The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

Expansion-Based Depth Map Estimation for Multi-View Stereo

Peng Song, Xiaojun Wu, Michael Yu Wang and Jianhuang Wu

Abstract— **This paper presents an algorithm for acquiring high-quality models from multiple calibrated photographs by computing and merging depth maps. The algorithm first computes depth maps from multi-view stereo using a proposed expansion-based approach that returns a 3D point cloud with noisy and redundant information. Then the estimated depth maps are merged into an accurate surface model by a cleaning, downsampling, surface normal estimation and Poisson surface reconstruction process. The proposed approach has been implemented and the experimental results with several real datasets demonstrate that the approach can produce accurate surface models efficiently.**

I. INTRODUCTION

It is important to understand the environment for a robot to interact with the surroundings. The knowledge about the shapes of objects is of particular importance in this context. In recent years, multi-view stereo has been a key ingredient in the acquisition of hight quality object surface model from multiple photographs which can be used to plan robot trajectories, grasp planning or other categories [1][2].

According to the taxonomy of Seitz et al. [3], multi-view stereo algorithms can be generally classified into four classes: 3D volumetric approaches [4] [5] [6], surface evolution approaches [7] [8] [9], feature extraction and expansion techniques [10] [11] and depth map based methods [12] [13] [14]. In practice, depth map based methods are not only easy to implement but also can reconstruct very accurate surface model. Generally, these methods involve two separate stages. Firstly, a depth map is computed for each viewpoint using binocular stereo. Secondly, the depth maps are merged to produce a 3D model. In these methods, the estimation of the depth maps is crucial to the quality of the final reconstructed 3D model.

This paper proposes a novel depth map based multi-view stereo approach. Firstly, depth maps are computed from multi-view stereo using an expansion-based approach which partitions an image into lots of small windows with fixed size and computes a reference depth value for each window, then the depth values of all the pixels in each window are expanded from the reference value. Secondly, an oriented point cloud are computed from the depth maps by a cleaning, downsampling and surface normal estimation process. In practice, for a well-textured object, the oriented point cloud will cover most part of the object surface. Finally, the Poisson Surface Reconstruction (PSR) method [15] is applied to convert the oriented point cloud into a triangulated surface model. Compared with traditional depth map based approaches, the most obvious unique of the proposed algorithm lies in the expansion-based depth map estimation which outputs dense and accurate depth map efficiently.

The paper is organized as follows. In Section 2, we present a brief review of several related works. In Section 3, our multi-view stereo reconstruction method is addressed in details. In Section 4, we present experimental results on several datasets. Finally, in Section 5, we draw some conclusions and present final comments about this work.

II. RELATED WORK

Rossi et al. [16] propose a 3D object reconstruction scheme using a robot arm. In their method, the camera calibration is eliminated because of using the known robot kinematics. Pugeaul et al. [17] employ local multi-modal edge features to represent object 3D information in a robot vision system where only some sparse 3D features are detected rather than a watertight 3D model. In their approach, the robot arm also facilitates the recovery of 3D information.

The inspiration for the approach presented in this paper is the work of Hernandez et al. [7]. They recover a complete model by deforming a mesh, initialized as the visual hull, to find a minimum cost surface in a cost volume which is merged from the depth maps for each viewpoint. In the deformation process, they also incorporate an additional silhouette terms to fuse silhouettes with stereo for reconstruction. In their algorithm, depth maps are computed by backprojecting the ray for each pixel into the visual hull and then reprojecting the depth interval onto neighboring views where window-based correlation is performed. The work presented here improves the depth map estimation approach by introducing an expansion-based approach. And we merge the depth maps by point cloud cleaning and downsampling, normal estimation and surface reconstruction with PSR approach.

Another related work has been recently reported by Goesele et al. [13]. They use a two-step technique. They first use robust window-based matching to compute reliable depth estimates. Then a volumetric method is applied to merge them into a single surface representation. Although their method is simple to implement, their models suffer from a large number of holes and very long processing times.

P. Song is with postgraduate of Shenzhen Graduate School, Harbin Institute of Technology, 518055, China, songpenghit@163.com

X. Wu is the corresponding author and with faculty of the Division of Control and Mechatronics Engineering, Shenzhen Graduate School, Harbin Institute of Technology, 518055, China, wuxj@hitsz.edu.cn

M. Y. Wang is with the Mechanical and Automation Engineering Department, Chinese University of Hong Kong, Shatin, N.T., Hong Kong, China, yuwang@mae.cuhk.edu.cn

J. Wu is with faculty of Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, 518055, China, jh.wu@siat.ac.cn

In contrast, our algorithm is very efficient and reconstructs complete object surface estimates for most well-textured objects.

Our work is similar to that of Bradley et al. [14] who propose to reconstruct an accurate model from multi-view stereo in two steps: binocular stereo matching on image pairs and surface reconstruction from depth maps. The binocular stereo algorithm creates depth maps from pairs of adjacent viewpoints and makes use of scaled window matching to improve the density and precision of depth estimates. And the surface reconstruction step creates a triangular mesh from the depth maps by a downsampling, cleaning and meshing procedure. However, we use different methods to compute depth maps from multi-view stereo. Our algorithm computes depth maps by an expansion-based approach, while Bradley et al. just use the basic binocular stereo matching method to compute depth map for each image.

Finally, recent work by Furukawa et al. [10] proposed a novel algorithm for calibrated multi-view stereo. The algorithm starts by computing a dense set of small rectangular patches covering the surfaces visible in the images and then converts the resulting patch model into an initial mesh model by PSR approach or iterative snapping. Finally, an optional final refinement algorithm is applied to refine the initial mesh to achieve even higher accuracy. In their work, Furukawa et al. compute a dense set of small rectangular patches by a match, expand and filter procedure.

III. Algorithm Description

The proposed multi-view stereo reconstruction approach can be decomposed into three different steps. In the first step, depth maps are computed from multi-view stereo by an expansion-based approach. For the point cloud merged from the estimated depth maps contains lots of outliers and redundant information, we clean and downsample the point cloud in the second step. In the third step, the surface normal of each point in the point cloud is estimated from the positions of the neighbors and PSR approach is applied to convert the oriented point cloud into a triangulated model. The following subsections describe these steps in more details.

A. Expansion-Based Depth Map Estimation

The proposed expansion-based depth map estimation approach is an improvement to the Hernandez et al.'s greedy depth map generation approach [7]. Therefore, we first give a short explanation to the greedy approach. The inputs of the approach are a sequence of calibrated images $I = \{I_0, I_1, .., I_{n-1}\}$ and the visual hull [18] of an object. For each image I_i, the approach selects k neighboring views against which to correlate I_i using robust window matching. For each pixel p in I_i, the approach computes the depth interval from the visual hull of the object which is the back-projected ray of p inside the visual hull. Then reproject the depth interval into selected neighboring views and compute the normalized cross-correlation (NCC) value between an $m \times m$ window centered on p and the corresponding windows

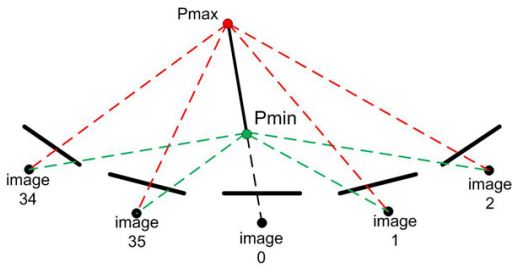


Fig. 1: Backproject the depth interval in the visual hull of an object into 4 neighboring views.

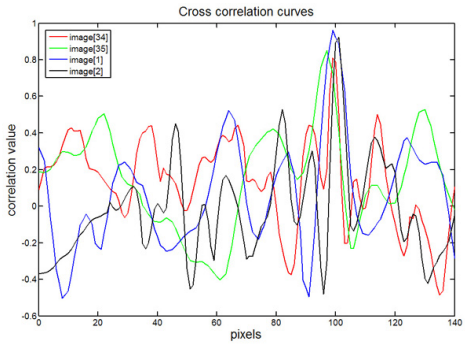


Fig. 2: Cross correlation curves of a pixel in one image of the Soldier sequence with a window size of 11×11 pixels.

centered on the projections in each of the image (see Fig. 1). For a given depth interval, its projection into the different images are all related by the epipolar constraint by which all the correlation curves generated by different views can be related into a single coordinate system [19]. Once the correlation curves are computed, the best candidate depth is chosen from them if its NCC value is larger than some threshold $thres1$ for at least two views in the k neighboring views (see Fig. 2). Note that for each pixel p in I_i, the best candidate depth is chosen to be the value of depth that maximizes NCC value, or none if no valid depth is found. After the best candidate depth is selected, the position of the 3D point corresponding to the pixel p can be computed easily by triangulation method. A detailed description of Hernandez et al.'s greedy depth map estimation approach can be seen in [7].

A drawback of this greedy approach is the computation time since searching the depth value for each pixel from the depth interval defined by the visual hull has a large redundancy of computation. Hernandez et al. speed up the greedy approach by partitioning an image into 3 different resolution layers, computing the depth interval from the visual hull for the lowest resolution layer and from the precedent layer for consecutive layers. In practice, the improvement is about 5 or 6 times faster for well textured images.

Our expansion-based approach extends this redundancy-based idea further. Specifically, our approach first partitions each image into lots of small windows with fixed size $M \times M$, then computes a depth value for the center pixel of each window using greedy approach. If we find a depth value

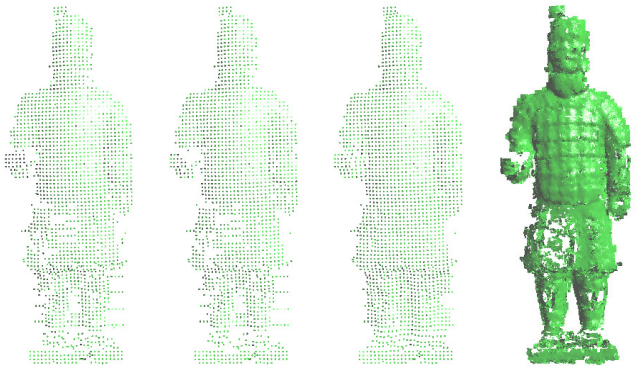


Fig. 3: Expansion-based depth map estimation steps for one image of the Soldier sequence. (a) A sparse point cloud merged from reference depth values. (b) The sparse point cloud after median-rejection process. (c) The sparse point cloud after expanding from neighboring windows. (d) The estimated depth map. From left to right, the number of 3D point in each point cloud is 1910, 1843, 2102 and 182981.

Input: Calibrated image sequence and the visual hull
Output: Depth map of each image
for *each image in imageList* **do**
  for *each expanding window* **do**
    Compute a reference depth value by greedy approach;
  end
  Reject the outliers by median-rejection method;
  for *the window without reference depth value* **do**
    Compute a reference depth value from its 3×3 neighboring windows.
  end
  for *each expanding window* **do**
    Search the depth value for each pixel in the window from the depth interval defined by the reference depth;
  end
end

Algorithm 1: The proposed expansion-based depth map estimation algorithm

whose confident value is larger than a threshold $thres2$, the depth value is taken as a reference depth value for the window. In practice, if we compute the 3D positions for all the reference depth values, we can obtain a sparse point cloud with many outliers (see Fig. 3 (a)). Therefore, a median-rejection method is applied for all the reference depth values of an image to reject the obvious outliers (see Fig. 3 (b)). Since our approach only selects the depth value with a high confident value, there will be many windows without reference depth value, especially for the surface area with little or no texture. Therefore, we compute a reference value for the window without it from its 3×3 neighboring windows, i.e., for a window without a reference depth value, if the number of neighboring windows with it in the 3×3 neighborhood is more than a fixed number a (in all our experimental result, $a = 4$), compute a reference depth value for the window as the median of the depth values of its neighboring windows. This process iterates for 5 times for all the experimental results. In Fig. 3 (c), we can see that the point cloud is more dense after this step. For all the windows of an image, our approach only computes depth value for the pixels of a window with reference depth value. Since each image is a picture of an object and we can expect it to be locally continuous, the depth value of the pixels in each window will not be very different from its reference depth value. Therefore, we search the depth values for all the pixels in the window from a depth interval with fixed length d centered at the reference depth along the optical ray. The depth map of one image of the Solider sequence computed by this approach is demonstrated in Fig. 3 (d) . Our expansion-based depth map estimation algorithm is also described in Algorithm 1.

Since the depth interval defined by the reference depth is much shorter than the depth interval defined by the visual hull, the computation time of the proposed approach is dominated by the reference depth computation step. Typically, the improvement of computation time for an image partitioned into 21×21 windows is around 20 times faster than the greedy approach. The output of this algorithm is depth map for each image. We just merge these depth maps into a point cloud which contains outliers and redundant information. For each 3D point in the point cloud, its confidence value and viewing direction are stored for post-processing.

B. Point Cloud Cleaning and Downsampling

The previous section produces a point cloud with lots of outliers and redundant information. On one hand, the point cloud is with lots of outliers generated for miscorrelation. On the other hand, the point cloud contains large amounts of redundant information due to duplicate reconstructions of parts of the geometry from multiple views. Therefore, we should reject the outliers and downsample the point cloud before surface reconstruction.

The outliers of the depth maps are rejected by a two-step approach. Firstly, incorporate the visual hull of the reconstruction object as a constraint to reject 3D points out of the visual hull. Then, build a voting octee from the estimated point cloud and select a threshold to eliminate miscorrelations. Given a set of point samples S and a maximum tree depth VT_d, the voting octree is the minimal octree with the property that every point sample falls into a leaf node at depth VT_d. Therefore, we build a voting octree for the point cloud which contains, for each voxel, the sum of the individual correlation scores contained in that voxel. This volume can be seen as a volume of surface probability where a voxel with a high score is very probable to contain the real object surface. Therefore, we threshold the voting octree by eliminating the voxels that have relatively lower score values than a threshold value $thres3$ to add robustness to the correlation approach. For input images with 2000×3000 pixels, typical resolutions of the voting octree are between

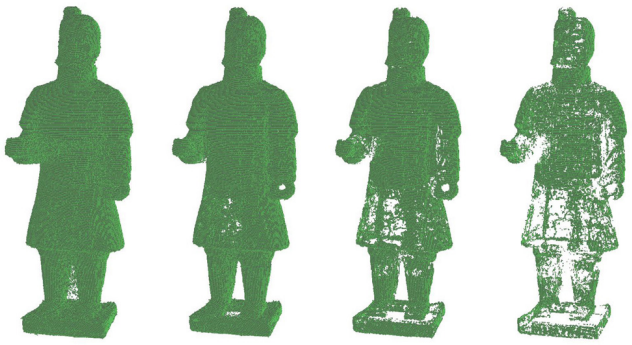


Fig. 4: Voting octree for the Soldier sequence with 10 levels of depth and different thresholds. From left to right, the threshold value is 0, 5, 10, 20.

10 and 11 levels. The different views of voting octree for the Soldier sequence after binarization with different thresholds are presented in Fig. 4.

To downsample the 3D point cloud, for each node at the maximum depth of the voting octree, we extract the point which has largest confidence value in the corresponding voxel. Due to the loss of image space when taking photographs, loss of resolution caused by the size of the correlation window and by the maximum camera baseline, the resolution of a 10-level voting octree is already very high for input images with 2000×3000 pixels. Therefore, for a 10-level voting octree using the point with largest confidence value in a voxel instead of all the points in the voxel will not reduce the accuracy of the estimated depth maps but decrease the size of the point cloud significantly. These extracted 3D points construct a new point cloud on the object surface with few outliers and smaller scale.

C. Normal Estimation and Surface Reconstruction

After the outliers are rejected and the scale of the point cloud are reduced, we need to estimate the exact surface normal for every point. As proposed by Hoppe in [20], the key ingredient of the estimation of surface normals from a point cloud is to associate an tangential plane with each of the data points. This is usually done in two stages. In the first stage, the tangential plane is estimated at each point from the positions of its K-nearest-neighborhood. Thus the orthogonal unitary vector to the tangent plane will be used as an approximation of the normal at such point. The second step is to select the orientation of the tangent plane so as to define a globally consistent orientation for the normal. In most cases, the determination of their orientations is not an easy task. Generally, the complicated minimal spanning tree technique is used to make the consistency of the normals. However, it is not hard to accomplish in our case. Since each estimated data point is known to be visible from a particular viewing direction, the point's tangent plane orientation can be inferred from that viewing direction [14].

After an oriented point cloud on the object surface with few outliers and relatively small scale has been produced, we use freely available package [21] of the PSR method to convert the oriented points into a triangulated mesh model. For the input point cloud with about one million points which is a common case for 10-level voting octree, the output triangulated mesh will have more than half million vertices which is too large for that resolution model. Therefore, we decimate the output mesh to with about 150,000 vertices.

TABLE I: Parameters of the datasets used in our experiments.

	Ima	*Resolution*	*PSize*	*Vertices*	*Time*
Soldier	36	3088×2056	1183749	150000	63.4
Minister	36	3088×2056	1198618	150000	56.2
BigHead	36	2008×3040	1248688	150000	72.1

IV. Experimental Results

To evaluate the contributions of our approach, we demonstrate the reconstructions of three real objects. The Terra Cotta Warrior Minister and Soldier image sequences are acquired in our lab with an electronic turntable and a fixed camera, while the BigHead sequence are courtesy of [22].

Table I lists the number of input images (Ima), their approximate size ($Resolution$), the oriented point cloud scale ($PSize$), the vertex number of the reconstructed model ($Vertices$) and running time in minutes ($Time$) for each dataset. Computation times are dominated by depth maps generation from multi-view stereo step. A typical computation time to reconstruct a surface model from 36 images of 6 Mpixels is about one hour on a Duo E7400 2.80GHz computer. Table II lists some running parameters of the proposed algorithm for each dataset. See the corresponding sections for details about the meaning and setting methods for these parameters.

A. Minister Sequence

The Terra Cotta Warrior Minister (see Fig. 5) is a 274.8mm tall, gray, strongly diffuse object with lots of details. The object is placed on a turntable and a sequence of images is taken with a fixed angle step which is 10 degrees. Therefore, the Minister sequence contains 36 images which all have a resolution of 3088×2056 pixels.

To generate depth map from multi-view stereo, the correlation for a given pixel is computed with 4 neighboring views for a typical sequence of 36 images. Then a 10-level voting octree is built from the depth maps and some outliers generated for miscorrelation are eliminated by thresholding the voting octree ($thres3 = 5.0$). The neighborhood that is used for estimating surface normal for each 3D point is defined by $k = 80$. See Table II for more details of the reconstruction parameters for the Minister sequence.

A complete reconstruction process for this object is presented in Fig. 6. In Fig. 6 (a), we dispose one view of an oriented point cloud computed from multi-view stereo using our approach. And the reconstructed model from the point cloud by PSR method is demonstrated in Fig. 6 (b) which shows that our approach can reconstruct an accurate surface model and correctly recover many minute details such as the

TABLE II: Running parameters for the datasets used in our experiments. See text for more details

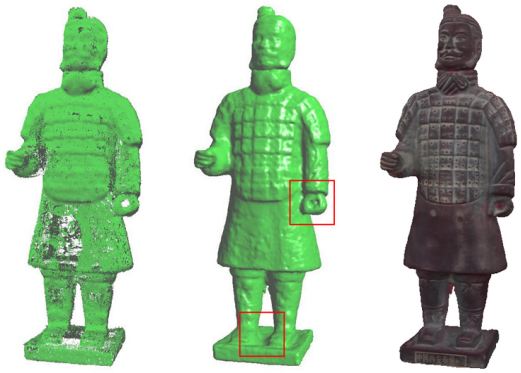
	k	$thres1$	$m \times m$	$M \times M$	$thres2$	$d(mm)$	VT_d	$thres3$	K
Minister	4	0.6	11×11	21×21	3.0	3.0	10	5.0	80
Soldier	4	0.6	11×11	21×21	3.0	3.0	10	6.0	80
BigHead	4	0.6	11×11	21×21	3.0	3.0	10	8.0	120

Fig. 5: Four samples of a total of 36 color images of the Terra Cotta Warrior Minister sequence.

Fig. 7: Four samples of a total of 36 color images of the Terra Cotta Warrior Soldier sequence.

(a) (b) (c)

Fig. 6: The main reconstruction steps for the Minister sequence. (a) Oriented point cloud. (b) Reconstructed surface model. (c) The textured model.



(a) (b) (c)

Fig. 8: The main reconstruction steps for the Soldier sequence. (a) Oriented point cloud. (b) Reconstructed surface model. (c) The textured model.

face and the texture on the chest of the Minister object. In Fig. 6 (c), the model after texture mapping is also presented.

B. Soldier Sequence

The Soldier object (see Fig. 7) is another Terra Cotta Warrior object captured in our lab. The result and running parameters can be seen in Table I and Table II respectively.

In Fig. 8 (a), an oriented pint cloud for the Soldier object is presented which shows that most shape information including some minute details such as the face has been correctly recovered. However, there are still a few surface areas of the Soldier object without estimated 3D points since these areas are with no texture or little texture or cannot be seen from any view. However, one limitation of the PSR method is that it will connect two disconnect regions when there are no samples between these two regions. Since we use the PSR method to reconstruct the surface model from the oriented point cloud directly, the surface area with no or little estimated 3D points may not be correctly recovered. In Fig. 8 (b) and (c), we can see that the structures at the left hand and the support platform of the Soldier object are not correctly recovered for this reason (illustrated by red squares).

C. BigHead Sequence

The BigHead object (see Fig. 9) is a very well textured object which is quite suitable for stereo reconstruction. Since the object support table is lack of texture which cannot be recovered using our method, we eliminate the table in the visual hull.

The complete reconstruction steps are demonstrated in Fig. 10. Since the body of the BigHead object is well textured, a quite uniform point cloud can be estimated (see Fig. 10 (a)). In Fig. 10 (b) and (c) we present the comparison between the surface model reconstructed by the proposed algorithm and the model reconstructed by Hernandez et al. which shows they have comparable accuracy. The textured model is also presented in Fig. 10 (d). Thanks to the expansion-based depth map estimation method, our approach is also very efficient. The running time of this dataset is 72.1 minutes (see Table

Fig. 9: Four samples of a total of 36 color images of the BigHead sequence.

(a) (b) (c) (d)

Fig. 10: The main reconstruction steps for the BigHead sequence. (a) Oriented point cloud. (b) and (c) Reconstructed surface models by our and Hernandez et al.'s approaches. (d) The textured model.

I) which is fast for 36 input images with 2008×3040 pixels.

V. Conclusions

We have developed a multi-view stereo algorithm based on computing and merging depth maps in which an expansion based scheme is utilized to accelerate the depth map generation. The experimental results with several real datasets demonstrate that the proposed approach can produce accurate surface models efficiently. It is claimed that the accuracy of our new algorithm can be comparable with some state-of-the-art techniques by reconstructing two datasets provided by Middlebury benchmark [23]. The calibration step in our multi-view stereo reconstruction scheme can also be replaced by robot arm kinematics.

Due to the window matching need to done with more than two neighboring views, the current implementation is targeted at small baseline image sequences to work well. Future work includes the integration of a wide-baseline matching scheme. Our future work will be also aimed at obtaining even higher accurate surface model by incorporating a subpixel optimization process for the estimated depth maps. For the limitation of the proposed approach mentioned in the work, we are considering to combine shape from silhouette approach with the present technique in order to reconstruct complete and accurate surface models with correct topology.

VI. Acknowledgments

This project is partially supported by Natural Science Foundation of China (NSFC50805031) and Science & Technology Basic Research Projects of Shenzhen (No. JC200903120184A, ZYC200903230062A), Foundation of the State Key Lab of Digital Manufacturing Equipment & Technology (No. DMETKF2009013).

References

[1] Z. Xue, A. Kasper, M. Zöllner, and R. Dillmann, "An automatic grasp planning system for service robots", in *Proceedings of the 14th International Conference on Advanced Robotics*, 2009, pp. 22-26.

[2] Z. C. Marton, L. Goron, R. B. Rusu, and M. Beetz, "Reconstruction and verification of 3D object models for grasping", in *Proceedings of the 14th International Symposium on Robotics Research*, Lucernce, Switzerland, 2009.

[3] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski, "A comparison and evaluation of multi-view stereo reconstrucion algorithms", in *CVPR*, vol. 1, 2006, pp. 519-526.

[4] G. Vogiatzis, P. Torr, and R. Cipolla, "Multi-view stereo via volumetric graph-cuts", in *CVPR*, 2005, pp. 391-398.

[5] G. Vogiatzis, C. Hernandez, P. H. Torr, and R. Cipolla, "Multi-view stereo via volumetric graph-cuts and occlusion robust photo-consistency", in *IEEE Trans. on PAMI* , vol. 29, no. 12, 2007, pp. 2241-2246.

[6] S. N. Sinha, P. Mordohai, and M. Pollefeys, "Multi-view stereo via graph cuts on the dual of an adaptive tetrahedral mesh", in *ICCV*, 2007, pp. 1-8.

[7] C. Hernandez and F. Schmitt, "Silhouette and stereo fusion for 3d object modeling", in *Computer Vision and Image Understanding*, vol. 96, no. 3, 2004, pp. 367-392.

[8] K. Kutulakos and S. M. Seitz, "A theory of shape by space carving", in *IJCV*, vol. 38, no. 3, 2000, pp. 199-218.

[9] J. Pons, R. Keriven, and O. Faugeras, "Modelling dynamic scenes by registering multi-view image sequences", in *CVPR*, 2005, pp. 822-827.

[10] Y. Furukawa and J. Ponce, "Accurate, Dense, and Robust Multi-View Stereopsis", in *CVPR*, 2007.

[11] M. Habbecke and L. Kobbelt, "A surface-growing approach to multi-view stereo reconstruction", in *CVPR*, 2007.

[12] P. Narayanan, P. Rander, and T. Kanade, "Constructing virtual worlds using dense stereo", in *ICCV*, 1998, pp. 3-10.

[13] M. Goesele, B. Curless, and S. M. Seitz, "Multi-view stereo revisited", in *CVPR*, 2006, pp. 2402-2409.

[14] D. Bradley, T. Boubekeur, and W. Heidrich, "Accurate multi-view reconstruction using robust binocular stereo and surface meshing", in *CVPR*, 2008, pp. 1-8.

[15] M. Kazhdan, M. Bolithp, and H. Hoppe, "Poisson surface reconstruction", in *Eurographics Symposium on Geometry Processing*, 2006, pp. 61-70.

[16] C. Rossi, S. Savino, and S. Strano, "3D object reconstruction using a robot arm", in *Proceedings of the Second European Conference on Mechanism Science*, 2008, pp. 513-521.

[17] N. Pugeault, E. Baseski, D. Kraft, F. Wörgötter, and N. Krüger, "Extraction of muliti-modal object representations in a robot vision system", in *Internaional Conference on Computer Vision Theory and Applications*, 2007.

[18] A. Laurentini, "The Visual Hull Concept for Silhouette Based Image Understanding", in *IEEE Trans. on PAMI*, vol.16, no.2, 1994, pp. 150-162.

[19] C. Hernandez and F. Schmitt, "Multi-stereo 3d object reconstruction", in *3DPVT*, 2002, pp. 159-166.

[20] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle, "Surface reconstruction from unorganized points", in *SIGGRAPH*, 1992, pp. 71-78.

[21] M. Kazhdan, M. Bolithp, and H. Hoppe. Poisson Reconstruction Package. Available: `http://www.cs.jhu.edu/~misha/Code/PoissonRecon/`

[22] C. Hernandez and F. Schmitt. BigHead dataset. Available: `http://www.tsi.enst.fr/3dmodels/`

[23] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. Dino and Temple datasets. `http://vision.middlebury.edu/mview/`