ROBUST RECONSTRUCTION OF 3D POINTS FROM IMAGES

Rui Rodrigues[*], *António Ramires Fernandes*

Departamento de Informática
Universidade do Minho
Portugal

ABSTRACT

This paper presents a robust approach for 3D point reconstruction based on a set of images taken from a static scene with known, but not necessarily exact or regular, camera parameters. The points to be reconstructed are chosen from the contours of images, and a *world-based formulation* of the reconstruction problem and associated epipolar geometry is used. The result is a powerful mean of transparently integrating contributions from multiple images, and increased robustness to situations such as occlusions or apparent contours. Two steps for adding robustness are proposed: *cross-checking*, which validates a reconstructed point taken from an image by projecting it on a special subset of the remaining images; and *merging*, which fuses pairs of reconstructed points that are close in 3D space and that were initially chosen from different images. Results obtained with a synthetic scene (for ground truth comparison and error assessment), and two real scenes show the improved robustness achieved with the steps proposed.

1. INTRODUCTION

Numerous techniques exist nowadays to extract 3D information from a set of images of a static scene, recorded with a moving video camera or with a set of static photographic cameras – *multi-viewpoint reconstruction techniques*.

The literature in these areas is extensive, as can be seen in [1, 2]. There are good theoretical tools and results, typically targeted to a subset of potential scenarios. However, *robust* extraction of 3D information from 2D images is still an open problem [3].

Most of those techniques rely on establishing *correspondences* between (or *matching*) image entities, and using *camera calibration* information to convert those correspondences into 3D or depth information. Common entities used for finding correspondences are points [4, 5, 6], lines or contours [7, 8], rectangular blocks or segments [9, 10]. Each of these entity types has some type of ambiguity associated such that given two entities in different images it is not trivial to determine whether they correspond or not [11].

This paper presents a method based on a reinterpretation of the underlying theory in [6]. The work in [6] provides good results in synthetic scenes but it is unable to cope with real scenes and non-exact camera calibration. We have extended [6] in order to increase robustness. We consider a method to be more robust than other if it is able to detect and reject more faulty 3D points, while still providing a good coverage of the scene. To achieve this, two steps were added to the original algorithm: *cross-checking* and *merging*.

Cross-checking validates a reconstructed point taken from an image by projecting it on a special subset of the remaining images, guaranteeing that at least in those images it is matched with a contour. The merging process fuses pairs of matching reconstructed points from different images.

These extensions allow the method to provide more robust results in scenes with noise, aliasing, faulty contour detection, highlights and approximate camera calibration.

Section 2 is devoted to the proposed re-formulation of the method of [6]. Sections 3 and 4 present the robustness steps. Section 5 shows the improvements introduced by those steps in terms of error reduction, and the applicability of the method to real scenes. Finally, conclusions and directions for future work will be laid out in Section 6.

2. 3D POINT RECONSTRUCTION BASED ON CONTOURS AND DISTANCE TRANSFORMS

The method is aimed at robustly reconstructing world (3D) points that project to contour points on images. It is assumed that camera calibration is known or estimated *a priori*, and that the scene is static *ie*, there are no moving objects.

The choice of contour points as the basic entities to be reconstructed is mainly based in two facts. First, the use of points allow us to cope with wide baseline camera settings, by not suffering from the perspective projection distortions that other entities (such as blocks or segments) do. Wide baseline settings have the advantage of better conditioning

[*] The author wishes to thank the support of FCT (Portuguese Foundation for Science and Technology) and Philips Research (Eindhoven)

the reconstruction problem [5].

The process of estimating the world point (*w-point*) $\mathbf{w}_p$ associated with contour point (*c-point*) $\mathbf{c}_{ip}$ in image $\mathbf{I}_i$ is described in the following algorithm:

- Let $\mathbf{V}_{\mathbf{c}_{ip}}$ be the view ray of $\mathbf{c}_{ip}$ associated to $\mathbf{I}_i$
- Choose a set of candidate world points $\mathbf{v}_{ip_1}$, $\mathbf{v}_{ip_2}$, ... from $\mathbf{V}_{\mathbf{c}_{ip}}$
- For each candidate world point $\mathbf{v}_{ip_x}$
    - For each image $\mathbf{I}_j$ $(j \neq i)$
        - Project $\mathbf{v}_{ip_x}$ in $\mathbf{I}_j$ as $\mathbf{p}_{\mathbf{v}_{ip_x},j}$
        - Let $d_{\mathbf{v}_{ip_x},j}$ be the distance of $\mathbf{p}_{\mathbf{v}_{ip_x},j}$ to the closest contour in $\mathbf{I}_j$
        - Add $d_{\mathbf{v}_{ip_x},j}$ to the error $\varepsilon_{\mathbf{v}_{ip_x}}$ of $\mathbf{v}_{ip_x}$
- The reconstructed point (*r-point*) $\mathbf{r}_{ip}$ corresponding to $\mathbf{w}_p$ is selected amongst the candidates with low error value

The distance function is useful to cope with missing or shifted contours, which can be originated by aliasing, noise or faults in the contour detection or errors in camera calibration. In terms of implementation, this distance function is pre-computed by applying a *distance transform* [12] to the contours of each image, generating a *distance map*. At reconstruction time, the overhead to compute the distance is reduced to reading a single value from the distance map.

The choice of candidates from the viewray is similar to choosing candidates in the various epipolar lines. More details can be found in [13]. However, this world-space formulation has a major advantage: *the correspondence, reconstruction and integration stages are merged into one*. When the best candidate is chosen, it is already a *reconstructed point* (*r-point*) and that choice already takes into account the contribution of multiple images. This is a reinterpretation of the depth candidate space proposed in [6].

3. MINIMUM SEARCH AND CROSS CHECKING

The step of choosing the best candidate is not trivial. Several issues relating the analysis of errors as a function of depth and the profiles of these error functions are detailed in [6], and hold in the case of 3D candidates as proposed in Section 2. The error function may have several local minima. Moreover, the correct depth may not be part of the function domain, given the discrete nature of the candidate set. Furthermore, the global minimum of the function may not correspond to the correct depth, due to noise, aliasing and errors in camera calibration. Selecting the candidate closest to the camera associated with an error function minimum below a threshold, is a simple solution for candidate selection. However experiments show that a large percentage of incorrectly reconstructed points is still accepted as valid. There is a need for a robust solution that filters these erroneous points and only keeps the best ones.

Cross-checking allows us to evaluate the validity of a candidate point from a reference camera. If the point is correctly estimated then it should project on a contour on all the cameras *similar* to the reference camera. Camera similarity is judged by their view frustum properties.

If the projection of the candidate is below a pre-defined pixel distance from a contour in all the cameras similar to the reference camera, then the candidate is accepted. Otherwise the candidate it is rejected.

Cross-checking can be integrated with minimum search as a filter to reject minima. Under this approach the closest candidate to the camera with a minimum that passes the cross-check will be selected. Only points for which no minimum passes the cross-check are rejected. A faster option is to test only the closest minimum, computing an r-point if the candidate passes the cross-check. If the candidate does not pass the cross-check it is discarded and no estimate is stored for the associated c-point. Hence, with integrated cross-checking a larger number of valid points is obtained.

4. MERGING MULTIPLE VIEWPOINT RECONSTRUCTIONS

The method described so far can be used to estimate a set of r-points for each image. We now present a simple approach to merge the multiple viewpoint reconstructions. The advantages of merging are twofold: (1) to register and reduce the resulting data sets; and (2) to further increase robustness.

Although registration algorithms for 3D points exist for some time now [14], our goal was to implement and experiment with an algorithm oriented to the specificities of our problem.

The merging process first searches for pairs of *close* r-points coming from *distinct* reference images. Consider a found pair of close r-points ($\mathbf{r}_{ip}$,$\mathbf{r}_{jq}$), associated respectively to the reference cameras $\mathbf{C}_i$ and $\mathbf{C}_j$ and the corresponding c-points $\mathbf{c}_{ip}$, $\mathbf{c}_{jq}$.

In order to establish a correspondence between $\mathbf{r}_{ip}$ and $\mathbf{r}_{jq}$, their projections must match, *ie*, the projection of $\mathbf{r}_{ip}$ in the image plane from $\mathbf{C}_j$ must match $\mathbf{c}_{jq}$, and *vice versa*. Matching at this stage occurs on the image plane, and with a small tolerance.

Merging is incremental, *ie* if the pairs ($\mathbf{r}_{ip}$,$\mathbf{r}_{jq}$) and ($\mathbf{r}_{jq}$,$\mathbf{r}_{kr}$) are merged, then ($\mathbf{r}_{ip}$,$\mathbf{r}_{jq}$,$\mathbf{r}_{kr}$) are merged together. The resulting 3D merged point (*m-point*) is the average of the r-points that were merged together. For each merged point the set of r-points, and its associated information is kept.

The larger the number of independently estimated r-points that are associated with a single m-point, the larger the supporting evidence that the m-point is indeed correctly estimated. For this reason, if a given r-point is not merged into any m-point, it is discarded from the final reconstruction.

Hence, merging provides yet another layer of validation of the estimates for each frame, thereby adding robustness to the overall method.

5. EXPERIMENTAL RESULTS

We now present a series of results obtained by applying our technique to a synthetic scene and two real scenes. The synthetic scene aims to show the technique's accuracy, due to available ground truth data. The real scenes show the applicability of the method in less-than-ideal situations. Table 1 shows properties of the experiments. Tests were performed on a Pentium IV at 2 GHz.

Scene	Bunny	Speaker	Boat
Type	Synthetic	Real	Real
Nr. of images	69	30	39
Resolution	512x512	1024x768	1024x768
Cam. calibration	Exact	Fair	Fair
Recon Time	< 45 sec	< 40 sec	< 130 sec

Table 1. Test Properties

5.1. Synthetic Scene - ”Bunny”

The selected synthetic model is a textured version of the famous bunny from the Stanford 3D Scanning Repository (Figure 1(a)). This model presents some challenges to reconstruction methods: due to the curvature of the surface, apparent contours are abundant; furthermore, the model is self-occluding in several places. Only contour junction points were estimated in this case (Figure 1(b)).

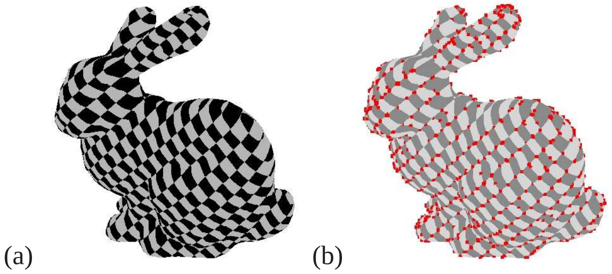

Fig. 1. (a) Bunny; (b) Bunny with m-points superimposed

The bunny is approximately 12 units wide by 12 tall (including the ears). The results show that, when using cross-checking with zero-pixel distance tolerance, and considering all images, 88.20% of the r-points have an error below 0.1 units. Without cross-checking this value drops off sharply to 57.80%.

After merging the results obtained with cross-checking, the percentage of the obtained m-points with error below 0.1 units rises considerably to 96.66%. Aplying merging to the results without cross-checking also increases robustness, granting 80.39% of m-points with error below 0.1 units.

Assuming for instance that a unit is equivalent to one centimetre, then our method, with cross-checking and merging, yields 96.66% of m-points that are at most one milimetre away from the original bunny's surface. Over 99% of the m-points have an error below 2 milimetres.

5.2. Real Scene - ”Speaker”

This scene consists of a small loudspeaker (Figure 2(a)). The speaker shows some highlights in the metallic parts such as the stand and the two round membranes. Experiments show that the method was able to cope with the highlights. Some self occlusions are also present in the images. Figure 2(b) shows the reconstructed points from three new views.

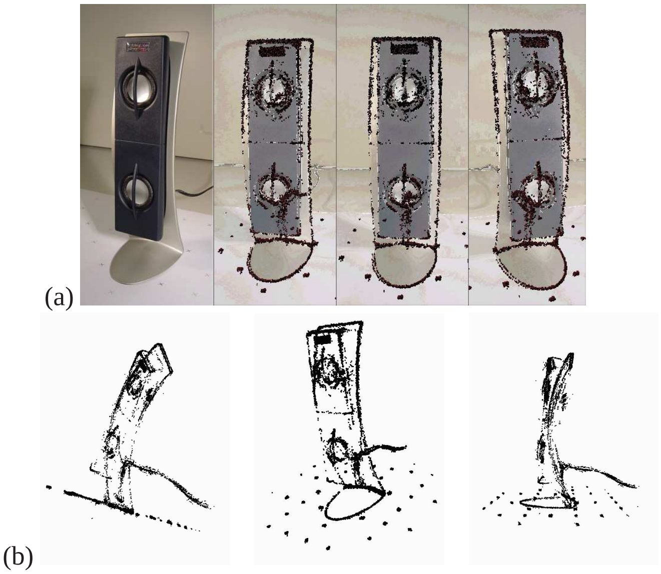

Fig. 2. ”Speaker”: (a) An image of the sequence, and the reconstructed points superimposed on some of the images; (b) Three new views of the reconstructed points

5.3. Real Scene - ”Boat”

The last scene contains a small wooden boat (Figure 3(a)). This scene is particularly challenging, due to the large amount of occlusions between the ropes, the masts and the cabinets on the deck, and also to the less-than-perfect contour extraction. Figure 3(b) shows the m-points obtained superimposed in one of the original images. Figures 3(c) and 3(d) show the m-points from two new viewpoints not covered by the original cameras.

6. CONCLUSIONS AND FUTURE WORK

The method proposed in this article was based in [6]. We have reinterpreted and greatly enhanced the previous work, allowing us to apply the method to complex real scenes. The method is able to cope with noise, aliasing, faulty contour detection, highlights, and approximate camera calibrations. Hence, it could be applied to real scenes, as shown in our experiments, with very encouraging results. In the future we will focus on extending the algorithm even further to include line reconstruction and model generation. We would also like to explore the possibility of feeding back reconstruction information to try to improve camera calibration.

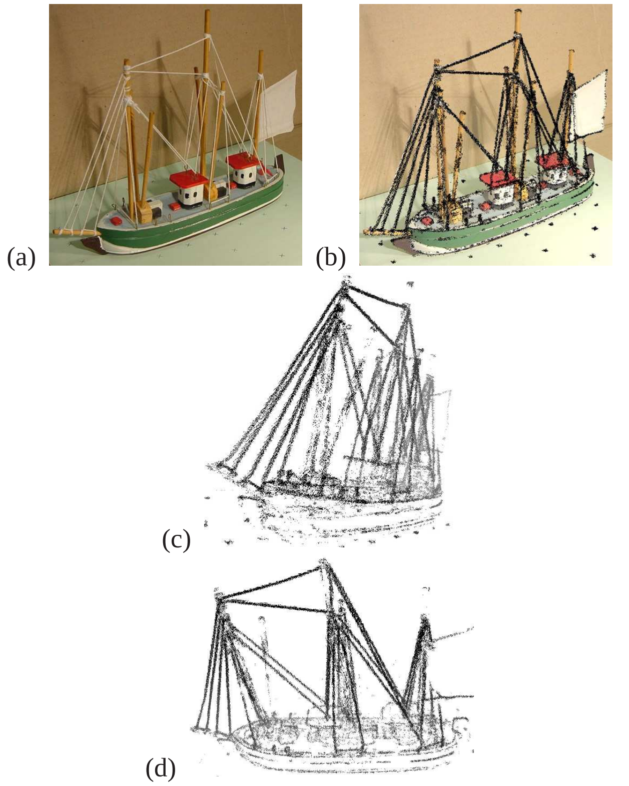

Fig. 3. (a) The boat; (b) the boat with m-points superimposed; (c) m-points from an arbitrary view; (d) Another arbitrary view of the m-points (darker points are closer to the camera)

7. REFERENCES

[1] Olivier Faugeras, *Three-Dimensional Computer Vision: A Geometric Viewpoint*, MIT Press, Cambridge, Massachusetts, 1993.

[2] Richard Hartley and Andrew Zisserman, *Multiple View Geometry in computer vision*, Press Syndicate of the University of Cambridge, 2000.

[3] Ruigang Yang, Greg Welch, and Gary Bishop, "Real-time consensus-based scene reconstruction using commodity graphics hardware," in *Proceedings of Pacific Graphics 2002*, Tsinghua University, Beijing, China, 2002.

[4] K. Kawamoto and A. Imiya, "Detection of spatial points and lines by random sampling and voting procedure," *Pattern Recognition Letters*, vol. 22, no. 2, pp. 199–207, February 2001.

[5] Dennis Tell and Stefan Carlsson, "Wide baseline point matching using affine invariants computed from intensity profiles," in *ECCV (1)*, 2000, pp. 814–828.

[6] Rui Rodrigues, António Ramires Fernandes, Kees van Overveld, and Fabian Ernst, "Reconstructing depth from spatiotemporal curves," in *Proceedings of the 15th International Conference on Vision Interface*, Calgary, Canada, 2002, pp. 252–259.

[7] J. Sato and R. Cipolla, "Affine reconstruction of curved surfaces from uncalibrated views of apparent contours," *IEEE Trans. Pattern Analysis and Machine Intell.*, vol. 21, no. 11, pp. 1188–1198, November 1999.

[8] Hanns Schulz-Mirbach and I. Weiss, "Projective reconstruction from curve correspondences in uncalibrated views," Technical Report TR-402-94-014, Technical University of Hamburg-Harburg, 1994.

[9] Stephen M. Smith, "ASSET-2: Real-time motion segmentation and object tracking.," *Real-Time Imaging*, vol. 4, no. 1, pp. 21–40, 1998.

[10] Reinhard Koch, Marc Pollefeys, and Luc Van Gool, "Realistic 3-d scene modeling from uncalibrated image sequences," *Journal of Visualization and Computer Animation*, vol. 11, pp. 115–127, 2000.

[11] A. Redert, E. Hendriks, and J. Biemond, "Correspondence estimation in image pairs," *IEEE Signal Processing Magazine*, vol. 16, no. 3, pp. 29–46, 1999.

[12] Gunilla Borgefors, "Distance transformations in digital images," *Computer Vision, Graphics, and Image Processing*, vol. 34, no. 3, pp. 344–371, 1986.

[13] Rui Rodrigues and António Ramires Fernandes, "Accelerated epipolar geometry computation for 3d reconstruction using projective texturing," in *Proceedings of SCCG'04*. 2004, to be published by ACM Press.

[14] Andrew W. Fitzgibbon, "Robust registration of 2d and 3d point sets," in *British Machine Vision Conference*, 2001.