

# Markov Random Field-Based Clustering for the Integration of Multi-view Range Images

Ran Song<sup>1</sup>, Yonghuai Liu<sup>1</sup>, Ralph R. Martin<sup>2</sup>, and Paul L. Rosin<sup>2</sup>

<sup>1</sup> Department of Computer Science, Aberystwyth University, UK  
`{res,yyl}@aber.ac.uk`

<sup>2</sup> School of Computer Science & Informatics, Cardiff University, UK  
`{Ralph.Martin,Paul.Rosin}@cs.cardiff.ac.uk`

**Abstract.** Multi-view range image integration aims at producing a single reasonable 3D point cloud. The point cloud is likely to be inconsistent with the measurements topologically and geometrically due to registration errors and scanning noise. This paper proposes a novel integration method cast in the framework of Markov random fields (MRF). We define a probabilistic description of a MRF model designed to represent not only the interpoint Euclidean distances but also the surface topology and neighbourhood consistency intrinsically embedded in a predefined neighbourhood. Subject to this model, points are clustered in an iterative manner, which compensates the errors caused by poor registration and scanning noise. The integration is thus robust and experiments show the superiority of our MRF-based approach over existing methods.

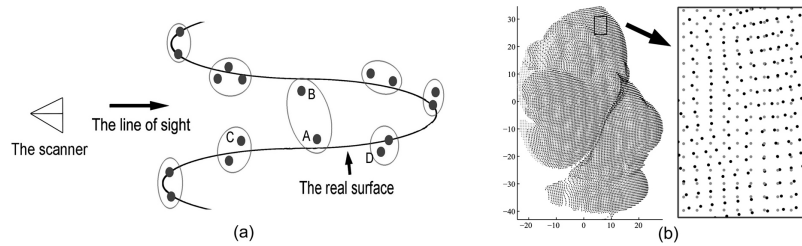
## 1 Introduction

3D surface reconstruction from multi-view 2.5D range images is important for a wide range of applications, such as reverse engineering, CAD and quality assurance, etc. Its goal is to estimate a manifold surface that approximates an unknown object surface using multi-view range images, each of which essentially represents a sample of points in 3D Euclidean space. These samples of points are usually described in local, system centred, coordinate systems and cannot offer a full coverage of the object surface. To reconstruct a complete 3D surface model, we need to register a set of overlapped range images into a common coordinate frame and then integrate them to fuse the redundant data contained in overlapping regions while retain enough data sufficiently representing the correct surface details. However, to achieve both is challenging due to its ad hoc nature. Scanning noise such as unwanted outliers and data loss typically caused by self-occlusion, large registration errors and connectivity relationship loss among sampled points in acquired data often lead to a poor integration. As a result, the reconstructed surface may include holes, false connections, thick and non-smooth or over-smooth patches, and artefacts. Hence, a good integration should be robust to inevitable registration errors and noise. Once multiple registered overlapping range images have been fused into a single reasonable point cloud, many techniques [1–3] can be employed to reconstruct a watertight surface.

## 2 Related Work

Existing integration methods can be classified into four categories: volumetric method, mesh-based method, point-based method and clustering-based method. The volumetric method [4–7] first divides the space around objects into voxels and then fuses the data in each voxel. But the comparative studies [8, 9] show they are time-consuming, memory-hungry and not robust to registration errors and scanning noise, resulting in poor reconstructed surfaces. The mesh-based method [10–13] first employs a step discontinuity constrained triangulation and then detects the overlapping regions between the triangular meshes derived from successive range images. Finally, it reserves the most accurate triangles in the overlapping regions and reconnects all remaining triangles subject to a certain objective function. Since the number of triangles is usually much larger than that of the sampled points, the mesh-based methods are computationally more expensive. Some mesh-based methods employ a 2D triangulation in the image plane to estimate the local surface connectivity as computation in a 2D subspace is more efficient. But projection from 3D to 2D may lead to ambiguities if the projection is not injective. The mesh-based methods are thus highly likely to fail in non-flat areas where no unique projection plane exists. The point-based method [14, 15] produces a set of new points with optimised locations. But its integration result is often over-smooth and cannot retain enough surface details in non-flat areas. The clustering-based method [8, 9] employs classical clustering methods to minimise dissimilarity objective functions. It surpasses previous methods as it is more robust to noise and registration errors. Also, the iterative clustering optimises the locations of points and thus generates much fewer ill-shaped triangles. However, this clustering, only based on Euclidean distance, does not consider local surface topology and neighbourhood consistency, leading to errors in non-flat areas. For instance, in Fig.1(a), although point *A* is closer to *B* and thus the clustering-based methods wrongly group them together, we would rather group *A* with *C* or *D* to maintain the correct surface topology.

In this paper, we propose a novel integration method. A MRF model is designed based on both statistical and structural information that clustering-based methods neglect. This model is then converted into a specific description



**Fig. 1.** (a) Local topology has a significant effect on the point clustering in non-flat areas (b) X-Y projection of point clouds from reference and registered range images where the gray points are the raw data and the black points are the registered ones

minimised in a clustering manner. The new method retains the advantages of clustering-based methods and is more robust as it uses more information from the input data. The integration is thus reliable in both flat areas and non-flat areas. It is worth mentioning that our method does not require a regular image grid whereas some state-of-the-art techniques [16, 17] rely heavily on it. The registered range images used as the input in this work are actually 3D unstructured point clouds due to large registration errors (Fig.1(b)). Generally, our method can also cope with the more general input-multiple 3D unstructured point clouds.

### 3 Markov Random Field-Based Clustering

MRF describes a system by local interaction and is able to capture many features of the system of interest by simply adding appropriate terms representing spatial or contextual dependencies into it. For this work, we denote a set of sites  $s = \{1, \dots, n\}$  representing the primitive points and define the label assignment  $x = \{x_1, \dots, x_n\}$  to all sites as a realisation of a family of random variables defined on  $s$ . We also define the feature space that describes the sites and understand it as a random observation field with its realisation  $y = \{y_1, \dots, y_n\}$ . We also denote a label set  $L = \{1, \dots, m\}$ , where each label corresponds to a class centroid.

An optimal labeling should minimise the posterior energy  $U(x|y) = U(y|x) + U(x)$ . Under the MRF assumption that the observations are mutually independent, the likelihood energy  $U(y|x)$  can be computed as  $U(y|x) = \sum_{i \in S} V(y_i|x_i)$ . In this work,  $V(y_i|x_i)$  is the Euclidean distance between point and class centroid:

$$V(y_i|x_i) = \|y_i - C_{x_i}\| = |y_i - C_{x_i}| \quad (1)$$

It can be understood in this way: given a label set  $C$  and an observation field  $y$  which has an observation  $y_i$  at point  $i$ , whether  $i$  should obey an assignment  $x_i$  which assigns it to a centroid  $C_{x_i}$  ( $C_{x_i} \in C$ ) depends on the distance between them. The smaller the distance, the larger the probability for such an assignment.

Once we define a neighbourhood  $N(i)$  for point  $i$ , the prior energy  $U(x)$  can be expressed as the sum of different types of clique energies:

$$U(x) = \sum_i V_1(x_i) + \sum_i \sum_{i' \in N(i)} V_2(x_i, x_{i'}) + \sum_i \sum_{i' \in N(i)} \sum_{i'' \in N(i)} V_3(x_i, x_{i'}, x_{i''}) + \dots \quad (2)$$

Here, single-point cliques  $V_1(x_i)$  are set to 0 as we have no preference which label should be better. Our MRF model is inhomogenous and anisotropic due to the specific definition of the neighbourhood. Furthermore, a range image does not have such a discrete property attached to each point as intensity. Considering the inhomogeneous sites with continuous labels, we cannot use a simplified form such as the Ising model [18] to ‘discourage’ the routine from assigning different labels to two neighbouring points. The difference between the normals is used to evaluate the binary clique energy representing the neighbourhood consistency:

$$U(x) = \sum_i \sum_{i' \in N(i)} V_2(x_i, x_{i'}) = \sum_i \sum_{i' \in N(i)} w |\mathbf{A}_{C_{x_i}} - \mathbf{A}_{C_{x_{i'}}}| \quad (3)$$

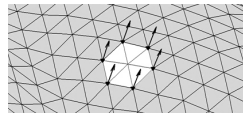
where  $\mathbf{A}_{C_{x_i}}$  and  $\mathbf{A}_{C_{x_{i'}}}$  are the unit normal vectors of the centroids  $C_{x_i}$  and  $C_{x_{i'}}$  respectively. The neighbourhood consistency constraint is thus based on the assumption that point normals should not deviate from each other too much in a small neighbourhood. The minimisation combining two types of energies arises from different value ranges. The range of the likelihood energy is usually dependent on the number and the positions of class centroids, whereas the range of the clique energy depends on the definition and the size of the neighbourhood system and the measurement of the difference between the normals. One weighting parameter  $w$  is thus necessary to balance the two kinds of energies.

We define a normal deviation parameter  $s$  related to local surface topology:

$$s_i = std(a_{i'} | i' \in N(i)), \quad a_{i'} = \cos \theta_{i'} = (\mathbf{A}_{i'} \cdot \mathbf{z}) / |\mathbf{A}_{i'}| |\mathbf{z}|, \quad i' \in N(i) \quad (4)$$

where  $std$  is the standard deviation function. We choose  $\mathbf{z}$  ( $z$  axis) as the reference direction because the scanning accuracy of a point depends on the including angle between its normal and the line of sight from the scanner which is along  $\mathbf{z}$  [13].

For most MRF applications in computer vision [19–22], 4 neighbouring pixels are chosen to produce a neighbourhood  $N(i)$  for a pixel  $i$ . But to define  $N(i)$  in a 3D unstructured point cloud is more difficult since the concept ‘pixel’ does not exist here. The simplest method is to use the  $k$ -nearest-neighbours algorithm ( $k$ -NN) to find the  $k$  points closest to  $i$ . But this method has its drawbacks. First, to employ  $k$ -NN needs extra computational cost. Second, more importantly, the neighbours produced by  $k$ -NN cannot deliver surface topological information. Due to incorrect registration and noise, some of the neighbours produced by  $k$ -NN may not be on the surface of point cloud. The following calculation of their normals will thus be inaccurate and makes the whole algorithm unreliable. In this work, before the integration, we first do a triangulation for the points in the overlapping regions and find the neighbouring triangles of point  $i$ . The vertices of these triangles, excluding  $i$ , are defined as the neighbours of  $i$ . The collection of all neighbours of  $i$  is defined as the neighbourhood of  $i$ , written as  $N(i)$ . Fig.2 shows a neighbourhood. The advantages of this definition are: (1) it reflects the local topology and makes it possible to evaluate the neighbourhood consistency defined as the difference between the normals in the MRF model; (2) there is no extra computational cost as the triangulation has been done at the beginning of the algorithm. Since points may have different number of neighbours, it is necessary to add a normalisation parameter to measure the clique energy. Also, the clique energy defined in this way is not relevant to scanning resolution, but the likelihood energy, described by the distance measurement is directly related to it. We thus use a constant  $c$  to balance the different magnitudes between the



**Fig. 2.** A neighbourhood and the normalised normal vectors attached to the neighbours

two types of energies. In a general form,  $w = s_i \times c \times R/n_i$ , where  $n_i$  is the number of neighbours of  $i$  and  $R$  is the scanning resolution of the input images.

If point  $i$  lies in a non-flat area such as a crease edge,  $s_i$  will probably be a large value and the clique energy will have more weight. Thus the error illustrated in Fig.1 will probably be avoided. If  $i$  is located in a flat area,  $s_i$  will be quite small or even equal to 0 (in a planar area). In this case, the algorithm is actually downgraded to the classical k-means clustering. This is desired as the k-means clustering works well in flat areas. In other words, whether a point can be assigned to a certain centroid depends upon not only the distance between them but also the local topology and neighbourhood consistency.

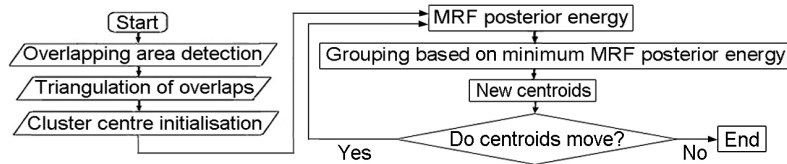
In image segmentation where MRF has already been widely used [19–21, 23], the label set is usually defined using an intensity set and its domain remains unchanged in the algorithm. This is a ‘pure’ MRF labeling problem and can be solved by some well-known methods such as graph cuts and loopy belief propagation [24]. But these methods cannot guarantee a reconstructed mesh with well-shaped triangles. In contrast, the clustering-based methods can achieve that due to the averaging for the calculation of new centroids in the clustering [8, 9]. We thus solve this combinatorial minimisation problem in a clustering manner. In our method, both the label set and its domain are changed iteratively. A double optimisation is thus achieved. One is done by MRF under a given label set and the other one is done by the iteration analogous with the routine of the k-means clustering. The changing label set is the centroids. Once one point  $i$  is assigned to a centroid  $C_{x_i}$ , it is labeled as  $C_{x_i}$  and its normal is also labeled as  $\mathbf{A}_{C_{x_i}}$ . Analogously, the normal of  $i'$  should be labeled using  $\mathbf{A}_{C_{x_{i'}}}$ . But estimating  $\mathbf{A}_{C_{x_{i'}}}$  needs  $n_i$  extra nearest neighbour searches for each point and that would significantly slow down the whole algorithm. We thus assume  $\mathbf{A}_{C_{x_{i'}}} = \mathbf{A}_{i'}$ , so

$$U(x|y) = \sum_i \sum_{i' \in N(i)} \frac{s_i}{n_i} \times c \times R \times |\mathbf{A}_{C_{x_i}} - \mathbf{A}_{i'}| + \sum_i |y_i - C_{x_i}| \quad (5)$$

## 4 Implementation

Fig.3 shows the workflow of the new algorithm. We employ the method proposed in [8] to define correspondences and detect overlapping areas. Then, the points in the overlapping areas are triangulated to compute normals and find neighbours.

The initialisation is vital to the integration result and the speed of convergence. Each point in the overlapping area is shifted along its normal towards



**Fig. 3.** The workflow of the MRF-based clustering algorithm

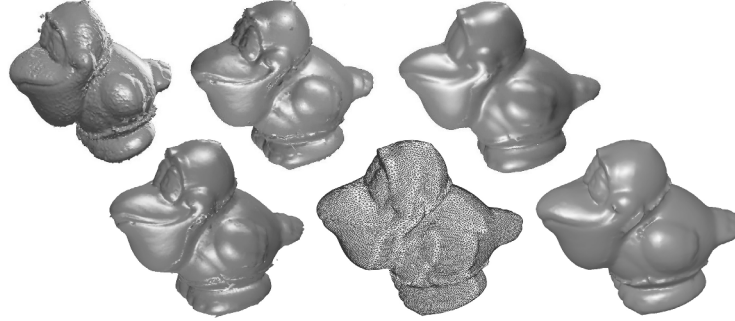
its corresponding point by half of the distance between them. For each point shifted from a point of the reference image, a sphere with a radius  $r = m \times R$  ( $m$  is a parameter controlling the density of the output point cloud) is defined. If some points fall into this sphere, then their original points (without shifting) and normals are retrieved. Their averages are then used to initialise cluster centroids and their normals. The centroid yielding the lowest energy is chosen to label the point. All points labeled with the same centroid are grouped as one class. The centroid of a new class and its normal can be recalculated by computing the mean values of the coordinates of the class members and their normals. The iteration ends when centroids do not change any more. Finally, the new point set consisting of the new centroids and the original points not in the overlapping areas is used to reconstruct a triangular mesh and a watertight surface.

In practice, we use a speeded-up scheme to find the centroid  $C_{x_i}$ . We first perform k-NN to find the  $k$  centroids closest to  $i$ . Then we just search the one minimising  $U(x|y)$  out of the  $k$  centroids.  $k$  is set to 5 in this work to ensure that the one searched can always minimise  $U(x|y)$  among all centroids. Otherwise it means that the balance constant  $c$  is not set to a reasonable value and the clique energy has too much weight. Thus the new method has the same computational complexity as the k-means clustering. The computational complexity of our algorithm is  $O(n_1 \log_2 n_2)$  where  $n_1$  and  $n_2$  are the numbers of points in the overlapping regions from both images and the reference image respectively.

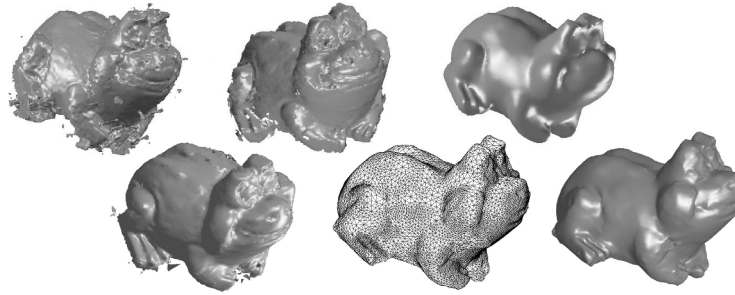
## 5 Experimental Results and Performance Analysis

In our experiments, the input range images are all downloaded from the OSU Range Image Database (<http://sampl.ece.ohio-state.edu/data/3DDB/RID/index.htm>). On average, each ‘Bird’ image has 9022 points and each ‘Frog’ image has 9997 points. We employed the method proposed in [25] for pairwise registration. Inevitably, the images are not accurately registered, since we found the average registration error (0.30mm for the ‘Bird’ images and 0.29mm for the ‘Frog’ images) is as high as half the scanning resolution of the input data. A large registration error causes corresponding points in the overlapping area to move away from each other and the overlapping area detection will thus be more difficult. The final integration is likely to be inaccurate accordingly. Fig.4 and 5 show the comparative integration results produced by existing methods and our method.

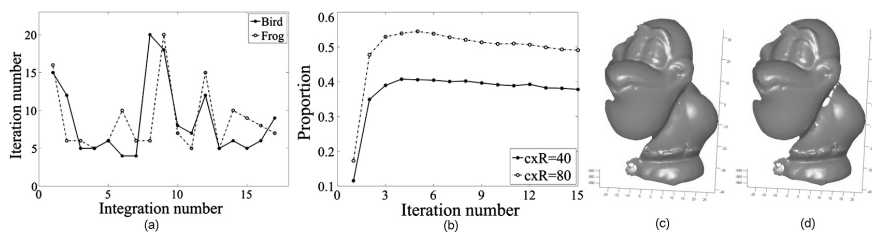
Fig.6(a) illustrates the convergence performance of the new method. The new method achieves a high computational efficiency in terms of iteration number required for convergence. We also compute the proportion of the points really affected by clique energy. Here, ‘really affected’ means one point was not labeled with its closest centroid due to the effect from clique energy. Fig.6(b) shows the statistics over the integration of two Bird images ( $0^\circ$  and  $20^\circ$ ) using different parameters. Clique energy gains more weight when the product of  $c$  and  $R$  is larger. So, Fig.6(b) shows more points are really affected by clique energy when  $c \times R = 80$ . But it does not mean a better integration can be achieved as shown in Fig.6(c) and (d). We can see that it is important to choose an appropriate  $c$ .



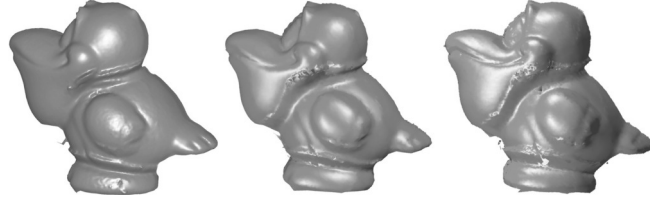
**Fig. 4.** Integration results of 18 ‘Bird’ images. Top left: volumetric method [5]. Top middle: mesh-based method [13]. Top right: fuzzy-c means clustering [9]. Bottom left: k-means clustering [8]. Bottom middle: the triangular mesh produced by the new method. Bottom right: the final integration result produced by the new method.



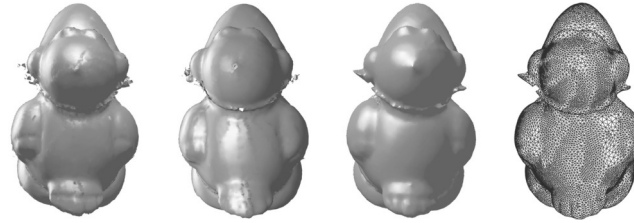
**Fig. 5.** Integration results of 18 ‘Frog’ images. Top left: volumetric method [5]. Top middle: mesh-based method [13]. Top right: fuzzy-c means clustering [9]. Bottom left: k-means clustering [8]. Bottom middle: the triangular mesh produced by the new method. Bottom right: the final integration result produced by the new method.



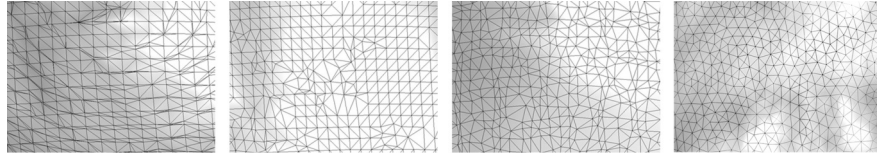
**Fig. 6.** (a): The convergence performance of the new mehtod. (b): The proportion of the points really affected by clique energy (c): Integration result when  $c \times R = 40$  (d): Integration result when  $c \times R = 80$ . Please note the holes around the neck.



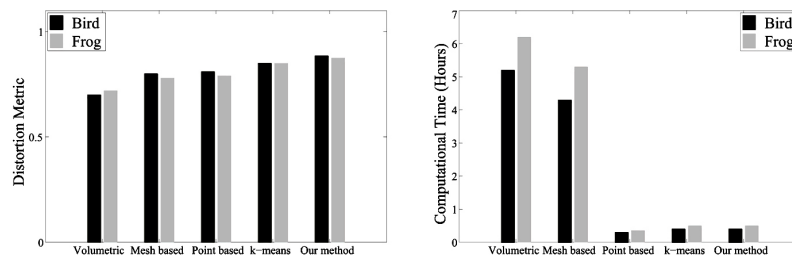
**Fig. 7.** Integration results of 18 ‘Bird’ images produced by different clustering methods. Left: MRF-based clustering; Middle: fuzzy-c means; Right: k-means.



**Fig. 8.** Integration re0.7ts (back view) of 18 ‘Bird’ images produced by different clustering methods. From left to right: k-means clustering, fuzzy-c means clustering, the reconstructed surface and triangular mesh produced by MRF-based clustering.



**Fig. 9.** Triangulated meshes of the integrated surface using different methods. From left to right: volumetric[5], mesh-based method[13], point-based method[15], our method.



**Fig. 10.** Different performance measures of integration algorithms. Left: distortion metric. Right: computational time.



Due to the different objective functions, it is very difficult to define a uniform metric such as the integration error [8, 9] for comparison because we reject the idea that the closest centroid is the best choice. However, Fig.7 and 8 highlight the visual difference of the integration results produced by classical clustering methods and our method. It can be seen that our method performs better, particularly in the non-flat regions such as the neck and the tail of the bird.

Even so, for a fair comparison, we introduce some measurement parameters widely used but not relevant to the objective function: 1) The distribution of interior angles of triangles. The angle distribution shows the global optimal degree of triangles. The closer the interior angles are to  $60^\circ$ , the more similar the triangles are to equilateral ones; 2) Average distortion metric [26]: the distortion metric of a triangle is defined as its area divided by the sum of the squares of the lengths of its edges and then normalised by a factor  $2\sqrt{3}$ . The value of distortion metric is in  $[0,1]$ . The higher the average distortion metric value, the higher the quality of a surface; 3) The computational time; Fig.9 and 10 show that the MRF-based algorithm performs better in the sense of the distribution of interior angles of triangles, the distortion metric and the computational time. All experiments were done on a Pentium IV 2.40 GHz computer. Additionally, because there is no specific segmentation scheme involved in the MRF-based clustering, our algorithm saves computational time compared with the techniques using some segmentation algorithms as a preprocessing before the integration [9].

## 6 Conclusion

Clustering-based methods proved superior to other existing methods for integrating multi-view range images. It has, however, been shown that classical clustering methods lead to significant misclassification in non-flat areas as the local surface topology are neglected. We develop a MRF-based method to tackle this problem and produce better integration results. It does not only focus on minimising the integration errors defined by the Euclidean distance, but also considers the effect from local topology and neighbourhood consistency. The reconstructed surfaces are geometrically realistic since the new method essentially uses more information contained within the input data. Also, it is applicable to more general data sources such as 3D unstructured point clouds.

*Acknowledgments.* Ran Song is supported by HEFCW/WAG on the RIVIC project. This support is gratefully acknowledged.

## References

1. Katulakos, K., Seitz, S.: A theory of shape by space carving. *Int. J. Comput. Vision* 38, 199–218 (2000)
2. Dey, T., Goswami, S.: Tight cocone: a watertight surface reconstructor. *J. Comput. Inf. Sci. Eng.* 13, 302–307 (2003)
3. Yemez, Y., Wetherilt, C.: A volumetric technique for surface reconstruction from silhouette and range data. *Comput. Vision Image Understanding* 105, 30–41 (2007)

4. Curless, B., Levoy, M.: A volumetric method for building complex models from range images. In: Proc. SIGGRAPH, pp. 303–312 (1996)
5. Dorai, C., Wang, G.: Registration and integration of multiple object views for 3d model construction. *IEEE Trans. Pattern Anal. Mach. Intell.* 20, 83–89 (1998)
6. Rusinkiewicz, S., Hall-Holt, O., Levoy, M.: Real-time 3d model acquisition. In: Proc. SIGGRAPH, pp. 438–446 (2002)
7. Sagawa, R., Nishino, K., Ikeuchi, K.: Adaptively merging large-scale range data with reflectance properties. *IEEE Transactions on PAMI* 27, 392–405 (2005)
8. Zhou, H., Liu, Y.: Accurate integration of multi-view range images using k-means clustering. *Pattern Recognition* 41, 152–175 (2008)
9. Zhou, H.: et al: A clustering approach to free form surface reconstruction from multi-view range images. *Image and Vision Computing* 27, 725–747 (2009)
10. Hilton, A.: On reliable surface reconstruction from multiple range images. Technical report (1995)
11. Turk, G., Levoy, M.: Zippered polygon meshes from range images. In: Proc. SIGGRAPH, pp. 311–318 (1994)
12. Rutishauser, M., Stricker, M., Trobina, M.: Merging range images of arbitrarily shaped objects. In: Proc. CVPR, pp. 573–580 (1994)
13. Sun, Y., Paik, J., Koschan, A., Abidi, M.: Surface modeling using multi-view range and color images. *Int. J. Comput. Aided Eng.* 10, 137–150 (2003)
14. Li, X., Wee, W.: Range image fusion for object reconstruction and modelling. In: Proc. First Canadian Conference on Computer and Robot Vision, pp. 306–314 (2004)
15. Zhou, H., Liu, Y.: Incremental point-based integration of registered multiple range images. In: Proc. IECON, pp. 468–473 (2005)
16. Bradley, D., Boubekeur, T., Berlin, T., Heidrich, W.: Accurate multi-view reconstruction using robust binocular stereo and surface meshing. In: CVPR (2008)
17. Christopher, Z., Pock Thomas, B.H.: A globally optimal algorithm for robust tv-l1 range image integration. In: Proc. ICCV (2007)
18. Li, S.: Markov random field modelling in computer vision. Springer, Heidelberg (1995)
19. Wang, X., Wang, H.: Markov random field modelled range image segmentation. *Pattern Recognition Letters* 25, 367–375 (2004)
20. Suliga, M., Deklerck, R., Nyssen, E.: Markov random field-based clustering applied to the segmentation of masses in digital mammograms. *Computerized Medical Imaging and Graphics* 32, 502–512 (2008)
21. Felzenszwalb, P., Huttenlocher, D.: Efficient belief propagation for early vision. *International Journal of Computer Vision* 70, 41–54 (2006)
22. Diebel, J., Thrun, S.: An application of markov random fields to range sensing. In: Proc. NIPS (2005)
23. Jain, A., Nadabar, S.: Mrf model-based segmentation of range images. In: Proc. Third International Conference on Computer Vision, pp. 667–671 (1990)
24. Szeliski, R., et al.: A comparative study of energy minimization methods for markov random fields with smoothness-based priors. *IEEE Transactions on PAMI* 30, 1068–1080 (2008)
25. Liu, Y.: Automatic 3d free form shape matching using the graduated assignment algorithm. *Pattern Recognition* 38, 1615–1631 (2005)
26. Lee, C., Lo, S.: A new scheme for the generation of a graded quadrilateral mesh. *Computers and Structures* 52, 847–857 (1994)