

Improving Shape from Shading with Interactive Tabu Search

Jing Wu, Paul L. Rosin, Xianfang Sun, and Ralph R. Martin

School of Computer Science & Informatics, Cardiff University, Cardiff, CF24 3AA, UK

E-mail: {J.Wu, Paul.Rosin, Xianfang.Sun, Ralph.Martin}@cs.cardiff.ac.uk

Received Nov 27, 2015

Revised Feb 4, 2016

Abstract Optimisation based shape from shading (SFS) is sensitive to initialization: errors in initialization are a significant cause of poor overall shape reconstruction. In this paper, we present a method to help overcome this problem by means of user interaction. There are two key elements in our method. Firstly, we extend SFS to consider a set of initializations, rather than to use a single one. Secondly, we efficiently explore this initialization space using a heuristic search method, tabu search, guided by user evaluation of the reconstruction quality. Reconstruction results on both synthetic and real images demonstrate the effectiveness of our method in providing more desirable shape reconstruction.

Keywords shape from shading, user interaction, tabu search

1 Introduction

Shading information provides one of the key clues to understand 3D structure from 2D images. A particular well-studied problem is to estimate 3D shape from pixel intensities in a single 2D image. This task is known as *shape-from-shading* (SFS) [1]. However, although it may reconstruct local shape details satisfactorily, SFS often fails to achieve correct reconstruction of overall structure due to its ill-posed nature [2, 3]. One main obstacle is the concave-convex ambiguity [2]—a single view point cannot distinguish between swapped convexity and concavity. In this paper, we analyse optimisation based SFS [1, 4, 5], and note that poor initialization can be a main cause of such concave-convex errors. We thus suggest a user-guided approach to improve initialization, which can provide a more desirable overall shape after reconstruction.

As summarized in two survey papers [6, 7], many SFS methods rely on optimisation. A classic solution is the variational approach [1] which iteratively estimates surface normals such that they provide balanced agreement with the observed intensities and a smoothness constraint. However, this method is known for its over-smoothing tendency. Worthington and Hancock [4] addressed this weakness by enforcing the surface normal to lie on the image irradiance cone, thus agreeing with the intensities as a hard constraint. Based on this optimisation framework, Huang and Smith [5] further proposed a structure preserving SFS method which uses a weighting scheme in the smoothness regularisation. This better preserves shape structures, and is thus the chosen underlying SFS method used in this paper.

To improve overall shape reconstruction, several interactive SFS approaches [8, 9, 10, 11] have been

proposed. Based on a graph constructed from singular points in intensity, Meyer *et al* [9] suggested to let the user correct reconstruction errors and add details by adjusting the relative heights of some singular points. However, due to the use of singular points, their approach is only applicable to images with frontal lighting. Zeng *et al* in [8] proposed to let the user specify the surface normals at some key locations. Then, local shapes around these locations are recovered and stitched together to form a global surface. The results reported are faithful and plausible. However, it is not easy for users to identify the critical locations and provide accurate normals there. In [10], the user makes relative changes to problematic normals by rotating them on a sphere. This is more intuitive for the user than specifying absolute normals. Our method can be viewed as further restricting the rotation to adhere to the image irradiance cone [4], ensuring that the resulting normal is always compatible with the image intensity—this has to be enforced separately in [10]. Furthermore, we simplify the user interaction required. The user simply indicates areas judged to be good or poor reconstructions, rather than having to carefully edit normals or heights, which can be tricky to get right.

The contributions of this paper are two fold. Firstly, instead of using a single SFS initialization, we make use of a set of initializations which form a finite search space. Secondly, we make use of a heuristic search method—tabu search—to find a suitable initialization to improve SFS reconstruction. Tabu search works by optimising an objective function, which in this case is not computed by an algorithm, but is instead provided via user interaction. Then, the objective depends upon the user’s understanding of the correct shape. There have been works making use of heuristic search, such as genetic algorithm [12] and simulated annealing [13], to solve SFS. The purposes of using these heuristic search strategies are, in general, the same as ours: to escape from the local optima that re-

sulted from traditional optimisation based SFS. In [12] and [13], additional priors were introduced to guide the global search, while in our approach, user judgements are used in place of these priors. Tabu search is used in our approach as it provides good performance and requires less time (and thus less user interaction) than alternatives such as simulated annealing and genetic algorithms [14].

In the next section, we further explain the motivation of the proposed method. Sections 3, 4, and 5 explain the main steps in detail, while Section 6 presents reconstruction results. Section 7 concludes the paper and offers possible future directions.

2 Problem Overview and Solution



(a) Results using initialization from negative image gradients



(b) Results using initialization from occluding boundary

Fig. 1. Influence of SFS initialization on final reconstruction, using two initialization schemes. Left: reconstruction directly from initialization normals themselves. Center: reconstruction after optimisation using Horn and Brooks’ variational approach [1]. Right: reconstruction after optimisation using Worthington and Hancock’s SFS framework [4].

Fig. 1 illustrates how initialization can influence the final SFS reconstruction. The input is a synthetic face image of Julius Caesar (see Fig. 6). Fig. 1(a) shows reconstructions where surface normals are initialized from negative image gradients, as used in [4]; Fig. 1(b) shows reconstructions where normals are ini-

tialized using linear interpolation of the normals at the boundary, as used in [15]. Two SFS optimisation approaches have been used [1, 4], and the reconstructed surfaces are shown in the middle and right columns of Fig. 1. In comparison to the ground-truth (in Fig. 6), we can see that no matter which SFS optimisation approach is used, reconstructions from the same initialization share common concave-convex errors, and these were already present in the initialization. The aim of this paper is to improve the reconstruction by correcting such undesired errors in the initialization, by means of simple user interaction.

Negative image gradient initialization is used as the basis for initialization in this paper. It estimates the surface normal at each pixel by (i) forcing the normal to lie on its irradiance cone, ensuring it is consistent with the observed intensity, and (ii) setting the direction of the projected normal in the image plane in the *opposite* direction to the image gradient. This approach assumes global convexity of the surface—the local bright parts in the image correspond to peaks in the surface. It is straightforward to make the opposite assumption that the local bright parts correspond to valleys, by setting the direction of the projected normal to be in the *same* direction as the image gradient (we call it a *positive* image gradient). Other assumptions can be made, e.g. that it corresponds to a saddle which is convex in the x direction but concave in y , or the opposite. Complex surfaces often have regions of all these shapes, so a global assumption is inappropriate, resulting in convex-concave errors. In turn, this suggests that different negative or positive image gradients should be used for the initialization of different regions. This observation is the foundation of the method proposed in this paper.

Based on the above, for each pixel, we consider four choices for the surface normal. To simplify the description, we rotate the current coordinate system to make the positive z -axis coincide with the lighting direction. Suppose (n_x, n_y, n_z) is the negative image

gradient initialization in this rotated coordinate system. Then the other three choices are $(-n_x, n_y, n_z)$, $(n_x, -n_y, n_z)$, $(-n_x, -n_y, n_z)$. Changes of sign indicate switching the convex-concave assumption along the x and/or y axes. By rotating back, we get the four candidate surface normals at this pixel location in the original coordinate system. Clearly, it is infeasible for the user to specify such a choice for each pixel. For a smooth surface, convexity or concavity is typically consistent over local neighbourhoods with an intensity pattern characterised as bright surrounded by dark. Watershed based methods [16] are suited to provide a segmentation of this kind. Then the user simply has to make such a choice for each segmented region.

Based on these ideas, the framework of our method can be summarized in Fig. 2, and Procedure 1 lists the steps of our method.

As described in Procedure 1, there are 4^K possible initializations involved, where K is the number of segmented regions. These candidate initializations form a finite search space. For simple surfaces where $K \leq 3$, it is feasible to use exhaustive search to find the optimal initialization which gives the most plausible reconstruction. However, for complex surfaces with $K > 6$, the exponential number of possibilities renders this infeasible. This is why we make use of a general heuristic search method to find an initialization which gives a reasonable reconstruction. Tabu search is both effective and efficient for this purpose. It employs local search to find potential solutions, but makes use of a memory structure, called the tabu list, to prevent entrapment in local solution regions. An objective function is optimised through the search, which in our method is not computed by an algorithm, but is instead evaluated via user interaction.

Note that there are two optimisations in our method. One is the optimisation of SFS initialization via tabu search, and the other is the SFS optimisation which we carry out using a structure preserving variational approach [5]. The former, which is the focus of

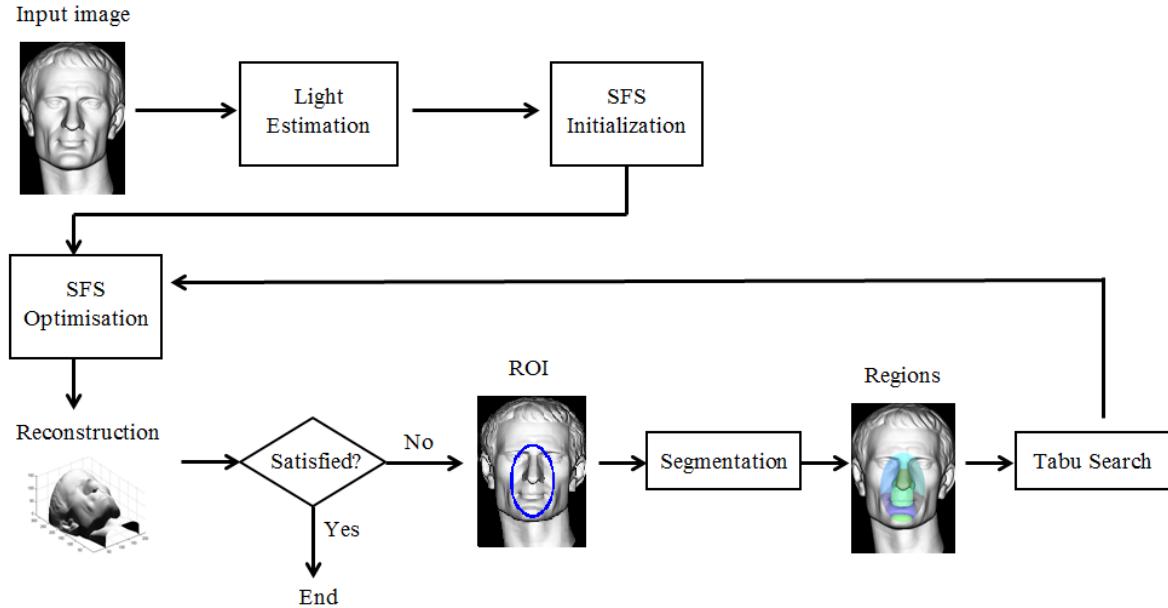


Fig. 2. Framework of our method.

Procedure 1. Steps in our method

1. (if required) estimate and correct the lighting direction
2. use the known or estimated lighting direction to initialize the surface normals with negative image gradients
3. using current initialization, reconstruct a surface via SFS optimisation
4. the user assesses the surface visually
5. if the current surface is satisfactory, output it and stop
6. otherwise, the user marks one or more problematic areas as regions of interest (ROI)
7. segment the ROI into K local regions, and associate each region with 4 candidate convex-concave patterns, which forms the search space
8. use tabu search (as explained below) to find a near optimal initialization in the current search space
 - (a) randomly generate a new candidate initialization which is a neighbour of the current initialization in the search space
 - (b) reconstruct a surface from the candidate initialization via SFS
 - (c) the user visually assesses the candidate surface and compares with current one
 - (d) put the undesired solution into tabu list to be avoided in following iterations.
 - (e) set the desired solution to be the current one
 - (f) go back to (a) and continue until satisfactory or the user wants to refine ROI
9. go back to step 3 and continue

this paper, aims to correct global and mid-scale structural errors, while the latter determines fine detail via local adjustment of surface normals.

3 Light Estimation and Shape from Shading

We follow the general assumption in orthographic SFS: Lambertian reflectance, uniform albedo, and a finite lighting at direction \mathbf{l} with unit intensity. Then the image irradiance equation can be written as:

$$\mathbf{I} = \mathbf{n} \cdot \mathbf{l} \quad (1)$$

where \mathbf{n} denotes surface normal. We make use of existing light estimation [15] and SFS [5] methods to estimate \mathbf{l} and \mathbf{n} . We also implement an interface as shown in Fig. 3 to allow the user to make corrections of the estimated lighting direction, and in turn to change the estimated surface normals.

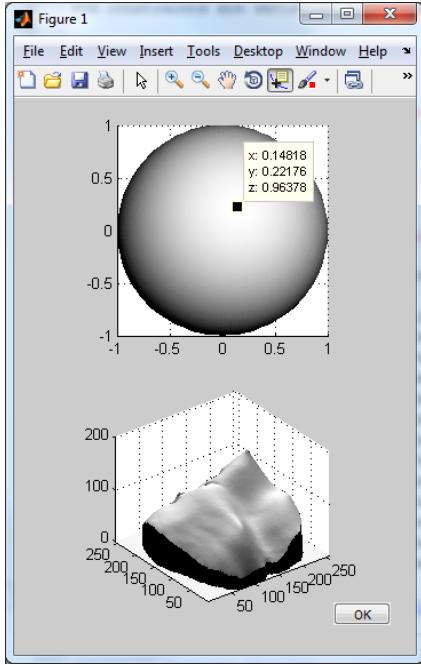


Fig. 3. Interface for the user to adjust lighting direction.

Structure-preserving SFS [5] is used in our approach to recover the surface normals. It introduces a weighting scheme formulated in Eq. (2) for the regularization of surface normals. The idea is that adjacent

pixels with closer intensities are more likely to have similar surface normal directions. As suggested by its name, this SFS method improves upon the classic variation SFS [1] on the preservation of shape structures.

$$\mathbf{n}^{(t+1)}(x, y) = \frac{\sum_{(i,j) \in \Omega(x,y)} \mathbf{W}(i, j) \mathbf{n}^{(t)}(i, j)}{\|\sum_{(i,j) \in \Omega(x,y)} \mathbf{W}(i, j)\|}, \quad (2)$$

In Eq. (2), $\Omega(x, y)$ denotes the 3×3 neighbourhood of pixel (x, y) , and $\mathbf{W}(i, j)$ is a measure of the intensity similarity between pixel (i, j) and the current pixel (x, y) . Applying the structure-preserving SFS in our approach, we use a Gaussian function (Eq.(3)) for the weights \mathbf{W} .

$$\mathbf{W}(i, j) = \exp\left(-\frac{(\mathbf{I}(i, j) - \mathbf{I}(x, y))^2}{2\sigma^2}\right), \sigma = 0.1. \quad (3)$$

The surface is finally integrated using the widely used integration algorithm of Frankot and Chellappa [17].

The above SFS approach needs a known lighting direction, which we estimated following the idea in [15]. Starting with an estimation of the surface normals interpolated from the boundary normals, the lighting direction \mathbf{l} is determined as the least-squares solution to Eq.(1), subject to non-negative lighting constraints [18]. However, the crude estimation of surface normals often cause errors in the estimated lighting direction. We thus implement the interface in Fig. 3 to let the user correct the lighting direction by rotation. Suppose \mathbf{l}_0 is the estimated lighting direction, from which an estimation of the surface normals, denoted \mathbf{n}_0 , can be obtained using the structure-preserving SFS. Both the lighting direction \mathbf{l}_0 and the reconstruction from \mathbf{n}_0 are displayed in the interface for user examination. If the user rotates the lighting direction to $\mathbf{l}_1 = \mathbf{R}\mathbf{l}_0$, where \mathbf{R} is the rotation matrix, then according to Eq. (1), $\mathbf{n}_1 = \mathbf{n}_0\mathbf{R}^{-1}$ is the corresponding surface normals (ignore the normals facing away from the user). Then the reconstruction from \mathbf{n}_1 is displayed. In this way, the user can rotate the lighting to the direction which gives the most desirable reconstruction.

4 Segmentation

The aim of segmentation is to divide the user marked ROIs into local regions in such a way that each region can be consistently approximated by just one of the four candidate convex-concave patterns. As noted in Section 2, such regions have a local intensity pattern of bright surrounded by dark. Suppose \mathbf{I} represents the intensity image with values in the range $[0, 1]$. The inverted image $1 - \mathbf{I}$ can be seen as a topographic relief with the values interpreted as altitudes. Then, the targeted local regions form basins in the relief, and a segmentation of these regions can be achieved by applying the watershed transformation [16].

However, due to noise and/or fine details in images, direct watershed transformation typically results in over-segmentation. Thus, a image filtering is first applied to remove noise and unnecessary details before the watershed transformation. We tried mean/median filters, bilateral filter [19], and L0 smoothing proposed in [20]. Among them, bilateral filter [19] gives the most reasonable results in terms of reducing the number of segments, and thus is the chosen filter in our approach. After the initial segmentation using watershed transformation, we further combine adjacent segments to reduce the number to be within a manageable range. The combination is based on the physical meaning of watershed: segments correspond to catchment basins. We calculate the shortest path between the bottoms of all adjacent basins. The two basins with the smallest path are combined. The process is repeated until all segments have been combined; the segmentation at each stage is stored for later use. Fig. 4 shows examples of segmentation at different stages. A large K (number of segments) results in many small regions which slows the search but provides finer user control, while a small K results in larger regions with more complex convex-concave patterns. An experiment examining a suitable choice of K will be presented in Section 6. Based on the experiment, we set the default

value of K to be $9m$, where m is the number of disjoint ROIs; the user can easily adjust K using a slider for more precise control, as shown in Fig. 4.

5 Tabu Search with User Interaction

The segmented regions are represented by an array of K variables each of which has an integer value in the range $[0, 3]$, representing the 4 candidate convex-concave patterns. An assignment of values represents a possible initialization in the search space. Our goal is to find an assignment which represents an initialization that gives a desirable reconstruction.

Table 1. An Example of a Tabu List. ‘X’ marks a banned value.

Segment	Current Status	Move to			
		0	1	2	3
1	0		X		
2	1				X
3	0				
:	:	:	:	:	:
K	3	X	X	X	

The search strategy in our application is based on tabu search [21, 22]. The initial solution—an array of zeros—represents negative image gradient initialization throughout the whole image. A *move* is defined as a value change of an entry in the current array. A move can be *good*, *bad*, or *undecided* as judged by the user after a visual inspection of the corresponding change to the reconstructed surface. A short-term tabu list records banned values for each entry in the recent past—less than t iterations ago. t is called the *tabu tenure*. If a move is judged to be *bad*, the *move to* value for this entry is marked ‘banned’ in the tabu list. If a

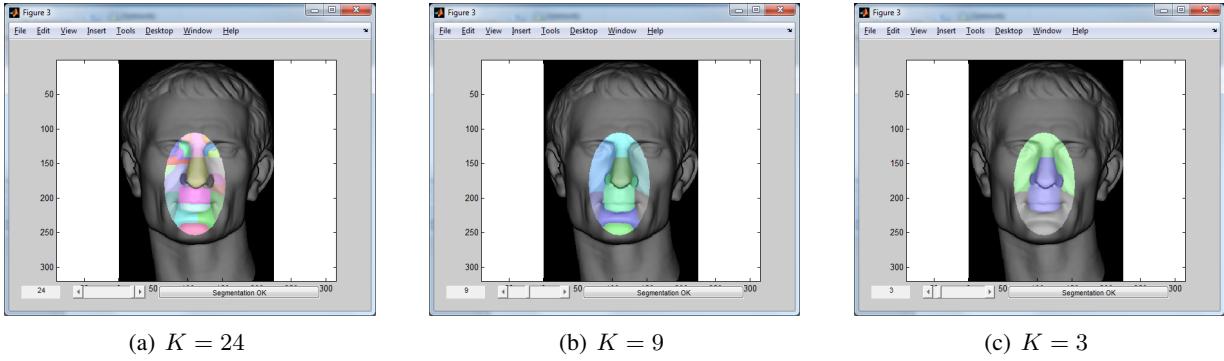


Fig. 4. Segmentation of ROIs. K is the number of segments. There is just one ROI covering the centre of the face in this example.

move is judged to be *good*, all other 3 values for this entry are marked ‘banned’. Table 1 shows an example of a tabu list with ‘X’ marking a ban. In our application, we set the tabu tenure to be 4, which means a banned entry value will be re-allowed after 4 iterations. Moreover, since the user can make poor judgements, two more rules are used: (i) if all 4 values for an entry are banned, they are all re-allowed, and (ii) if a *good* move to value is later judged to be *bad*, all other 3 values for that entry are re-allowed.

The immediate neighbourhood of the current solution includes $4K - 1$ candidates, resulting from a single move. Because of the limits to the amount of interaction a user can reasonably perform, not all candidate solutions can be examined. Instead, for each entry, we set a probability for each of the 4 values. The probability indicates how likely we believe the value for an entry is correct. Initially, the probabilities are distributed equally, i.e. 0.25 for each value. A *bad* move will decrease the probability to 0.1 (empirically chosen), while the probabilities of the other 3 values in the same entry will be equally increased to keep the sum to be 1. Similarly, a *good* move will increase the probability to 0.7 (empirically chosen as well), while the others in the entry will be decreased accordingly. During each iteration, a set of moves is generated by stochastically choosing an un-banned candidate for each region ac-

cording to their probabilities. A new initialization is obtained by taking all the moves, and a surface is reconstructed from the new initialization. These moves are then judged by their effects on the new reconstruction. Although there may be many ‘moves’, only those with visually obvious improvements or deteriorations will be judged to be *good* (using a left click on the region), or *bad* (using a right click), and the others are left to be *undecided* (no action is required). After examination, an improved solution is achieved by replacing the entries in the current array with those corresponding to *good* moves. Iteration then continues until satisfactory. Algorithm 1 summarizes our tabu search process.

Fig. 5 shows the interface for user interaction. Four panels are shown during each iteration. Panel (a) shows the original image for user reference. Panel (b) shows the regions that need user judgements. After stochastic moves in these regions, a new reconstruction is found and shown in panel (c). The user visually inspects it, and compares it to the current reconstruction (shown in panel (d)).

6 Experimental Results

Experiments have been carried out on both synthetic and real-world images. All images are openly

¹ <https://users.cs.cf.ac.uk/J.Wu/>

Algorithm 1: Tabu search to find an alternative SFS initialization

Data: first initialization: \mathbf{n}_0

Result: a better alternative initialization: \mathbf{n}

initialize status vector: $\mathbf{statusV} \leftarrow \text{zeros}(K, 1)$;

initialize probability matrix: $\mathbf{ProbMat} \leftarrow 0.25 * \text{ones}(K, 4)$;

initialize tabu list: $\mathbf{TabuList} \leftarrow \text{zeros}(K, 4)$, set tabu tenure: $t = 4$;

reconstruct surface \mathbf{Z} from \mathbf{n}_0 ;

while not satisfied with \mathbf{Z} **do**

- $\mathbf{TabuList} \leftarrow \mathbf{TabuList} - 1$;
- for** each region k **do**

 - repeat**

 - random generate $\mathbf{statusV}_{new}(k)$ according to $\mathbf{ProbMat}(k, :)$;
 - until** $\mathbf{TabuList}(k, \mathbf{statusV}_{new}(k)) < 0$;

 - end**
 - get new initialization \mathbf{n}_{new} from $\mathbf{statusV}_{new}$;
 - reconstruct surface \mathbf{Z}_{new} from \mathbf{n}_{new} ;
 - for** each region k **do**

 - if** \mathbf{Z}_{new} improved from \mathbf{Z} within and around region k **then**

 - $\mathbf{TabuList}(k, \sim \mathbf{statusV}_{new}(k)) = t$;
 - $\mathbf{ProbMat}(k, \mathbf{statusV}_{new}(k)) = 0.7$;
 - reduce $\mathbf{ProbMat}(k, \sim \mathbf{statusV}_{new}(k))$ accordingly;
 - $\mathbf{statusV}(k) = \mathbf{statusV}_{new}(k)$;

 - else**

 - $\mathbf{TabuList}(k, \mathbf{statusV}_{new}(k)) = t$;
 - $\mathbf{ProbMat}(k, \mathbf{statusV}_{new}(k)) = 0.1$;
 - increase $\mathbf{ProbMat}(k, \sim \mathbf{statusV}_{new}(k))$ accordingly;

 - end**

 - end**
 - get initialization \mathbf{n} from $\mathbf{statusV}$;
 - reconstruct surface \mathbf{Z} from \mathbf{n} ;

end

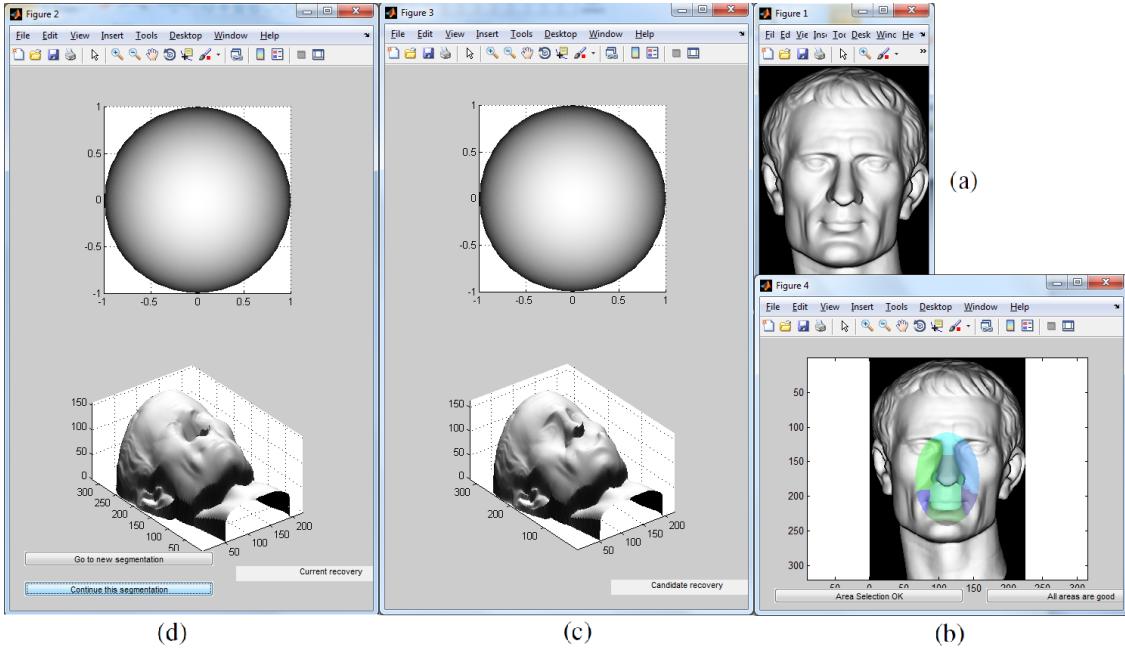


Fig. 5. Using the user interface for our interactive SFS. Panel (a): original image for user reference; panel (b): the segmented regions requiring user judgements; panel (c): new reconstruction with stochastic moves; and panel (d): current reconstruction to be improved.

available¹, except for the ‘horse’ image in Fig. 11 (b) which can be obtained from [10]. Evaluation of the results relies on (i) visual inspection of the reconstructed surfaces, and (ii) for synthetic images, reconstruction errors compared with ground truth on both surface normals and integrated depths.

6.1 Results on synthetic images

Fig. 6 and 7 display reconstructions from two synthetic images: Caesar² and bunny Hi³. Images were generated under orthographic projection, frontal lighting, and Lambertian reflectance. The reconstructions with user interaction were obtained after 1-2 refinements using the displayed segmentations. These reconstructions show visually improved overall shapes with reduced concave-convex errors. Fig. 8 and 9 quantitatively confirmed these improvements, showing error maps of the reconstructions compared with the ground-

truth in surface normal domain and depth domain respectively. Both the average angular deviations and depth errors are approximately halved by user interaction. However, our reconstructions still differ from the ground-truth. We believe it is due to two main causes. Firstly, the assumptions made in SFS are not always satisfied. Structure preserving SFS still assumes a smooth surface, and this causes some oversmoothing of certain features as can be seen in the reconstructions of Caesar’s face and the bunny Hi surface. Furthermore, our assumption that boundary normals lie in the image plane may be far from correct: an example can be seen in the bunny’s left ear. Secondly, the stochastic nature of general heuristic search will typically lead to an *improved* rather than *optimal* solution. As only limited user interaction is feasible, it is hard to achieve a satisfactory result when there are many fine details resulting in many small segments.

To explore how the fineness of segmentation in-

² AIM@SHAPE-VISIONAIR Shape Repository, 2006. <http://visionair.ge.imati.cnr.it/ontologies/shapes/>

³ The Stanford 3D Scanning Repository, 1996. <http://graphics.stanford.edu/data/3Dscanrep/>

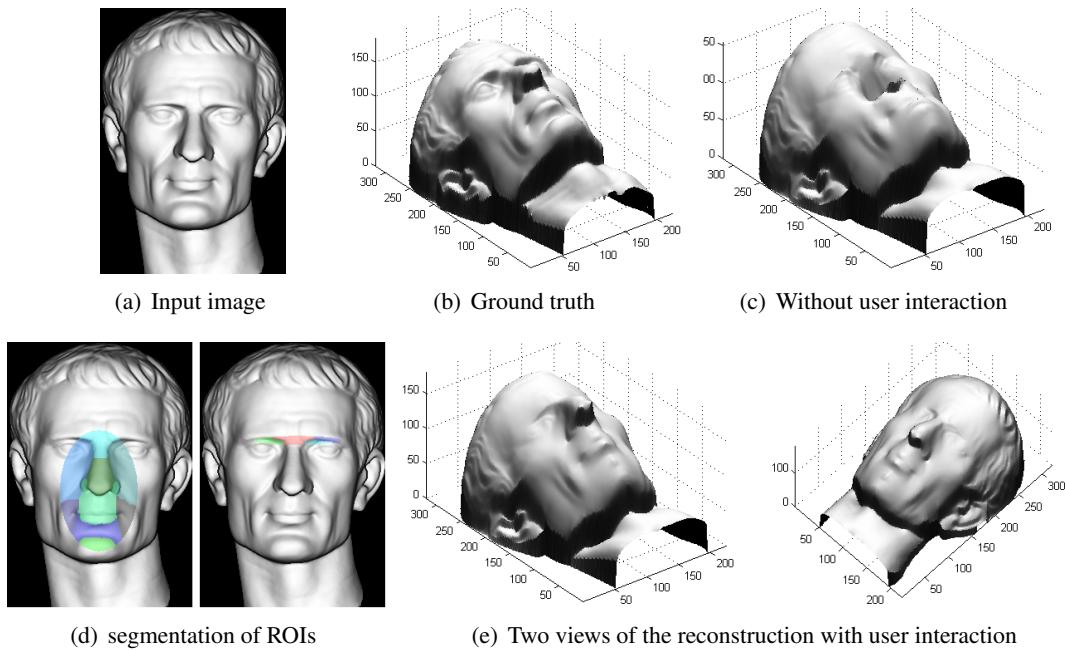


Fig. 6. Reconstruction from synthetic ‘Julius Caesar’ image.

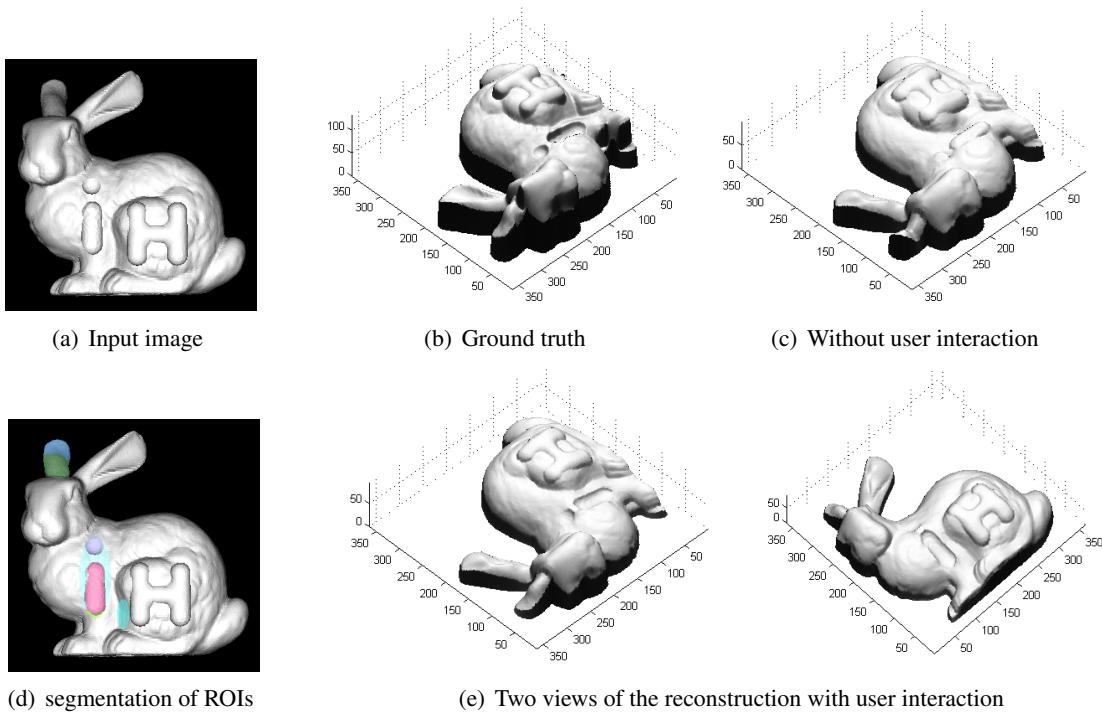


Fig. 7. Reconstruction from synthetic ‘bunny Hi’ image.

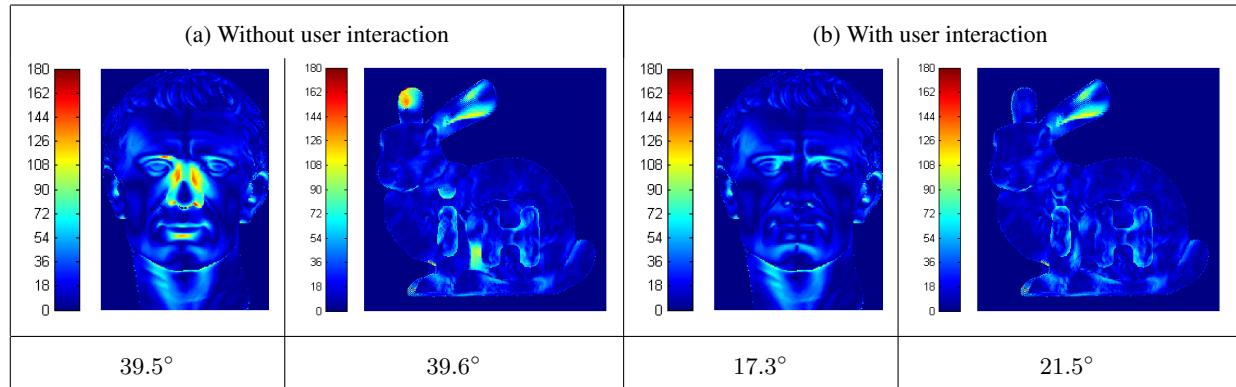


Fig. 8. Error maps of surface normals and average angular deviation from ground-truth

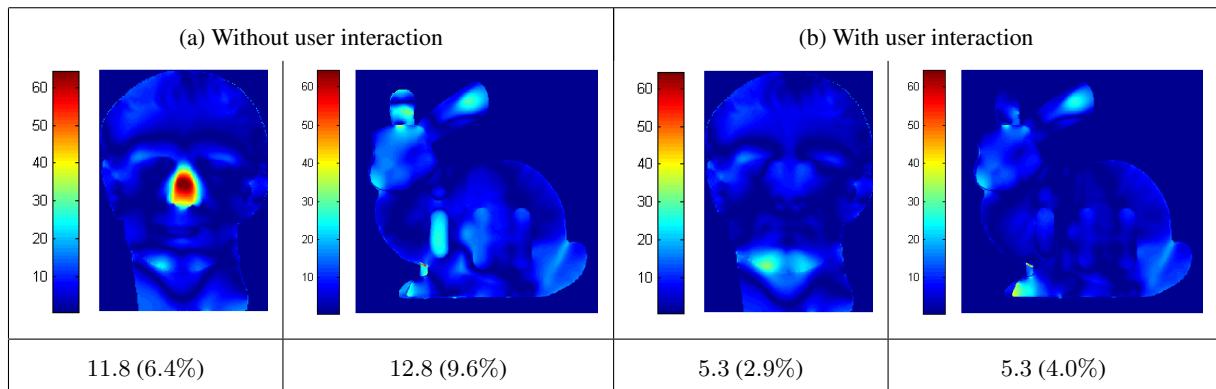


Fig. 9. Error maps of integrated depths and average depth error with its percentage to ground-truth depth

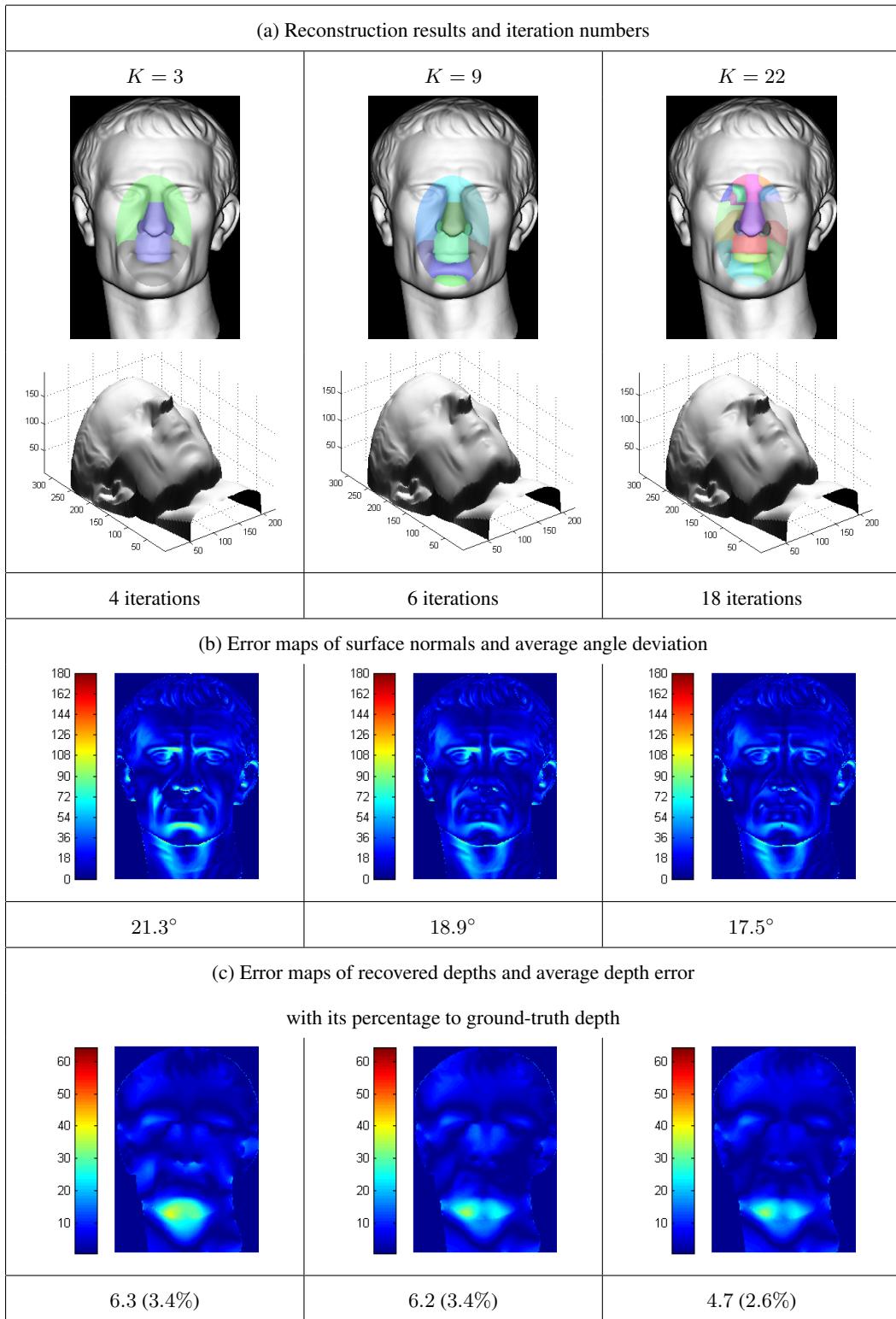


Fig. 10. Effects of segmentation numbers

fluences the accuracy of reconstruction and the experience of user interaction, we carry out a further experiment on the ‘Julius Caesar’ image. Using the slider shown in Fig. 4, we separately choose three segmentation fineness levels: ‘coarse’ $K = 3$, ‘default’ $K = 9$, and ‘fine’ $K = 22$, and without further ROI refinements, we interactively improve the reconstructions as much as we can using each of the segmentations. The segmentations and corresponding reconstructions and error maps are shown in Fig. 10, together with the iteration numbers required. It is clear that with finer segmentation, the user gets more accurate result, at the cost of more iterations and thus more interactions. For each marked ROI, we empirically set the default number $K = 9$ which, in this example, was demonstrated to be a good balance between accuracy and ease of use with clearly reduced surface normal errors in two more iterations. In general, we suggest the user should use a moderately small K , between 6 and 9, and achieve fine details through further refinements with smaller ROIs.

6.2 Results on real-world images

Given a colored real-world image, we first transform it into HSL color space and use the L channel as input. Then, we estimate the lighting direction and correct it using the interface shown in Fig. 3 if required. Then the reconstruction is carried out using the corrected lighting direction. Fig. 11 shows the real-world images used in our experiments and the estimated/corrected lighting directions.

Fig. 12 compares our reconstructions on the well-known pepper image with those from three traditional SFS methods [23, 24, 25]. All the surfaces are illuminated from the lighting direction in Fig. 11 (a). It seems that our reconstructions, even without user interaction, outperform the traditional SFS reconstructions, confirming the effectiveness of structure preserving SFS [5]. With user interaction, the distortion at the right side of the pepper is very well corrected. However, the pepper’s stem is not correctly recovered be-

cause of its depth discontinuity—such problems still provide a challenge to SFS.

Fig. 13 compares our reconstruction from the pepper image with those in an earlier interactive SFS paper [8], while Fig. 14 compares our reconstruction from a horse image with those obtained in another interactive SFS paper [10]. In the first comparison, our results are illuminated with the same lighting direction as in Fig. 12. In the latter case, the results are illuminated with lighting direction $(-1, 1, 1)$ as in [10] to enable a direct comparison. Our reconstructions have comparable overall shapes to those provided by these earlier interactive methods. The method in [8] better reconstructs the pepper stem as it stitches local reconstructions. The method in [10] recovers more details of the horse’s hair as it is based on direct editing of surface normals. However, our method has a significant advantage over these earlier interactive SFS methods: the user interaction required in our method does not involve direct editing of surface normals, but simply judge whether an automatically determined change in the surface is more desirable or not. Users do not need the expertise or specific training in use of the surface normal domain, making our method more intuitive and usable by a wider user population.

To demonstrate that our method is simple and intuitive, Fig. 15 shows some reconstructions using our method and interface produced by a 7-year-old boy from images of his toys (Fig. 11(c), (d)), and the reconstructions produced by the first author of this paper. While the boy had a limited attention span, and as a result his reconstructions are generally not as good as those produced by the author, the improvements over the initial reconstructions are still clear.

6.3 Timing

Our experiments show we can achieve an obviously improved reconstruction within 15 iterations during the first round of refinement, and fewer iterations in the following rounds. A reasonable final reconstruction

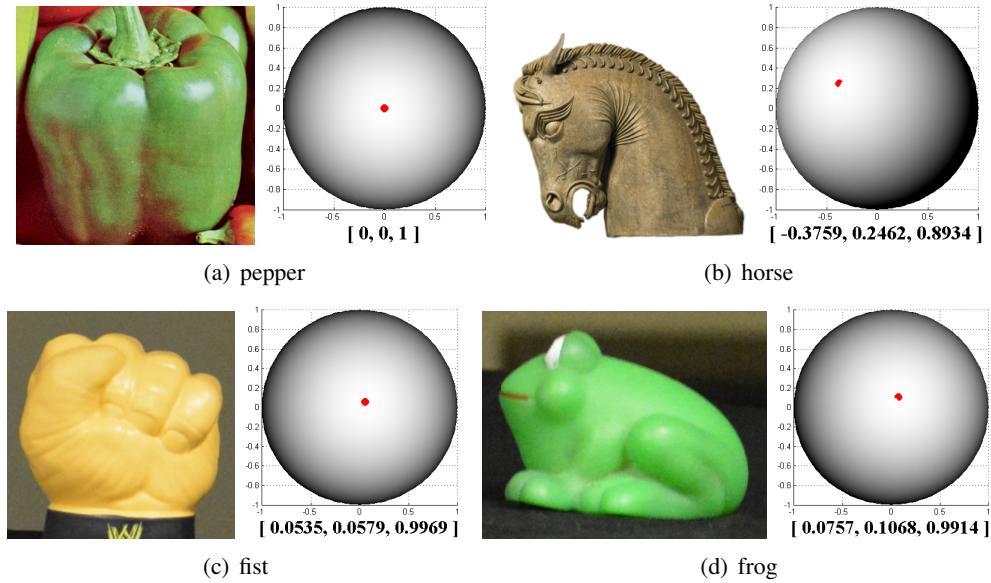


Fig. 11. Real-world images and their estimated/corrected lighting directions

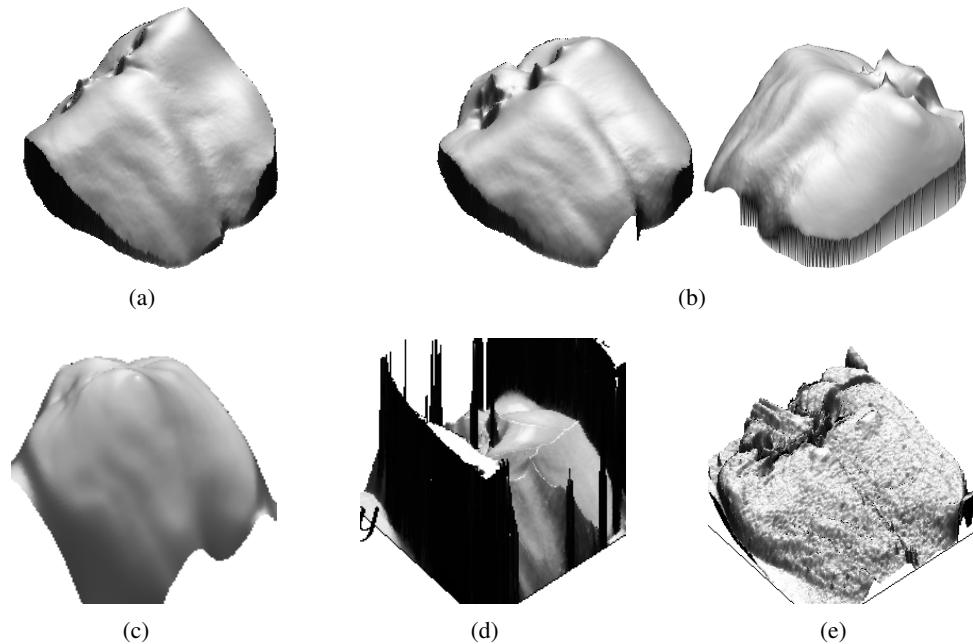
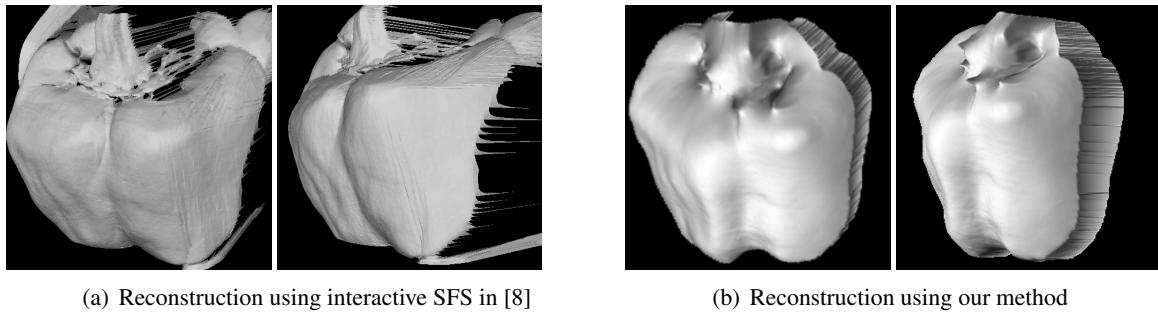


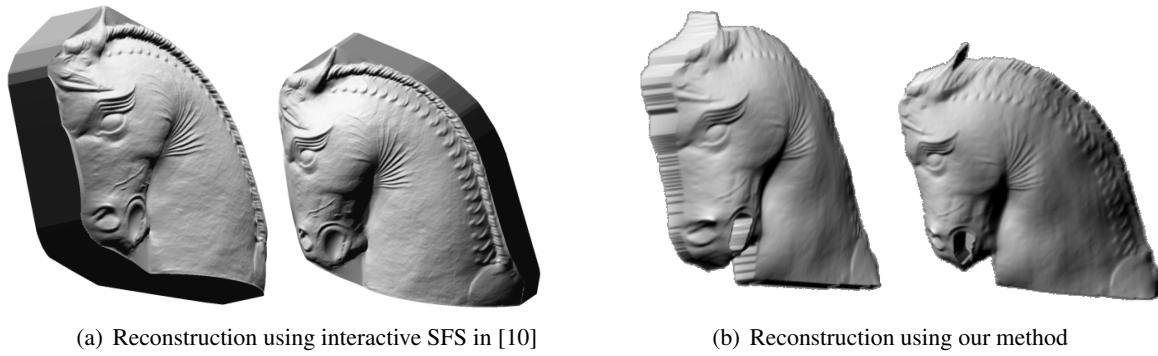
Fig. 12. Reconstructions from pepper image. (a) and (b) are our reconstructions: (a) is without user interaction, (b) is with user interaction; (c), (d), and (e) are reconstructions from [23], [24], and [25] respectively.



(a) Reconstruction using interactive SFS in [8]

(b) Reconstruction using our method

Fig. 13. Comparison with interactive SFS method [8]. ((a) Copyright ©2005, IEEE)



(a) Reconstruction using interactive SFS in [10]

(b) Reconstruction using our method

Fig. 14. Comparison with interactive SFS method [10]. ((a) Copyright ©2008 ACM, Inc.)

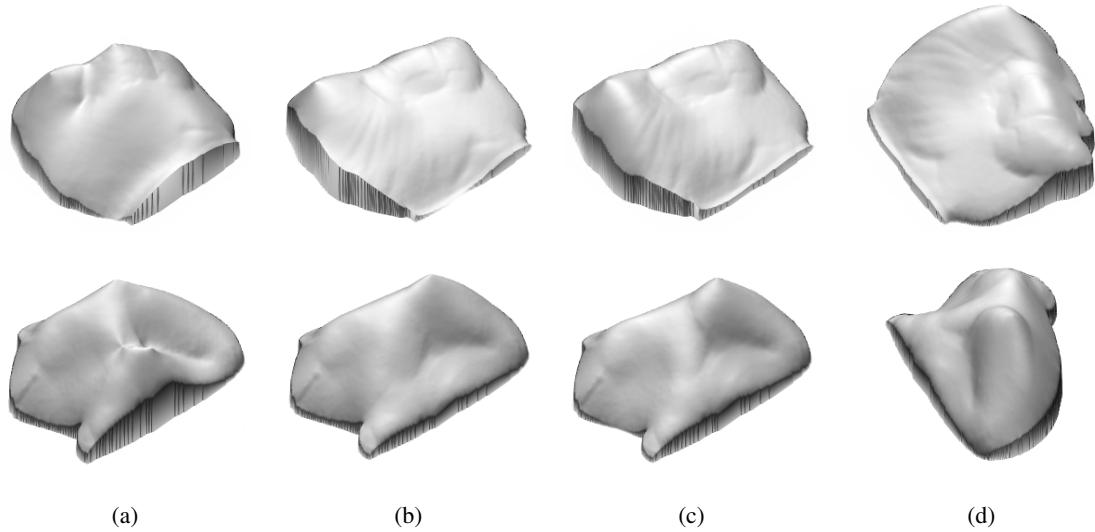


Fig. 15. Results produced by a 7 year old boy. (a): Reconstruction without user interaction; (b): reconstruction by the boy; (c) and (d): two views of the first author's reconstruction.

is usually achieved within 3 rounds of refinement. The whole process usually takes less than 20 minutes (our implementation is based on MATLAB R2013a, using a computer with a 2.70 GHz Intel Xeon E5 CPU processor). Most of the user's time is spent on rotating the model or light direction in 3D on the screen to assess the shape of different areas.

7 Conclusions

We have presented an interactive framework to improve shape recovery in shape from shading, based on better SFS initialization. The user identifies and improves poor local reconstruction via a tabu search process. Experimental results on both synthetic and real images demonstrate the effectiveness of our method, and also the simple intuitiveness of our method compared to other interactive SFS methods.

Our future work will focus on 1) further simplifying the user interaction by making use of belief propagation to associate probabilities with potential reconstructions; 2) exploring alternative SFS optimisation methods to the current structure-preserving one in order to improve the reconstruction of discontinuities.

References

- [1] Horn B K P, Brooks M J. The variational approach to shape from shading. *Computer Vision, Graphics, and Image Processing*, 1986, 33(2): 174-208.
- [2] Ramachandran V S. Perceiving shape from shading. *Scientific American*, 1988, 259(2): 76-83.
- [3] Belhumeur P N, Kriegman D J, Yuille A L. The bas-relief ambiguity. *International Journal of Computer Vision*, 2001, 35: 1040-1046.
- [4] Worthington P L, Hancock E R. New constraints on data-closeness and needle map consistency for shape-from-shading. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1999, 21(12): 1250-1267.
- [5] Huang R, Smith W A P. Structure-preserving regularisation constraints for shape-from-shading. In: *Proc. the 13th International Conference on Computer Analysis of Images and Patterns*. Sept. 2009, pp. 969-976.
- [6] Durou J -D, Falconeb M, Sagonab M. Numerical methods for shape-from-shading: a new survey with benchmarks. *Computer Vision and Image Understanding*, 2008, 109(1): 22-43.
- [7] Zhang R, Tsai P -S, Cryer J E, Shah M. Shape from shading: a survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1999, 21(8): 690-706.
- [8] Zeng G, Matsushita Y, Quan L, Shum H -Y. Interactive shape from shading. In: *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2005*. June 2005, pp. 343–350.
- [9] Meyer A, Briceno H M, Bouakaz S. User-guided shape from shading to reconstruct fine details from a single photograph. In: *Proc. the 8th Asian Conference on Computer Vision*. Oct. 2007.
- [10] Wu T -P, Sun J, Tang C -K, Shum H -Y. Interactive normal reconstruction from a single image. *ACM Transactions on Graphics—Proc. ACM SIGGRAPH Asia 2008*, 2008, 27(5), article no. 119.
- [11] Governi L, Furferi R, Puggelli L, Volpe Y. Improving surface reconstruction in Shape from Shading using easy-to-set boundary conditions. *International Journal of Computational Vision and Robotics*, 2013, 3(3): 225-247.

- [12] Saito H, Usami K. Shape from shading using genetic algorithm. In: *Proc. the IECON '93., International Conference on*. Nov. 1993, pp. 1620–1625.
- [13] Crouzil A, Descombes X, Durou J -D. A multiresolution approach for shape from shading coupling deterministic and stochastic optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2003, 25(11): 1416-1421.
- [14] Sinclair M. Comparison of the performance of modern heuristics for combinatorial optimization on real data. *Computers & Operations Research*, 1993, 20(7): 687-695.
- [15] Huang R, Smith W A P. Shape-from-shading under complex natural illumination. *IEEE International Conference on Image Processing 2011*. Sept. 2011, pp. 13-16.
- [16] Meyer F. Topographic distance and watershed lines. *Signal Processing*, 1994, 38(1): 113-125.
- [17] Frankot R T, Chellappa R. A method for enforcing integrability in shape from shading algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1988, 10(4): 439-451.
- [18] Basri R, Jacobs D W. Lambertian reflectance and linear subspaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2003, 25(2): 218-233.
- [19] Tomasi C, Manduchi R. Bilateral filtering for gray and color images. In: *Proc. IEEE International Conference on Computer Vision 1998*. Jan. 1998, pp. 839-846.
- [20] Xu L, Lu C, Xu Y, and Jia J. Image smoothing via L0 gradient minimization. *ACM Transactions on Graphics—Proc. ACM SIGGRAPH Asia 2011*, 2011, 30(6), article no. 174.
- [21] Glover F. Tabu search—part 1. *ORSA Journal on Computing*, 1989, 1(2): 190-206.
- [22] Glover F. Tabu search—part 2. *ORSA Journal on Computing*, 1990, 2(1): 4-32.
- [23] Daniel P, Durou J -D. From deterministic to stochastic methods for shape from shading. In: *Proc. the 4th Asian Conference on Computer Vision*, Jan. 2000, pp. 187-192.
- [24] Falcone M, Sagona M. An algorithm for the global solution of the Shape-from-Shading model. In: *Proc. the 9th International Conference on Image Analysis and Processing (volume I)*, Sept. 1997, pp. 596-603.
- [25] Tsai P -S, Shah M. Shape from Shading Using Linear Approximation. *Image and Vision Computing*, 1994, 12(8): 487-498.



Jing Wu received her BSc and MEng in computer science and technology from Nanjing University, China in 2002 and 2005 respectively. She received her PhD in computer science from the University of York, UK in 2010. She is currently a research associate at the School of Computer Science & Informatics, Cardiff University, UK. Her research interests include 3D reconstruction, shape-from-shading, computer-aided bas-relief generation, 3D face analysis and recognition.



Paul L. Rosin is a professor at the School of Computer Science & Informatics, Cardiff University. Previous posts include lecturer at the Department of Information Systems and Computing, Brunel University London,

UK, research scientist at the Institute for Remote Sensing Applications, Joint Research Centre, Ispra, Italy, and lecturer at Curtin University of Technology, Perth, Australia. His research interests include the representation, segmentation, and grouping of curves, knowledge-based vision systems, early image representations, low level image processing, machine vision approaches to remote sensing, methods for evaluation of approximation algorithms, medical and biological image analysis, mesh processing, non-photorealistic rendering and the analysis of shape in art and architecture.



Xianfang Sun received his PhD in Control Theory and Its Application from the Institute of Automation, Chinese Academy of Sciences in 1994. He is currently a lecturer at the School of Computer Science & Informatics, Cardiff University, UK. His main research interests include computer vision, computer graphics,

pattern recognition and artificial intelligence.



Ralph R. Martin obtained his PhD in Engineering in 1983 from Cambridge University. Since then he has been at Cardiff University, where he now holds a Chair and leads the Visual Computing research group. He is also a Guest Professor at Tsinghua and two other Universities in China, and is Director of Scientific Programmes of the One Wales Research Institute of Visual Computing. His publications include over 250 papers and 14 books covering such topics as solid modelling, surface modelling, reverse engineering, intelligent sketch input, mesh processing, video processing, computer graphics, vision based geometric inspection, and geometric reasoning. He is a Fellow of the Learned Society of Wales, the Institute of Mathematics and its Applications, and the British Computer Society. He is on the editorial boards of “Computer Aided Design”, “Computer Aided Geometric Design”, “Geometric Models”, and other journals.