Note that the distance between camera and flash of slave (2∼3cm) is negligible compared with the distance to the scene. Therefore, in the proposed system, it is assumed that the location of the slave's camera and flash is identical.

After estimating the lighting direction vectors of different illumination location, surface normal is calculated by evaluating Eq. (2). Among a few existing methods of recovering depth map from surface normal map, we use the successive over-relaxation solver [4]. Note that it leads to fast convergence by slightly sacrificing the accuracy.

3. Image Preprocessing

3.1. Camera Calibration

Geometric and radiometric distortion of smartphone's imaging system often degrades the accuracy of recon-

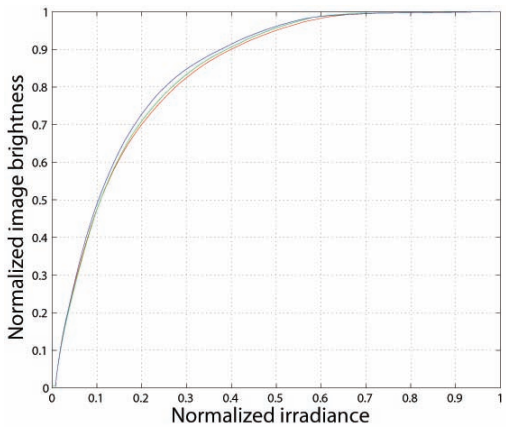


Figure 2. Camera response function of R,G,B channel

structed 3D shape. Geometric (tangential and radial) distortion due to the small lens with short focal length is easily calibrated by performing camera calibration using Camera Calibration Toolbox [3].

On the other hand, radiometric distortion caused by nonlinear sensor response invokes a nontrivial problem since it distorts the observed radiance of scene reflection. Therefore I_i in Eq. (2) remains erroneous if the distortion is not corrected. In our approach, Robertson's HDRI (high dynamic range imaging) method is employed, which uses multiple images with different exposure to estimate the camera response function [11]. Figure 2 shows the estimated camera response function of the smartphone's camera used in our implementation. By taking the inverse of the camera response function, pixel intensity becomes linearly proportional to the observed radiance.

Note that both geometric and radiometric calibration are performed only once in offline processing, which causes no additional burden in on-the-fly computation.

3.2. Foreground/Background Segmentation

In this paper, we focus on 3D shape acquisition of an object of interest. It leads to additional computation and memory consumption if the background pixels are not filtered. Therefore, it is necessary to segment the foreground object from the background.

In our approach, we employ GrabCut [12] method which is an interactive segmentation method. First, user specifies the bounding rectangle around the object to reduce the domain as shown in Figure 3(a). Then, GrabCut gives the initial segmentation (Figure 3(b)). The incorrect initial segmentation is refined by providing user strokes on the foreground (Figure 3(c)) and running GrabCut iteratively. The final result is shown in Figure 3(d).

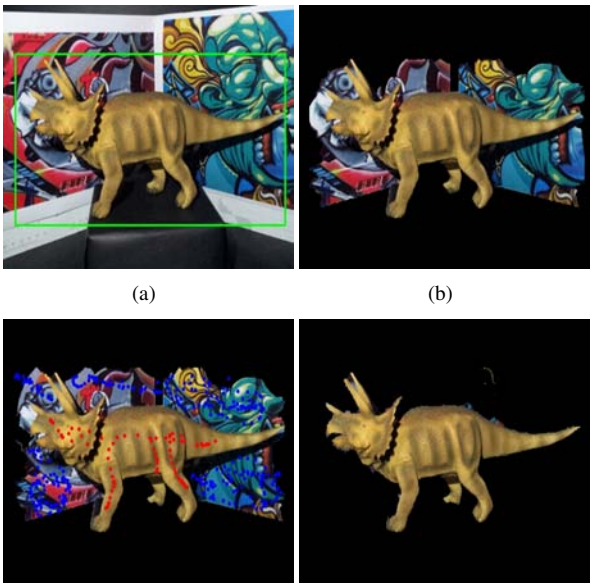


Figure 3. Foreground/background segmentation for the object of interest. (a) Initial user input by specifying a bounding rectangle. (b) Initial segmentation result. (c) Further user input by putting strokes on the foreground. (d) Final segmentation result.

4. Relative Pose Estimation of Smartphones

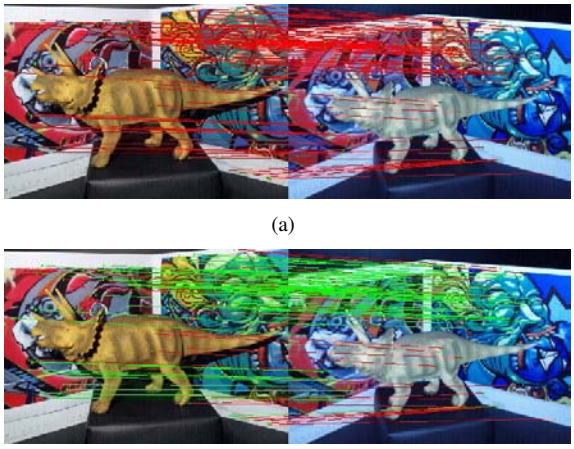
In order to solve Eq. (2), light direction vectors $\mathbf{S_i}$ should be identified in advance. Illumination direction of the slave can be obtained by estimating relative pose between the master and slave smartphones.

4.1. Feature Extraction and Matching

In our approach, pose estimation is performed by extracting and matching SURF features [2] of the images from master and slave. While the slave projects illumination, it also captures the image and computes SURF descriptors. All processes on the slave are triggered by the master's command. The SURF descriptors are transferred to the master through Bluetooth communication. In SURF descriptors' matching, conventional RANSAC is applied to remove outliers. Figure 4(a) shows the matching result of SURF descriptors after applying RANSAC.

4.2. Pose Estimation

The relative pose between the master and the slave, *i.e.* camera and light source, is computed by employing five-point algorithm [9]. The output of five-point algorithm is the essential matrix of images from master and slave. First, five matched feature points are selected randomly and five-point algorithm is executed to find the possible candidates of essential matrix. Among the candidates (usually 1$\sim$10), one of them with minimum estimation error is selected as



(a)

(b)

Figure 4. Initial matching of feature points. (a) Initial SURF matching. (b) After applying RANSAC and five-point algorithm (Inlier matching is marked in green).

the best candidate of the given five points. Before evaluating the estimation error of a particular candidate $\mathbf{E}_i$, the outlier features are removed by computing the matching error and testing if the error is smaller than a threshold τ (which is 5 in our implementation) as follows.

$$\mathbf{q}_2^{\mathrm{T}}\mathbf{K}_2^{-\mathrm{T}}\mathbf{E}_i\mathbf{K}_1^{-1}\mathbf{q}_1 < \tau \tag{3}$$

where $\mathbf{q}_1$ and $\mathbf{q}_2$ are the location of the corresponding feature points in master and slave images. $\mathbf{K}_1$ and $\mathbf{K}_2$ denote the intrinsic matrix of the master and slave camera, respectively. If the feature point pair $\mathbf{q}_1$ and $\mathbf{q}_2$ passes the test, they are selected as inlier. The correspondence of the selected inliers are marked as green in Figure 4 (b). Then the cumulative error ε_i is computed for all inlier points as follows.

$$\varepsilon_i = \sum_{k=1}^{n} \mathbf{q}_{2,k}^{\mathrm{T}}\mathbf{K}_2^{-\mathrm{T}}\mathbf{E}_i\mathbf{K}_1^{-1}\mathbf{q}_{1,k} \tag{4}$$

where n is the number of inlier points. The candidate $\mathbf{E}$ with the minimum cumulative error is selected as the best essential matrix for the selected five points.

This procedure is repeated for as many five-point selections as possible, since outliers may exist in a selection. In our implementation, we test 3,000 random selections of five points. The final essential matrix with the minimum cumulative error is selected as the best one of all different selections.

Finally, the obtained essential matrix $\mathbf{E}$ is decomposed into the multiplication of rotation matrix $\mathbf{R}$ and translation vector $\mathbf{t}(= (\mathrm{t}_x, \mathrm{t}_x, \mathrm{t}_x)^{\mathrm{T}})$ as follows. The decomposition is done by using SVD (singular value decomposition).

$$\mathbf{E} = \mathbf{RS}$$

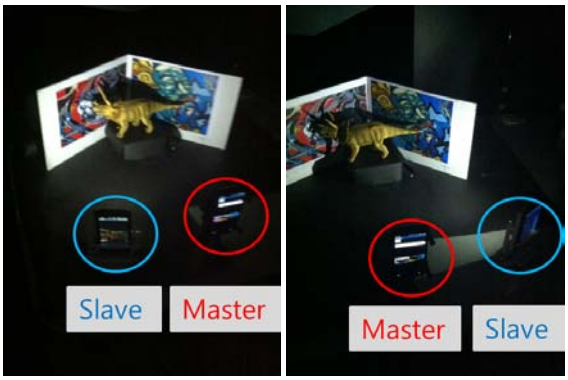


Figure 5. Experimental setup.

Steps	Execution Time	Image Resolution
Segmentation	3.4	640×480
Feature Extraction	5.0	
Feature Matching	1.4	
Pose Estimation	14.7	
Normal Map	1.0	320×240
Depth Map	1.4	
Total	26.7	

Table 1. Execution time (in seconds).

$$\text{where, } \mathbf{S} = \begin{pmatrix} 0 & -\mathrm{t}_z & \mathrm{t}_y \\ \mathrm{t}_z & 0 & -\mathrm{t}_x \\ -\mathrm{t}_y & \mathrm{t}_x & 0 \end{pmatrix} \tag{5}$$

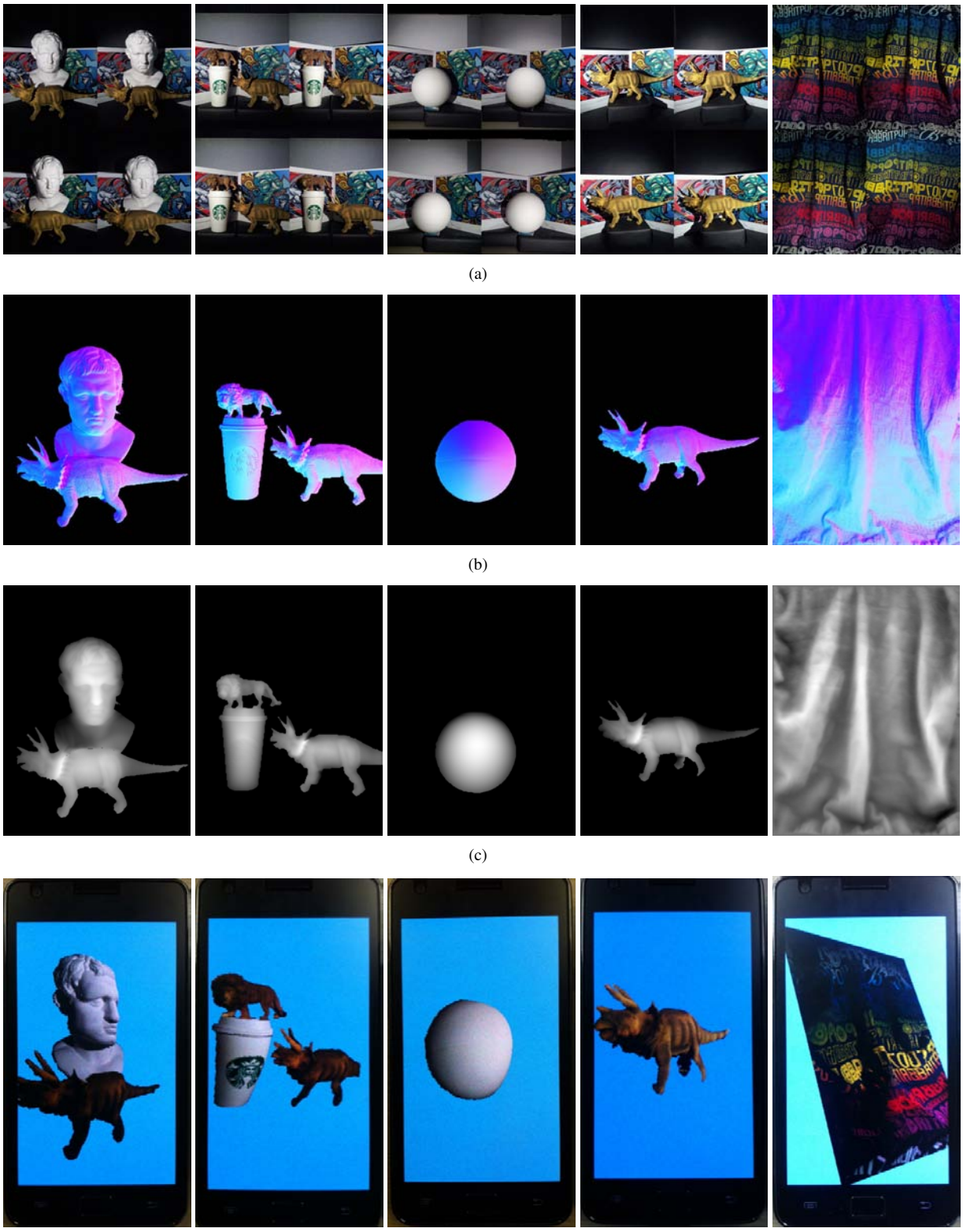
5. Experimental Result

Figure 5 shows the experimental setup of the proposed system. The proposed algorithm is implemented on Samsung SHW-M250S (Galaxy SII) smartphone which equips with 1.2GHz dual core CPU (ARM Cortex A9) and 8M-pixel camera with an LED flash. The algorithm is implemented by C language and JNI (Java Native Interface) using Android NDK on Android 2.3. Although there are many floating point operations, we do not implement fixed-point computation since ARMv7-architecture supports hardware FPU (floating point unit) in Cortex A9 CPU core.

In our experiment, we use ten input images to recover the normal more accurately. The input images have 640×480 resolutions. The resolution of normal map and depth map are decimated to half (320×240) to save the running time.

5.1. Result and Discussion

Figure 6 shows the examples of reconstructed 3D shape for different scenes. It shows that the proposed method produces the normal (Figure 6(b)) and depth map (Figure 6(c)) almost correctly. Note that some degradation like shape bending still exists as observed in Figure 6(d), of which rea-



(a)

(b)

(c)

(d)

Figure 6. Reconstructed 3D shape. (a) Input images. Ten images are used as input. Four of them are shown here. (b) Estimated normal map. (c) Reconstructed depth image. (d) 3D polygon model rendered on the smartphone.

son is summarized as follows. First, on the contrary to the common assumption, the direction of flash light is not parallel, because the distance between the smartphone flash and the object should be close enough. Secondly, the flash light intensity is not uniform in all direction. Thirdly, the camera vignetting effect is not calibrated perfectly by the smartphone's image signal processor (ISP). Nevertheless, the result proves reasonable possibility of 3D shape acquisition using smartphone's camera and flash. The quality of reconstructed 3D shape is better and denser than the conventional binocular stereo or shape from motion techniques.

Table 1 shows the execution time measured for the test scene shown in the first column of Figure 6. In order to reduce the computational burden in feature extraction and matching, the maximum number of SURF features is limited to 300. In depth map estimation from normal map, we apply the successive over-relaxation solver instead of complex method such as conjugate gradient solver. This method not only provides comparable result but also is three times faster than the conjugate gradient solver.

5.2. Limitation

The proposed algorithm has a few limitations which are mainly due to the hardware incapability of the smartphone. First, the flash light is not bright enough in common indoor environment. Therefore, the proposed system requires the ambient illumination to be very dim or eventually turned off. This can be overcome by using brighter flash. Secondly, the current execution time may not be fast enough, while it can be further improved by applying more intensive optimization techniques. The execution time can be much faster than the current one by using parallel computation on mobile GPU. Last, the user interface is uncomfortable due to the small screen size. Despite of these limitations, we believe the proposed system opens the possibility of 3D shape capture by common users using smartphone.

6. Conclusion

In this paper, we proposed an active 3D shape reconstruction method using a pair of smartphones. The proposed method recovered 3D shape successfully by customizing photometric stereo on the smartphone's environment. The proposed system is the first one which uses smartphone's camera and flash cooperatively to capture 3D shape. In future work, we expect to increase the accuracy by combining stereo matching and photometric stereo.

References

[1] C. Arth, D. Wagner, M. Klopschitz, A. Irschara, and D. Schmalstieg. Wide area localization on mobile phones. *Proc. International Symposium on Mixed and Augmented Reality*, pages 73–82, October 2009. 1

[2] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool. Surf: Speeded up robust features. *Computer Vision and Image Understanding*, 110(3):346–359, June 2008. 2, 3

[3] J. Y. Bouguet. *Camera Calibration Toolbox for Matlab*. 3

[4] M. Davis and J. McCammon. Solving the finite difference linearized poission-boltzmann equation: A comparison of relaxation and conjugate gradient methods. *Journal of Computational Chemistry*, 10(3):386–391, April 1989. 2

[5] M. Fischler and R. Bollers. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, June 1981. 2

[6] A. Hartl, L. Gruber, C. Arth, S. Hauswiesner, and D. Schmalstieg. Rapid reconstruction of small objects on mobile phones. *Proc. Seventh IEEE Workshop on Embedded Computer Vision*, pages 20–27, June 2011. 1

[7] T. Higo, Y. Matsushita, N. Joshi, and K. Ikeuchi. A hand-held photometric stereo camera for 3D modeling. *IEEE International Conference on Computer Vision*, pages 1234–1241, September 2009. 1

[8] W. Lee, K. Kim, and W. Woo. Mobile phone-based 3D modeling framework for instant interaction. *Proc. IEEE International Workshop on 3D Digital Imaging and Modeling*, pages 1755–1762, October 2009. 1

[9] D. Nister. An efficient solution to the five-point relative pose problem. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26(6):756–770, June 2004. 2, 3

[10] Q. Pan, C. Arth, G. Reitmayr, E. Rosten, and T. Drummond. Rapid scene reconstruction on mobile phones from panoramic images. *Proc. International Symposium on Mixed and Augmented Reality*, pages 55–64, October 2011. 1

[11] M. A. Robertson, S. Borman, and L. Stevenson. Dynamic range improvement through multiple exposures. *Proc. IEEE International Conference on Image Processing*, 3:159–163, October 1999. 3

[12] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM Trans. on Graphics*, 23(3):309–314, August 2004. 3

[13] G. Schindler. Photometric stereo via computer screen-lighting for real-time surface reconstruction. *Proc. International Symposium on 3D Data Processing, Visualization and Transmission*, June 2008. 1

[14] R. J. Woodham. Photometric method for determining surface orientation from multiple images. *Optical Engineering*, 19(1):139–144, January 1980. 2