ACCURATE MULTIVIEW STEREO RECONSTRUCTION WITH FAST VISIBILITY INTEGRATION AND TIGHT DISPARITY BOUNDING

R. Toldo[a], F. Fantini[a], L. Giona[a], S. Fantoni[a] and A. Fusiello[b]

[a] 3Dflow srl, Strada le grazie 15, 37134 Verona, Italy -
{roberto.toldo, filippo.fantini, luca.giona, simone.fantoni}@3dflow.net
[b] Dipartimento di Ingegneria Elettrica, Gestionale e Meccanica, University of Udine,
Via Delle Scienze, 208 - 33100 Udine, Italy -
andrea.fusiello@uniud.it

KEY WORDS: Multiview Stereo, Image-based Modelling, Surface Reconstruction, Structure from Motion

ABSTRACT:

A novel multi-view stereo reconstruction method is presented. The algorithm is focused on accuracy and it is highly engineered with some parts taking advantage of the graphics processing unit. In addition, it is seamlessly integrated with the output of a structure and motion pipeline. In the first part of the algorithm a depth map is extracted independently for each image. The final depth map is generated from the depth hypothesis using a Markov random field optimization technique over the image grid. An octree data structure accumulates the votes coming from each depth map. A novel procedure to remove rogue points is proposed that takes into account the visibility information and the matching score of each point. Finally a texture map is built by wisely making use of both the visibility and the view angle informations. Several results show the effectiveness of the algorithm under different working scenarios.

1 INTRODUCTION

The goal of Multi-view Stereo (MVS) is to extract a dense 3D surface reconstruction from multiple images taken from known camera viewpoints. This is a well studied problem with many practical and industrial applications. Laser scanners yield to very accurate and detailed 3D reconstructions. However, they are based on expensive hardware, difficult to carry and rather complex to set, especially for large-scale outdoor reconstructions. In all these cases, MVS can be applied successfully.

In this paper we present a novel multiview stereo method. The algorithm is focused on accuracy and it is highly engineered with some parts taking advantage of the GPU. In addition, it is seamlessly integrated to the output of an underlying structure and motion (SaM) pipeline (Gherardi et al., 2011). As a matter of fact, sparse structure endowed with visibility information is very reliable and can improve both speed and accuracy of a MVS algorithm by reducing the search space and the ambiguities. Following (Campbell et al., 2008) a number of *candidate depths* are first extracted for each pixel of the image. These hypothesis are used as input of a Markov Random Field (MRF) optimization to extract a final depth map. Votes for each depth map are accumulated on a discrete 3D volume and an iterative technique based on visibility is employed to remove spurious points. Finally, a mesh is generated using the Poisson reconstruction algorithm (Kazhdan et al., 2006) and textures are applied on it.

The paper is structured as follows: in Section 2 the most relevant multiview stereo techniques are reviewed, in Section 3 the method is presented and in Section 4 results are shown on challenging and real datasets. Finally in Section 5 conclusions are drawn.

2 PREVIOUS WORK

The problem of reconstructing a 3D scene from multiple views, have been tackled by many researchers. In (Seitz et al., 2006) several multiview stereo algorithms are presented and a taxonomy is drawn. According to the authors, six fundamental properties differentiate MVS algorithms: reconstruction algorithm, scene representation, photoconsistency measure, visibility model, shape prior and initialization requirements. We will follow this general taxonomy to present the evolution of multiview stereo algorithms.

According to (Seitz et al., 2006), there are mainly four classes of multiview stereo techniques. In the first one, a surface is generated by the definition of a cost function directly on a 3D volume (Seitz and Dyer, 1999, Treuille et al., 2004). Several heuristics can be carried out to extract the surface. Some approaches extract an optimal surface by defining a volumetric MRF (Roy and Cox, 1998, Vogiatzis et al., 2005, Sinha and Pollefeys, 2005, Furukawa, 2008, Kolmogorov and Zabih, 2002). A second class of algorithms is composed by methods that iteratively find an optimal surface by minimizing a cost function. Space carving is a popular technique that falls into this category. An initial conservative surface is defined to contain the entire scene volume and it is iteratively modeled by carving away portion of volumes considering visibility constraints (Kutulakos and Seitz, 2000, Slabaugh et al., 2004). The third category is composed by methods that compute a depth map for each view (Szeliski, 1999a, Kolmogorov and Zabih, 2002, Gargallo and Sturm, 2005). These depth maps can be merged as a post process stage (Narayanan et al., 1998). The fourth class is composed by algorithms that, instead of performing dense matching for each pixel, extract and match a subset of feature points for each image and then fit a surface to the reconstructed features (Manessis et al., 2000, Taylor, 2003).

The scene can be represented in many ways in a reconstruction pipeline. Many times it is represented by a discrete occupancy function (e.g. voxels) (Fromherz and Bichsel, 1995, Vogiatzis et al., 2005) or a function encoding distance to the closest surface (e.g. level sets) (Faugeras and Keriven, 2002, Pons et al., 2005). Some algorithms represent the scene as a depth map for each input view (Szeliski, 1999a, Drouin et al., 2005). The different depth maps can be merged into a common 3D space at a later stage. Other algorithms make use of polygon meshes to represent the surface as a set of planar polygons. This representation can be used in the central part of a reconstruction pipeline since it is well-suited for visibility computation (Fua and Leclerc, 1995) or it can be computed in the final part, starting from a dense points cloud (Kazhdan et al., 2006, Hiep et al., 2009, Vu et al., 2012). Many modern algorithms employ different representation over their reconstruction pipeline. Our algorithm, for example, falls into this category.

According to (Seitz et al., 2006), photoconsistency measures may be defined in scene space or image space. When photoconsistency is defined in *scene space*, points or planar polygons are moved in 3D and the mutual agreement between their projections on images is evaluated. This can be done by using the variance (Seitz and Dyer, 1999, Kutulakos and Seitz, 2000) or a window matching metric (Jin et al., 2003, Hernández Esteban and Schmitt, 2004). This is the prevalent case and also our algorithm implicitly works in scene space. Normalized cross correlation is the most commonly used window based matching metric. In contrast, *image space* methods use the scene geometry to create a warping between images at different viewpoints. The photoconsistency measure is given by the residual between the synthetic and original views (Pons et al., 2005, Szeliski, 1999b). Some algorithms use also silhouettes (Fua and Leclerc, 1995, Sinha and Pollefeys, 2005, Hernández Esteban and Schmitt, 2004) or shadows (Savarese et al., 2001) to enhance the reconstruction. However, this techniques are very sensitive to light changes.

A visibility model is used to specify which views to consider when matching regions or pixels using photo consistency. A common approach is to use a predicted estimation of the geometry to determine the visibility model (Fromherz and Bichsel, 1995, Kutulakos and Seitz, 2000, Kutulakos, 2000, Vogiatzis et al., 2005, Hernández Esteban and Schmitt, 2004, Sinha and Pollefeys, 2005). Other methods simply use clusters of nearby cameras (Hernández Esteban and Schmitt, 2004, Savarese et al., 2001), or employ different heuristics to detect outliers views and do not consider them during reconstruction(Hernández Esteban and Schmitt, 2004, Kang et al., 2001, Gargallo and Sturm, 2005, Drouin et al., 2005). Instead, our method exploits the robust visibility information coming from a structure and motion pipeline.

Shape priors are often implicitly or explicitly imposed to the generated surface in order to bias the reconstruction to have desired characteristics. Some methods search for a *minimal surface* either imposing to start from a gross initial shape, by smoothing points with high-curvatures (Tasdizen and Whitaker, 2004, Diebel et al., 2006, Sinha and Pollefeys, 2005) or by imposing planarity (Fua and Leclerc, 1995, Furukawa et al., 2009). Other methods implicitly search for a *maximal surface* since they does not impose any surface smoothness (Fromherz and Bichsel, 1995, Seitz and Dyer, 1999, Kutulakos, 2000, Kutulakos and Seitz, 2000, Treuille et al., 2004, Saito and Kanade, 1999). Finally some approaches optimize an image-based smoothness terms (Szeliski, 1999a, Kang et al., 2001, Kolmogorov and Zabih, 2002, Gargallo and Sturm, 2005, Campbell et al., 2008). This kind of prior fits nicely into 2D MRF solvers.

In addition to a set of calibrated images, many algorithm require additional information on the scene to bound the reconstruction. Usually this is done by defining a rough bounding box (Kutulakos and Seitz, 2000, Kutulakos, 2000, Vogiatzis et al., 2005, Campbell et al., 2008) or by simply limiting the range of disparity values in *image space* methods (Szeliski, 1999a, Kolmogorov and Zabih, 2002, Gargallo and Sturm, 2005).

In recent years, many algorithms have shifted the focus on large scale reconstructions. This is a challenging problem, since dealing more data leads to computational and robustness problems. In (Hiep et al., 2009) the large scale reconstruction problem is solved by defining a minimum s-t cut based global optimization that transforms a dense point cloud into a visibility consistent mesh followed by a mesh-based variational refinement that captures small details, smartly handling photoconsistency regularization and adaptive resolution. The computation can be carried on the graphics processing unit (GPU) (Vu et al., 2012), knocking down the computing times. Multi-view stereo algorithms are well-suited for general purpose GPU programming (GPGPU), since many of their tasks can be executed in parallel. In (Furukawa et al., 2010) the large scale problem is solved with a *divide et impera* solution. The collection of photos are decomposed into overlapping sets that can be processed in parallel, and finally merged. The algorithm have been successfully tested on datasets with over ten thousand images, yielding a 3D reconstruction with nearly thirty million points.

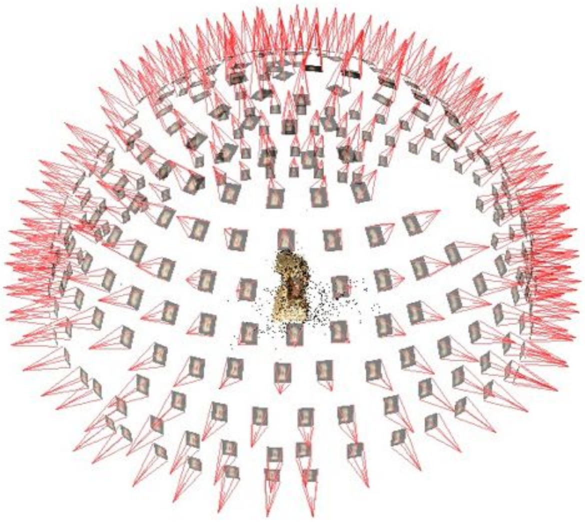


Figure 1: An example of structure and motion (cameras) produced by Samantha.

3 METHOD

In this section a novel multiview stereo method will be presented. The algorithm resembles (Campbell et al., 2008) in some parts since it uses a similar MRF depth map optimization at its core. Some parts have been implemented to run on GPU.

The focus of the method is on accuracy and on tight integration with our structure and motion pipeline "Samantha" (Gherardi et al., 2011), which produces a sparse cloud of 3D *keypoints* (the "structure") and the internal and external parameters of the cameras (the "motion"); see Fig.1. The method is completely automatic, as no user input is required.

3.1 Extraction of depth hypothesis

The goal of this phase is to extract a number of *candidates depths* for each pixel $\mathbf{m}$ and for each image I_i. These hypothesis will be later used as labels in a MRF that extracts the final depth map $\delta_i(\mathbf{m})$. Similarly to many multiview stereo algorithms, a pixel-level matching along epipolar lines is used, with Normalized Cross Correlation (NCC) as the matching metric, which gives a good tradeoff between speed and robustness to photometric nuisances.

Every depth map is created independently from the others. The extraction of candidate depths is performed by considering the reference image I_i and a number (we used three) of neighboring views $\mathcal{N}(I_i)$. The choice of the near views can be critical. To obtain as much information as possible one should be assured that the neighbor view are viewing the very same part of the scene. This is nearly impossible to estimate without an a-priori knowledge of the scene.

To solve this problem, we leverage on the sparse structure and the visibility information provided by Samantha, with a simple "overlap" measure based on the Jaccard index:

$$d_J(I_1, I_2) = \frac{|\mathcal{V}(I_1) \cap \mathcal{V}(I_2)|}{|\mathcal{V}(I_1) \cup \mathcal{V}(I_2)|} \tag{1}$$

where I_1 and I_2 are two images and $\mathcal{V}(I)$ is the set of 3D keypoints visible in image I. By choosing the three views with the highest overlap measure with I_i we are implicitly guaranteed that they are close to I_i and looking at the same part of the scene.

The candidate depths for each pixel are searched along the optical ray, or equivalently, along the epipolar line of each neighboring image using block matching and NCC. In this way, a correlation profile $C_j(\zeta)$, parameterized with the depth ζ, is computed for every pixel $\mathbf{m}$ and every neighbor image $I_j \in \mathcal{N}(I_i)$.

As suggested by (Campbell et al., 2008) candidates depth correspond to local peaks of the correlation (peaks with a NCC value lower than 0.6 are discarded). In principle:

$$\delta_i(\mathbf{m}) = \arg\operatorname{localmax}_\zeta C_j(\zeta) \quad j \in \mathcal{N}(i) \tag{2}$$

where localmax is an operator that returns a fixed number of local maxima. In practice, each of the local peaks of $C_j(\zeta)$ casts a vote (weighted by its score value) on a discrete histogram along the optical ray. At the end of the process, the k bins of the histograms with the highest score are retained (we used $k = 5$) as the candidate depths for the pixel. These k candidate depths for a point $\mathbf{m}$ are stored in the map δ and the corresponding correlation values in the map γ.

The number of histogram bins can be critical to the accuracy of the algorithm. In order to avoid any loss of fine details, we keep track of the depth inside each bins using a moving average approach, where the weight is given by the match score itself.

3.1.1 Depth Range estimation The search range of each pixel depth can heavily impact the performance of the algorithm: an effective heuristic to delimit the search range improve both the running times and the candidate depths estimates.

Some algorithms assume the depth range to be known, but this assumption does not hold in many real cases. The search range can be limited by approximating a global surface or independently for each pixel.

A first order approximation is represented by a bounding volume, which can be readily extracted from the SaM point cloud. The intersection of the optical ray with the volume is easy to compute, but the resulting search range on the optical ray can be still too wide.

In order to limit further the search range of the candidate depth, we compute the boundary independently for each pixel by using the information coming from the structure and motion. For each image pixel we consider the five closest keypoints that have been reconstructed by Samantha. Let $\mathbf{x}$ denote the corresponding 3D keypoint position and let $\mathbf{x}_o$ be its projection on the optical ray of the pixel. The search range along the optical ray is defined as the union of the (five) intervals with center in $\mathbf{x}_o$ and radius $||\mathbf{x} - \mathbf{x}_o||$. An example is shown in Fig. 2.

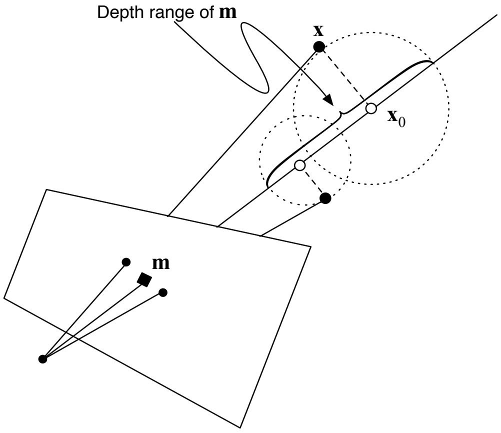


Figure 2: The depth range of a pixel $\mathbf{m}$ (black square) is estimated using the closest keypoints (black circles). Every keypoint have a corresponding 3D point $\mathbf{x}$, which projects onto the optical ray of $\mathbf{m}$ at $\mathbf{x}_0$ and produces a depth interval of radius $||\mathbf{x} - \mathbf{x}_0||$ centered at $\mathbf{x}_0$. The union of these intervals is the depth range of $\mathbf{m}$.

3.1.2 Rectification Rectification is a widely used technique in stereo analysis, however it is not very common in the multiple view framework. Given two views, rectification forces the epipolar line to be parallel and horizontal (Fusiello et al., 2000). The idea behind rectification is to define two new camera matrices which preserve the optical centers but with image planes parallel to the baseline. The computation of the correlation window on horizontal lines avoid the need of bilinear interpolation, and lends itself easily to GPU implementation.

The images in the rectified space are linked to the original image space by a 2D homography. As a consequence, any point on the epipolar line is linked by a 1D homography to the same epipolar line expressed in the original space. The general multiview stereo framework is thus unchanged; matching is performed in rectified space and transformed back in original space by means of an homography.

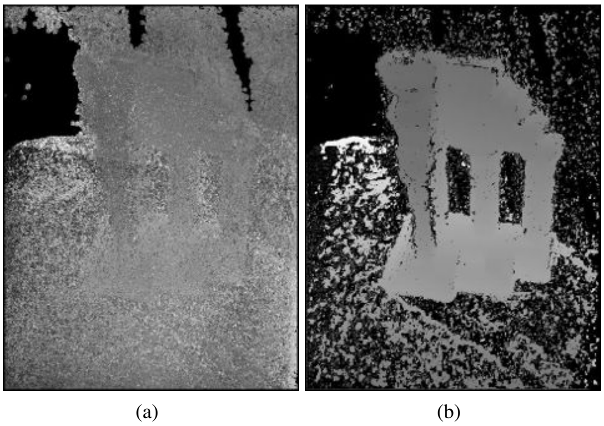


Figure 3: An example of depth map before (a) and after (b) the MRF optimization. The map (a) shows the the candidate depth with the best score.

3.2 Depth Map generation

The final depth map is generated from the depth hypothesis using a discrete MRF optimization technique over the (regular) image grid. The MRF assigns a label $l \in \{l_1 \dots l_k, l_{k+1}\}$ to each pixel $\mathbf{m}$, where the first k labels correspond to the candidate depths and l_{k+1} is the *undetermined* state. The cost function to be minimized consist – as customary – of an unary function E_{data} that depend on the value at the pixel and a smoothness term E_{smooth} that depends on pairwise interaction.

The smoothness term is modeled as described in (Campbell et al., 2008), whereas the data term is based on (Ganan and McClure, 1985):

$$E_{\mathrm{data}}(\mathbf{m}, l) = 1 - \frac{{}^{l}\gamma_i(\mathbf{m})^2}{{}^{l}\gamma_i(\mathbf{m})^2 + |\mathcal{N}(i)|} \tag{3}$$

where ${}^{l}\gamma(\mathbf{m})$ is the NCC peak score of the l^{th} candidate depth for pixel $\mathbf{m}$ — as explained in Section 3.1 – and $|\mathcal{N}(i)|$ is the number of neighboring views of I_i. The undetermined label is given a fixed score of 0.4. With respect to the original formulation, the Geman-McClure score function improve further the robustness against spurious matches.

The MRF is solved with a sequential tree-reweighted message passing optimization (Kolmogorov, 2006) that we implemented in CUDA. An example of depth map optimization is shown in Fig. 3.

3.3 Visibility accounting

Depth maps are lifted in 3D space to produce a *photoconsistency* volume φ, represented by an octree that accumulates the scores coming from each depth map δ_i.

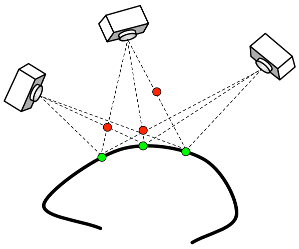
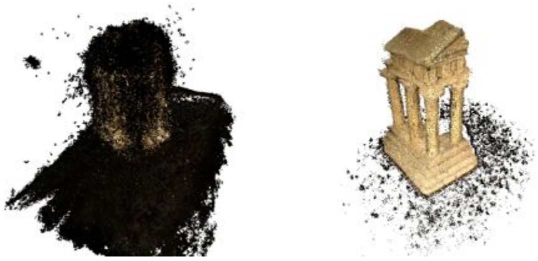


Figure 4: Rogue points (red) can be identified as occlusors of actual surface points (green).

In order to avoid any loss of accuracy, a moving average approach have been used inside each bin. At the end of the lifting process, each cell $\mathbf{x}$ contains a 3D point position $\mathrm{pos}(\mathbf{x})$ - which can be shifted with respect to the cell center - and a photoconsistency value $\varphi(\mathbf{x})$ given by the sum of the correlation scores of the points that fall in that bin.



(a)

(b)

Figure 5: An example of photoconsistency volume (with color) before (a) and after (b) the spurious points removal.

The photoconsistency volume at this stage contains a lot of spurious points, which do not belong to a real surface (see Fig. 5.a for an example). They are characterized by two features: i) their photoconsistency is generally lower than actual surface points, and ii) they usually occludes actual surface points (Fig. 4).

This observation leads to an iterative strategy where the photoconsistency of an occlusor is decreased by a fraction of the photoconsistency of the occluded point. Points with negative photoconsistency are eventually removed. The procedure is summarized in Algorithm 1. An example of rogue points removal is shown in Fig. 5.

Algorithm 1 VISIBILITY SPURIOUS POINTS REMOVAL

```
Input: photoconsistency map $\varphi(\mathbf{x})$
Output: photoconsistency map $\varphi(\mathbf{x})$

1. For each image $I_i$:
   (a) Project each point $\mathbf{x}$ s.t. $\varphi(\mathbf{x}) > 0$ on image $I_i$.
   (b) Group projected points in pixel cells and order them by depth.
   (c) For each cell:
       i. let $\mathbf{x}_k$ be the point with the highest $\varphi(\mathbf{x})$ visible by image $I_i$.
       ii. for each point $\mathbf{x}$ occluding $\mathbf{x}_k$:
           $\varphi(\mathbf{x}) \leftarrow \varphi(\mathbf{x}) - \varphi(\mathbf{x}_k)/|\mathcal{V}(\mathbf{x}_k)|$
2. Remove points $\mathbf{x}$ s.t. $\varphi(\mathbf{x}) < 0$.
3. Remove isolated points.
4. Iterate through steps 1,2,3 until no more points are removed.
```

Algorithm 2 MULTIVIEW STEREO

```
Input: $N$ images $I_1 \dots I_N$
Output: photoconsistency map $\varphi(\mathbf{x})$

1. Initialize $\varphi(\mathbf{x}) = 0$
2. For each image $I_i$, build the depth map $\delta_i$ as follows:
   (a) for each point $\mathbf{m} \in I_i$,
       i. for each $I_j$ with $j \in \mathcal{N}(i)$ (neighborhood of $I_i$)
       ii. compute $C_j(\zeta)$, the NCC of $\mathbf{m}$ along its epipolar line in $I_j$,
       iii. compute depths candidates for $\mathbf{m}$: $\delta_i(\mathbf{m}) = \arg\mathrm{localmax}_\zeta C_j(\zeta)$ with $j \in \mathcal{N}(i)$,
       iv. record the correlation score of the candidates in: $\gamma_i(\mathbf{m}) = C_j(\delta_i(\mathbf{m}))$ for some $j$.
   (b) assign a unique depth to every point of $I_i$, by MRF relaxation of the depth map $\delta_i$, and update $\gamma_i$ accordingly.
3. For each depth map $\delta_i$, lift it in 3D space as follows
   (a) for each point $\mathbf{m} \in \delta_i$,
       i. $\varphi(\mathbf{x}) = \varphi(\mathbf{x}) + \gamma_i(\mathbf{m})$ where $\mathbf{x}$ is the point at depth $\delta_i(\mathbf{m})$ along the optical ray of $\mathbf{m}$ in $I_i$.
4. Remove spurious points using visibility (Algorithm 1),
5. Compute approximate normal at each point,
6. Run Poisson surface reconstruction.
```

At the end of the process a surface is generated using the Poisson algorithm (Kazhdan et al., 2006). A normal for each 3D point is computed by fitting a plane using the closest neighbors. Normal direction is disambiguated with visibility.

Finally a texture map is built by wisely making use of both the visibility and the view angle information.

The overall procedure is summarized in Algorithm 2. As an example, Fig. 6 shows the reconstructed surface of a marble artifact from 7 pictures of size ≈ 15 Mpixels.

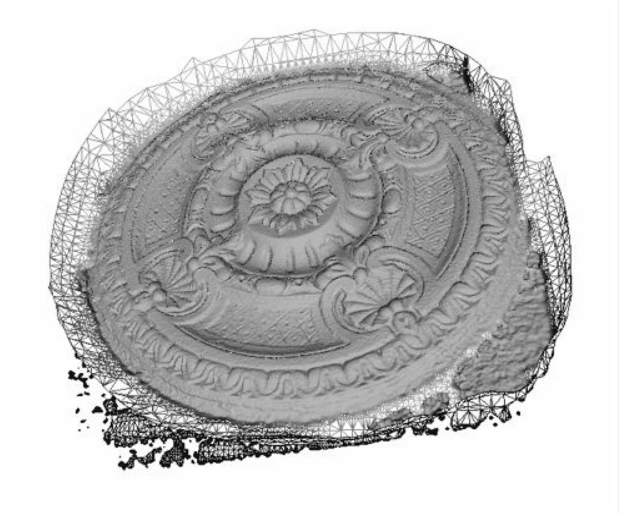


Figure 6: Reconstructed surface of a marble artifact. Please observe how the fine details are reproduced.

4 RESULTS

In order to test our algorithm, we run it on several real-cases datasets from the MVS benchmark presented in (Strecha et al., 2008). The datasets are public and can be downloaded from the author website[1]. They are composed of challenging urban scenes. In particular, a fountain, a church gate and a yard have been captured. The number of images ranges from a minimum of 8 to a maximum of 30, the size being ≈ 6 Mpixels.

The results are reported in Figs. 7, 8. Although camera matrices were already available, we run Samantha with known internal parameter to produce structure and camera matrices.

From a qualitative point of view, our method matches the best results of the benchmark. Unfortunately, at the time of writing, the benchmarking service was not available, so we cannot show a quantitative comparison.

The running time of the method is linear with respect to the number of views. On the average, it took about 10 minutes to generate candidate depth hypothesis for each depth map and 30 seconds to compute the MRF optimization. The rogue points removal based on visibility took from 5 to 10 minutes to compute, while the mesh generation with took Poisson from 10 to 20 minutes. All the experiments were carried out on a entry level machine equipped with a Quad-Core 3.2Ghz CPU and a GeForce GTX 460 GPU.

5 DISCUSSION

In this work we presented a novel multiview stereo algorithm that fully takes advantage of the output of a structure and motion pipeline. The experiments carried out showed the effectiveness of the method with real cases. Future developments will aim to knock down the computing times by moving more computation on the GPU (specifically, the stereo correlation) and to develop of specific a detail-preserving surface generation algorithm.

Acknowledgments

Thanks to G. Vogiatzis, F. Costanzo, F. Funtò, and F. Tornisello for useful hints and discussions. The pictures of the marble artifact have been taken by F. Tornisello.

REFERENCES

Campbell, N., Vogiatzis, G., Hernández, C. and Cipolla, R., 2008. Using multiple hypotheses to improve depth-maps for multi-view stereo. In: European Conference on Computer Vision.

Diebel, J., Thrun, S. and Brünig, M., 2006. A bayesian method for probable surface reconstruction and decimation. ACM Transactions on Graphics 25(1), pp. 39–59.

Drouin, M., Trudeau, M. and Roy, S., 2005. Geo-consistency for wide multi-camera stereo. In: IEEE Conference on Computer Vision and Pattern Recognition, Vol. 1, pp. 351–358.

Faugeras, O. and Keriven, R., 2002. Variational principles, surface evolution, pde's, level set methods and the stereo problem. In: IEEE International Summer School on Biomedical Imaging.

Fromherz, T. and Bichsel, M., 1995. Shape from multiple cues: Integrating local brightness information. In: International Conference for Young Computer Scientists, Vol. 95, pp. 855–862.

Fua, P. and Leclerc, Y., 1995. Object-centered surface reconstruction: Combining multi-image stereo and shading. International Journal of Computer Vision 16(1), pp. 35–56.

Furukawa, Y., 2008. High-fidelity image-based modeling. PhD thesis, University of Illinois at Urbana-Champaign.

Furukawa, Y., Curless, B., Seitz, S. and Szeliski, R., 2009. Reconstructing building interiors from images. In: IEEE International Conference on Computer Vision, IEEE, pp. 80–87.

Furukawa, Y., Curless, B., Seitz, S. and Szeliski, R., 2010. Towards internet-scale multi-view stereo. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1434–1441.

Fusiello, A., Trucco, E. and Verri, A., 2000. A compact algorithm for rectification of stereo pairs. Machine Vision and Applications 12(1), pp. 16–22.

Ganan, S. and McClure, D., 1985. Bayesian image analysis: an application to single photon emission tomography. In: American Statistical Association, pp. 12–18.

Gargallo, P. and Sturm, P., 2005. Bayesian 3d modeling from images using multiple depth maps. In: IEEE Conference on Computer Vision and Pattern Recognition, Vol. 2, pp. 885–891.

Gherardi, R., Toldo, R., Garro, V. and Fusiello, A., 2011. Automatic camera orientation and structure recovery with samantha. In: 3D Virtual Reconstruction and Visualization of Complex Architectures, pp. 38–5.

Hernández Esteban, C. and Schmitt, F., 2004. Silhouette and stereo fusion for 3d object modeling. Computer Vision and Image Understanding 96(3), pp. 367–392.

[1] http://cvlab.epfl.ch/˜strecha/multiview/denseMVS.html

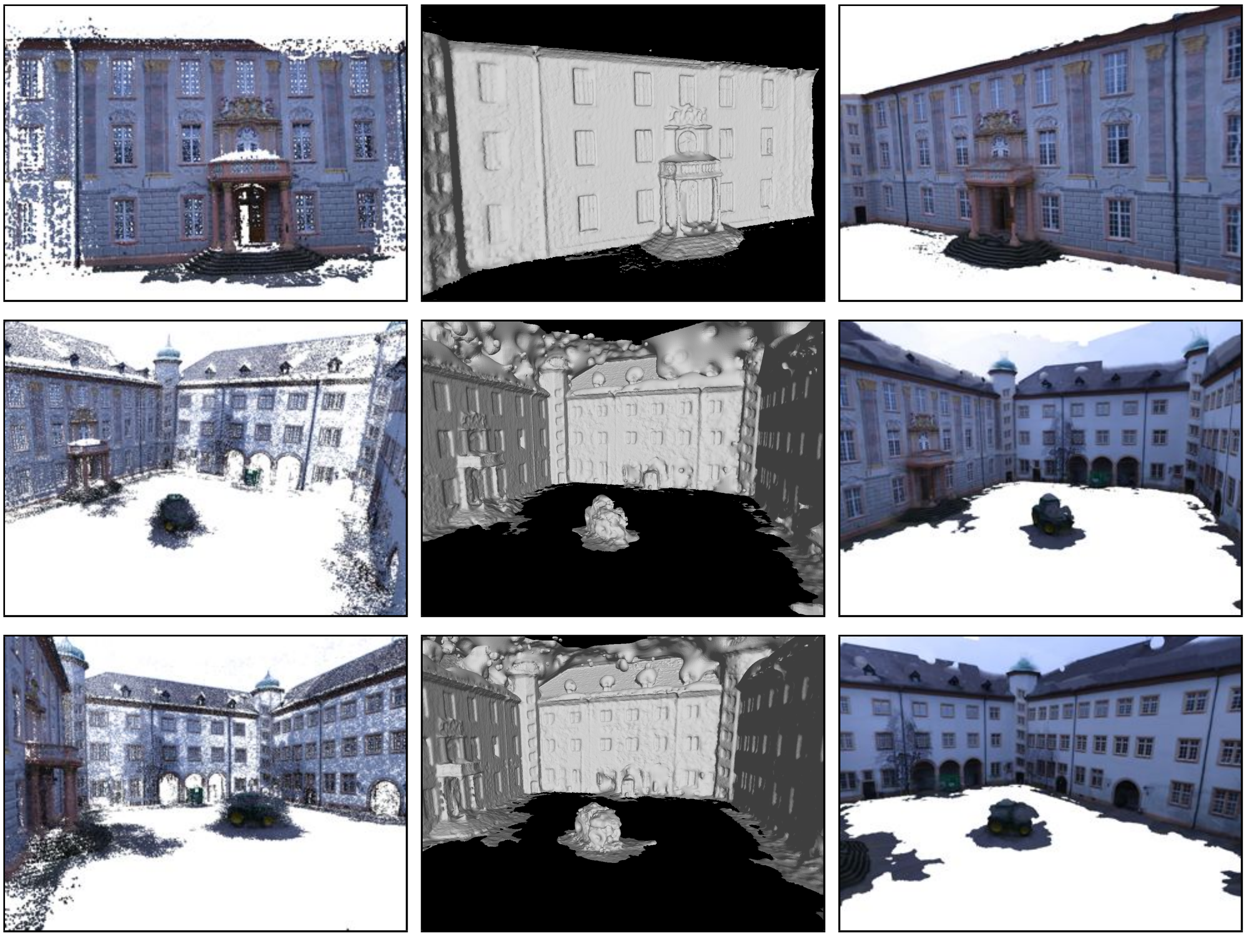


Figure 7: Multiview Stereo Results. From left to right, Stereo Points, Shaded Surfaces and Textured Surfaces are reported. The datasets are, from top to bottom, Entry-P10, Castle-P19, and Castle-P30.

Hiep, V., Keriven, R., Labatut, P. and Pons, J., 2009. Towards high-resolution large-scale multi-view stereo. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1430–1437.

Jin, H., Soatto, S. and Yezzi, A., 2003. Multi-view stereo beyond lambert. In: IEEE Conference on Computer Vision and Pattern Recognition, Vol. 1, pp. I–171.

Kang, S., Szeliski, R. and Chai, J., 2001. Handling occlusions in dense multi-view stereo. In: IEEE Conference on Computer Vision and Pattern Recognition, Vol. 1, pp. I–103.

Kazhdan, M., Bolitho, M. and Hoppe, H., 2006. Poisson surface reconstruction. In: Eurographics symposium on Geometry processing, pp. 61–70.

Kolmogorov, V., 2006. Convergent tree-reweighted message passing for energy minimization. IEEE Transactions on Pattern Analysis and Machine Intelligence 28(10), pp. 1568–1583.

Kolmogorov, V. and Zabih, R., 2002. Multi-camera scene reconstruction via graph cuts. In: European Conference on Computer Vision, pp. 8–40.

Kutulakos, K., 2000. Approximate n-view stereo. In: European Conference on Computer Vision, pp. 67–83.

Kutulakos, K. and Seitz, S., 2000. A theory of shape by space carving. International Journal of Computer Vision 38(3), pp. 199–218.

Manessis, A., Hilton, A., Palmer, P., McLauchlan, P. and Shen, X., 2000. Reconstruction of scene models from sparse 3d structure. In: IEEE Conference on Computer Vision and Pattern Recognition, Vol. 2, pp. 666–671.

Narayanan, P., Rander, P. and Kanade, T., 1998. Constructing virtual worlds using dense stereo. In: IEEE International Conference on Computer Vision, pp. 3–10.

Pons, J., Keriven, R. and Faugeras, O., 2005. Modelling dynamic scenes by registering multi-view image sequences. In: IEEE Conference on Computer Vision and Pattern Recognition, Vol. 2, pp. 822–827.

Roy, S. and Cox, I., 1998. A maximum-flow formulation of the n-camera stereo correspondence problem. In: IEEE International Conference on Computer Vision, pp. 492–499.

Saito, H. and Kanade, T., 1999. Shape reconstruction in projective grid space from large number of images. In: IEEE Conference on Computer Vision and Pattern Recognition, Vol. 2.

Savarese, S., Rushmeier, H., Bernardini, F. and Perona, P., 2001. Shadow carving. In: IEEE International Conference on Computer Vision, Vol. 1, pp. 190–197.

Seitz, S. and Dyer, C., 1999. Photorealistic scene reconstruction by voxel coloring. International Journal of Computer Vision 35(2), pp. 151–173.

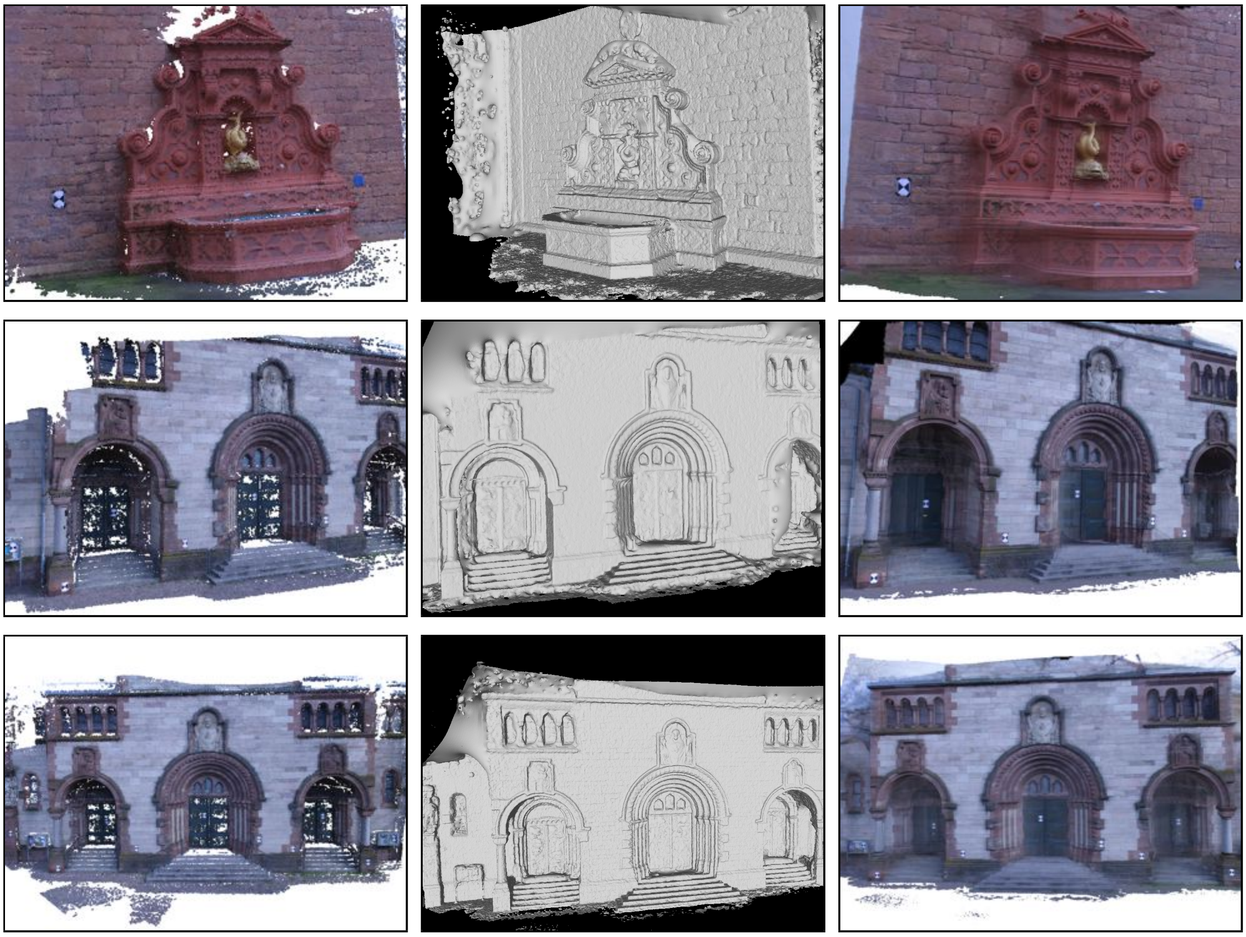


Figure 8: Multiview Stereo Results. From left to right, Stereo Points, Shaded Surfaces and Textured Surfaces are reported. The datasets are, from top to bottom, Fountain-P11, Herz-Jesu-P8, and Herz-Jesu-P25.

Seitz, S., Curless, B., Diebel, J., Scharstein, D. and Szeliski, R., 2006. A comparison and evaluation of multi-view stereo reconstruction algorithms. In: IEEE Conference on Computer Vision and Pattern Recognition, Vol. 1, pp. 519–528.

Sinha, S. and Pollefeys, M., 2005. Multi-view reconstruction using photo-consistency and exact silhouette constraints: A maximum-flow formulation. In: IEEE Internation Conference on Computer Vision, Vol. 1, pp. 349–356.

Slabaugh, G., Culbertson, W., Malzbender, T., Stevens, M. and Schafer, R., 2004. Methods for volumetric reconstruction of visual scenes. International Journal of Computer Vision 57(3), pp. 179–199.

Strecha, C., Von Hansen, W., Van Gool, L., Fua, P. and Thoennessen, U., 2008. On benchmarking camera calibration and multi-view stereo for high resolution imagery. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8.

Szeliski, R., 1999a. A multi-view approach to motion and stereo. In: IEEE Conference on Computer Vision and Pattern Recognition, Vol. 1.

Szeliski, R., 1999b. Prediction error as a quality metric for motion and stereo. In: IEEE International Conference on Computer Vision, Vol. 2, pp. 781–788.

Tasdizen, T. and Whitaker, R., 2004. Higher-order nonlinear priors for surface reconstruction. Pattern Analysis and Machine Intelligence, IEEE Transactions on 26(7), pp. 878–891.

Taylor, C., 2003. Surface reconstruction from feature based stereo. In: IEEE International Conference on Computer Vision, pp. 184–190.

Treuille, A., Hertzmann, A. and Seitz, S., 2004. Example-based stereo with general brdfs. In: European Conference on Computer Vision, pp. 457–469.

Vogiatzis, G., Torr, P. and Cipolla, R., 2005. Multi-view stereo via volumetric graph-cuts. In: IEEE Conference on Computer Vision and Pattern Recognition, Vol. 2, pp. 391–398.

Vu, H., Labatut, P., Pons, J. and Keriven, R., 2012. High accuracy and visibility-consistent dense multi-view stereo. Pattern Analysis and Machine Intelligence, IEEE Transactions on 34(99), pp. 889–901.