HAL
open science

Towards Full 3D Helmholtz Stereovision Algorithms

Amael Delaunoy, Emmanuel Prados, Peter Belhumeur

To cite this version:

Amael Delaunoy, Emmanuel Prados, Peter Belhumeur. Towards Full 3D Helmholtz Stereovision Algorithms. ACCV 2007 - 8th Asian Conference on Computer Vision, Nov 2010, Queenstown, New Zealand. pp.39-52, 10.1007/978-3-642-19315-6_4 . inria-00525867

HAL Id: inria-00525867

https://inria.hal.science/inria-00525867

Submitted on 14 Oct 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Towards Full 3D Helmholtz Stereovision Algorithms

Amaël Delaunoy[1], Emmanuel Prados[1], and Peter N. Belhumeur[2]

[1] INRIA Grenoble / LJK
[2] Columbia University

Abstract. Helmholtz stereovision methods are limited to binocular stereovision or depth maps reconstruction. In this paper, we extend these methods to recover the full 3D shape of the objects of a scene from multiview Helmholtz stereopsis. Thus, we are able to reconstruct the complete three-dimensional shape of objects made of any arbitrary and unknown bidirectional reflectance distribution function. Unlike previous methods, this can be achieved using a full surface representation model. In particular occlusions (self occlusions as well as cast shadows) are easier to handle in the surface optimization process. More precisely, we use a triangular mesh representation which allows to naturally specify relationships between the geometry of a point of the scene and its surface normal. We show how to implement the presented approach using a coherent gradient descent flow. Results and benefits are illustrated on various examples.

1 Introduction

Reconstructing shape and appearance of objects from images is still one of the major problems in computer vision and graphics. In this work, we are interested in recovering a full and dense representation of the object's three-dimensional shape. Multiview reconstruction systems are commonly used to estimate such a model, as they provide information from many viewpoints around the object of interest. Among these approaches, variational methods have been popular because they can be used to solve a wide variety of vision problems. The idea is to minimize an energy functional that depends on the considered object surface and on the input images, whose minima is reached at the object of interest. Many methods have been proposed in order to solve this problem, but these approaches are often limited in the kind of appearance they can handle. To overcome these limitations, we exploit Helmholtz reciprocity and propose a single framework for normal estimation and normal integration using a triangular mesh-based deformable model.

1.1 Helmholtz Stereovision: a Reconstruction Approach for Real World Objects

Most of the multiview reconstruction algorithms rely on image correspondences (as done for instance in multiview stereo [1]) or shading (using the normal information in multiview shape from shading [2, 3] or multiview photometric stereo [4]). When texture information (stereo case) is good enough or Lambertian assumption is sufficiently verified, those methods have been proved to give good results with surfaces that are nearly. They then obtain either accurate correspondences or accurate normal estimates.

But when the scene is not Lambertian, which is the case for most (if not all) real world scenes, such cues are not valid and algorithms fail to reconstruct accurately the surface. In order to solve this problem, many alternatives have been proposed. Some authors consider specular highlights as outliers [4], and consider a large number of images in order to compensate. Some others modify the input images to have specular free images, and be photometric invariant [5, 6]. Another approach is to have a robust similarity measurement [7, 8]. All these methods try to compensate the non-Lambertian components in order to run reconstruction algorithms designed for the Lambertian case.

Some authors consider another strategy: they propose to use more general reflectance / radiance parametric models. See for instance [9–11]. In practice, these approaches suffer from several limitations. First of all the reflectance model has to be known in advance and this constrains scene to be composed by materials consistent with the chosen reflectance model. Such algorithms tend to solve non-linear systems of thousands of variables (one reflectance / radiance model per surface point), or need additional assumptions (single or fixed number of materials, single specular component, etc.). Those models are difficult to be optimized and generally require to alternatively estimate the reflectance and the shape. They are numerically unstable and easily tend to get stuck in local minima [3]. Moreover, the algorithms are generally ill-posed, and then require strong regularization which over-smooth the obtained results. Finally, the reflectance and illumination models need also to be approximated. Although such algorithms show reasonable results for perfect synthesized scenes, their application to real-world scenes is complex and requires accurate camera and light calibration.

A different approach introduced by [12] uses radiance samples of reflectance exemplars to match it with the observed images. A direct matching allows to estimate the surface normal in order to reconstruct the shape by normal integration. Although it can deal with many materials and anisotropic BRDF, material samples are needed. This can be a restrictive assumption in concerned applications.

During the last decade, some authors have proposed to use Helmholtz reciprocity in order to perform 3D reconstruction [13–17]. In practice, Helmholtz reciprocity is exploited by taking a pair of images under a single light source, where camera centers and light positions are exchanged at each shot. It uses the fact that in this particular setup, for a single reciprocal pair, the relationship between two radiances of a single surface point is independent of the reflectance. Contrary to works described in previous paragraphs, methods based on Helmholtz reciprocity [13–17] allow to accurately estimate the normals at one point, independently of the reflectance model. This can be used to obtain a 3D surface. In this context, modeling the reflectance, having material samples or being photometric invariant is not required. Nevertheless, contrary to most of multiview stereovision algorithms, the state of the art in Helmholtz reconstruction is limited to depth map reconstructions. In this paper, we push the envelope by proposing a Full 3D multiview Helmholtz stereovision method.

1.2 A Surface-Based Approach

Until now, all the previous Helmholtz reconstruction methods were camera-view centered. On the contrary, we propose to change the surface representation and to adopt an

object-centered strategy. Thus, instead of using a 2.5D surface as it used to be done previously, we represent the object's surface by a closed and dense 2D manifold embedded in the euclidean 3D space. The interest of this choice is two-fold. First this allows to naturally recover a full 3D surface when the other approaches only recover a depth map (or a needle map). Secondly, this allow to easily and properly handle visibility (shadows as well as self occlusions). Figure 1 sheds light on these advantages: the picture on the left illustrates the case of conventional Helmholtz Stereovision methods, that recover a surface based on a virtual view (here in red). The drawing on the right illustrates the proposed approach which is surface-based instead of view-based reconstruction. The red contours show the optimal surface each method can recover. Clearly, the conventional approaches are confronted with difficulties with visibility since it generates discontinuities in the depth maps and the needle maps. Also, this difficulty spreads at the integration steps which behave badly in presence of discontinuities.

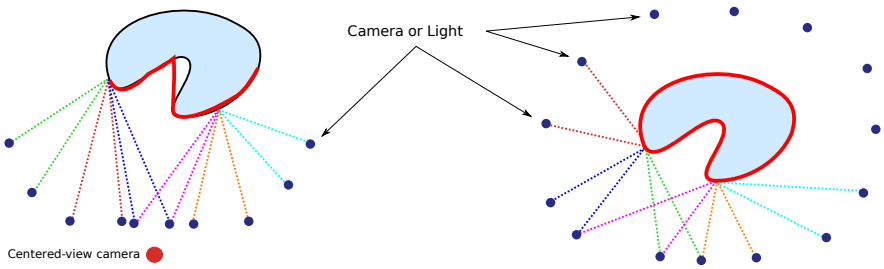

Fig. 1. Recovered surfaces by conventional Helmholtz Stereopsis versus proposed approach (surface in red). Left: case of conventional Helmholtz Stereo methods which recovers the shape based on a virtual view (in red). Right: the proposed surface-based approach.

In this work, as we will discuss in section 3, we represent the 2D surface using a triangle mesh. In our context, where the surface normals play a key role, this representation offers significant advantages. In particular, it intrinsically links a depth to its surface normal which is, in the case of piecewise planar surfaces like triangular meshes, well defined on each facet. Also, this allows us to naturally combine both the normal integration and normal estimation when these were two separate stages in most of previous approaches [13–15].

1.3 Contributions

This article presents several improvements to state-of-the-art 3D reconstruction techniques exploiting Helmholtz reciprocity. First, we model the problem as an energy minimization problem which is completely surface-based (when all previous methods were camera based). Secondly, we present a method for solving this problem and show how to implement it on triangular surface meshes by using a coherent discrete gradient flow. Finally, since we optimize a full surface, the method is able handle visibility. This also allows to fully reconstruct dense 3D surfaces of complex objects in a single framework.

2 Helmholtz Stereopsis - Variational Formulation

As described previously, Helmholtz reciprocity exploits the fact that BRDFs are generally symmetric and therefore, for any incoming angle $\hat{i}$ and outgoing direction $\hat{o}$, we have $\beta(\hat{i},\hat{o}) = \beta(\hat{o},\hat{i})$ where β is the BRDF function. By interchanging light and camera positions, one can exploit the constraint on β in radiance equations in order to solve 3D reconstruction problems.

Given a camera – light pair, one can write the radiance equation I_c of a scene seen from a camera c as:

$$I_c = \alpha\beta(\mathbf{v}_c, \mathbf{v}_l)\frac{\mathbf{v}_l \cdot \mathbf{n}}{|\mathbf{v}_l|^3} \ , \tag{1}$$

where $\mathbf{v}_c$ is the vector from the camera center to the point, $\mathbf{v}_l$ is the vector from the light center to the view-point and α is a constant. $\mathbf{n}$ is the surface normal of the considered point and $\beta(\mathbf{v}_c, \mathbf{v}_l)$ is the BRDF at the surface point (See Figure 2). The same radiance equation can be written for modeling the radiance I_l with the BRDF $\beta(\mathbf{v}_l, \mathbf{v}_c)$. Using Helmholtz reciprocity allows us to write $\beta(\mathbf{v}_c, \mathbf{v}_l) = \beta(\mathbf{v}_l, \mathbf{v}_c)$. This equality then defines the Helmholtz stereopsis constraint for all the point of the surface $\mathcal{S}$:

$$\left[\ I_c\ \frac{\mathbf{v}_c}{|\mathbf{v}_c|^3} - I_l\ \frac{\mathbf{v}_l}{|\mathbf{v}_l|^3}\ \right] \cdot \mathbf{n} = 0\ . \tag{2}$$

Now, we are going to formulate this constraint in the variational framework via a weighted area functional defined over the surface of the object.
We denote by $\pi_c(\mathbf{x})$ (resp. $\pi_l(\mathbf{x})$) the projection of a point in space $\mathbf{x}$ in the camera c (light l respectively), and I_c (or I_l) its corresponding intensity value in the image. For more clarity, we also denote

$$\mathbf{h}(\mathbf{x}) = \left[I_c(\pi(\mathbf{x}))\frac{\mathbf{v}_c}{|\mathbf{v}_c|^3} - I_l(\pi(\mathbf{x}))\frac{\mathbf{v}_l}{|\mathbf{v}_l|^3} \right]\ . \tag{3}$$

In this case, the surface that "best" verifies Equation (2) can be obtained by minimizing the following energy functional defined over the surface, with respect to the surface itself:

$$E_{HS}(\mathcal{S}) = \int_{\mathcal{S}} \left(\mathbf{h}(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x})\right)^2\ \nu_{\mathcal{S},c,l}(\mathbf{x})\ \mathrm{d}s\ , \tag{4}$$

where $\nu_{\mathcal{S},c,l}$ is the characteristic function such that $\nu_{\mathcal{S},c,l}(\mathbf{x}) = 1$ if $\mathbf{x}$ is visible from both images, or 0 otherwise. $\mathrm{d}s$ is the element of area of the surface. This problem formulation allows thus to naturally integrate both multiview geometry and normal constraint. The functional (4) constrains the surface normals to be on the orthogonal plane of $\mathbf{h}(\mathbf{x})$. This is an ill-posed problem since there is an unlimited choice for the normal. In this context several reciprocal pairs are needed in order to better pose the problem and Energy (4) has to be adapted to multiview settings. At the end, we then consider the energy to minimize as the sum of all energies for all Helmholtz pairs i:

$$E_{HS}(\mathcal{S}) = \sum_i \int_{\mathcal{S}} \left(\mathbf{h}_i(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x})\right)^2\ \nu_{\mathcal{S},c_i,l_i}(\mathbf{x})\ \mathrm{d}s\ , \tag{5}$$

where i is the ith camera / light pair.

In the next section, we show how to minimize this energy via gradient descent when the surface is represented by a triangle mesh. To simplify the notations, we will just consider and compute the gradient of the functional (4). The gradient of (5) is then obtained by summing the gradients of all the camera / light pairs.

3 Optimization for Triangle Mesh Representation

3.1 Choice of Representation

In section 1.2 we show the interest of using an intrinsic and full surface representation. In practice, there exists several possibilities for such representation. We propose to minimize energy (5) using a gradient descent algorithm. The choice of the representation must then be consistent with the surface evolution technique which is the base of gradient descent methods. Here we chose to use the Lagrangian framework and to represent the surface by a triangle mesh. Several reasons motivate this choice. In recent years, Lagrangian methods have taken advantage of significant advances in mesh processing allowing these methods to enjoy practical properties such as topological changes [18, 19]. In Lagrangian methods, the gradient is computed directly from the discrete representation, whereas in the Eulerian framework the continuous gradient is computed and then discretized. Performing gradient descent in the context of discrete representations allows to make the minimization coherent with the handled numerical object. In other respects, the visibility of a point from a vantage point is well defined and easy to compute with a mesh representation (by using graphic hardware). In practice, it is easier to check the visibility with such a representation than with a level-set representation. All these reasons make nowadays Lagrangian methods more and more popular. Also these methods have recently proved their strong potential for 3D applications [8, 20, 21].

3.2 Shape Gradient and Evolution Algorithm

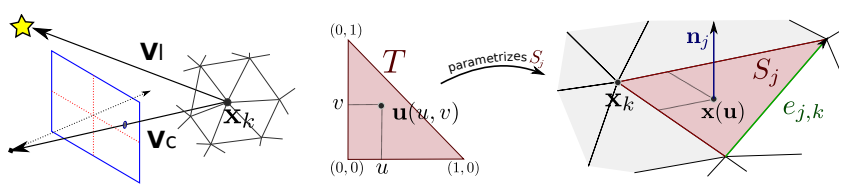

Fig. 2. Notations. Parametrization of the discrete representation of the surface into a triangle mesh.

From now, we consider that the surface is a piecewise planar triangular mesh. Let $\mathbf{X} = \{\mathbf{x}_1 \dots \mathbf{x}_n\}$ be a discrete mesh, $\mathbf{x}_k$ being the kth vertex of $\mathbf{X}$, and let $\mathcal{S}_j$ be the jth triangle of $\mathbf{X}$. With such a representation, functional (4) can be rewritten as:

$$E_{HS}(\mathcal{S}) = \sum_j \int_{\mathcal{S}_j} (\mathbf{h}(\mathbf{x}) \cdot \mathbf{n}_j)^2 \ \nu_{\mathcal{S},c,l}(\mathbf{x}) \, \mathrm{d}s_j \ , \tag{6}$$

where $\mathbf{n}_j$ is the normal to $\mathcal{S}_j$ and where the sum is over all the triangles of the mesh $\mathbf{X}$. Figure 2 illustrates these notations.

We propose to optimize E_{HS} with respect to $\mathcal{S}$ using the shape gradient [22]. Let $\mathbf{V}$ be a vector field defined on all the vertices $\mathbf{x}$ of the mesh $\mathbf{X}$ representing the surface deformation. Let us consider the evolution of E_{HS} according to the deformation $\mathbf{V}$. In other words, we assume that the vertices $\mathbf{x}_k[t]$ of $\mathbf{X}[t]$ are moving according to $\mathbf{x}_k[t] = \mathbf{x}_k^0 + t\mathbf{V}_k$. The method for computing the gradient of E_{HS} with respect to $\mathcal{S}$ consists in computing the directional derivative of $E(\mathcal{S})$ for this deformation $\mathbf{V}$, i.e., $\frac{d}{dt}E(\mathcal{S}[t])\Big|_{t=0}$, and then in rewriting it as a scalar product of $\mathbf{V}$, i.e. as $\langle \mathbf{V}, \mathbf{G}\rangle = \sum_k \mathbf{V}_k \cdot \mathbf{G}_k$. The obtained vector $\mathbf{G}$ is called the *gradient* and the energy necessarily decreases when deforming the surface according to its opposite direction $-G$. Indeed, for $\mathbf{x}_k[t] = \mathbf{x}_k^0 - t\mathbf{G}$, we have $\frac{d}{dt}E(\mathcal{S}[t])\Big|_{t=0} = -\langle \mathbf{G}, \mathbf{G}\rangle \leq 0$, see [20].

To obtain the gradient of our energy E_{HS}, we have then to calculate the expression of $\frac{d}{dt}E_{HS}(\mathcal{S}[t])\Big|_{t=0}$ and express it as a scalar product of $\mathbf{V}$. In appendix A, we have detailed this calculus in the general case where a functional $\int \mathcal{S} g(\mathbf{x}, \mathbf{n}(\mathbf{x}))\mathrm{d}s$ is minimized. By replacing $g(\mathbf{x}, \mathbf{n}(\mathbf{x}))$ by $(\mathbf{h}_i(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}))^2$, we get:

$$\frac{d}{dt}E(\mathcal{S}[t])\Big|_{t=0} = \sum_j \sum_{k\in K_j} \mathbf{V}_k \cdot \left\{ \frac{\mathbf{e}_{j,k}}{A_j} \wedge \int_{\mathcal{S}_j} \Big((\mathbf{h}\cdot\mathbf{n}_j)^2\, \mathbf{n}_j - 2\,(\mathbf{h}\cdot\mathbf{n}_j)\,\mathbf{h} \Big)\, \mathrm{d}s_j \right.$$
$$\left. - \int_{\mathcal{S}_j} 2(\mathbf{h}\cdot\mathbf{n}_j)\nabla_{\mathbf{x}}(\mathbf{h}\cdot\mathbf{n}_j)\, \phi_k(\mathbf{x})\, \mathrm{d}s_j \right\}, \quad (7)$$

where A_j is the area of $\mathcal{S}_j$, $\mathbf{e}_{j,k}$ is the edge of $\mathcal{S}_j$ that is at the opposite of vertex k; K_j is the set of the indexes of the three vertices of the triangle $\mathcal{S}_j$ and $\phi_k : \mathcal{S} \to \mathbb{R}$ is the piecewise linear interpolating basis function such that $\phi_k(\mathbf{x}_k) = 1$ and $\phi_k(\mathbf{x}_i) = 0$ if $i \neq k$.
Then the L^2 gradient descent flow using the triangular mesh uses the gradient $\nabla E(\mathbf{X}) = M^{-1}\dfrac{\partial E}{\partial \mathbf{X}}(\mathbf{X})$, where M is the *mass* matrix and $\dfrac{\partial E}{\partial \mathbf{X}}(\mathbf{X})$ is directly given by the part in braces of Equation (7). One classically approximates M by the diagonal mass lumping $\tilde{M}$, where $\tilde{M}_{ii}$ is the area of the Voronoi dual cell of $\mathbf{x}_i$ times the identity matrix Id_3; for more details, see e.g. [20, 23]. The evolution algorithm used here is:

$$\begin{cases} \mathbf{X}[0] = \mathbf{X}^0\,, \\ \mathbf{X}[t+1] = \mathbf{X}[t] - dt\; \tilde{M}^{-1}\dfrac{\partial E}{\partial \mathbf{X}}(\mathbf{X}[t])\,, \end{cases} \quad (8)$$

where $\mathbf{X}^0$ is some initial mesh.

4 Experimental Results

In this section we present results using the gradient developed in section 3 (Equation (7)) that directly corresponds to the minimization of our original Energy (Equation (6)).

Let us emphasize here that the gradient descent flow used to obtain the result is exactly the one described previously. In particular, it does not require additional terms or parameters such as surface smoothing present in most variational formulations. Of course, adding such a term would help in being more robust to noise and calibration errors. The only parameters used by us are for numerical computations, like for instance the numerical integration over the triangles. All those can be easily estimated automatically. Since this is a gradient descent approach, one needs some reasonable initialization of the surface such as the visual hull to avoid local minima.

The experiments were implemented in C++ and OpenGL using the CGAL library for mesh computation running on a standard 2.4GHz linux machine – and the topology adaptive meshes of [18]. The optimization starts from an initial condition which is the visual hull in our case. A coarse-to-fine approach is applied to help prevent from local minima. The rendered results use only one constant normal per facet (flat shading).

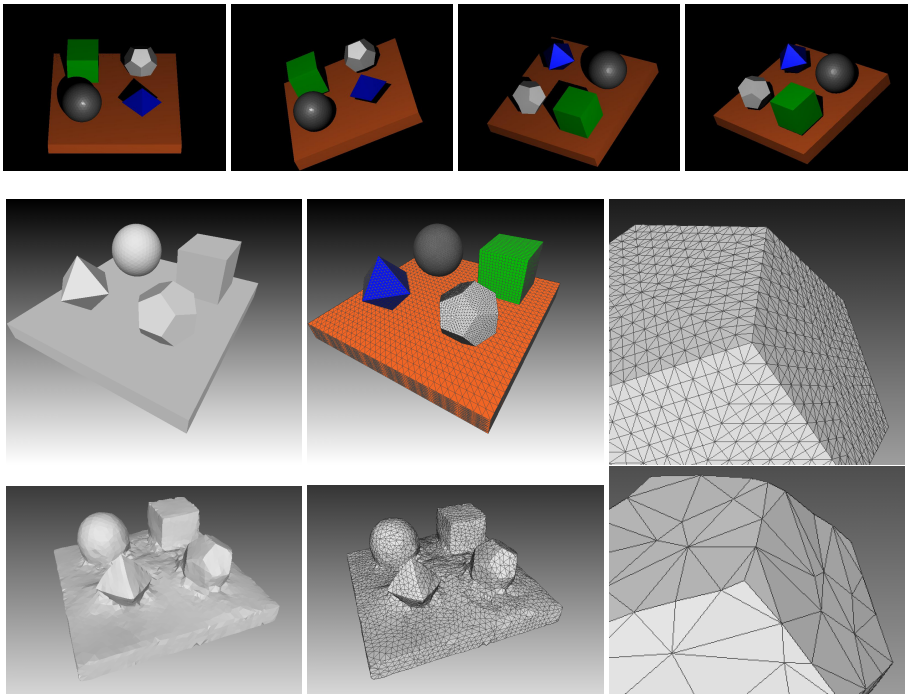

Fig. 3. A simple synthetic example. Top row: 2 camera / light pairs out of the 32. Middle row: ground truth surface; ground truth mesh representation; Bottom row: final result; mesh representation of the result; details on the mesh.

We first apply our method to synthesized data. Figure 3 show an example of simple objects disposed on a plane, where images were generated using non-Lambertian reflectance. This dataset is composed of 32 reciprocal pairs with images of resolution 800×600, placed all around the object of interest. This example shows that our method

is able to recover the surface whereas previous Helmholtz stereo methods, where visibility is not accounted for, would fail. In order to solve this problem, they would need to cluster the camera positions to find several view-centered cameras, integrate multiple normal maps and then merge the final reconstructions into a single surface. Our method is simpler in the sense that it works without additional steps.

Even though the number of vertices is low, the gradient flows tends to place them in their correct location. In particular triangle edges perfectly match the one in the images. Again, this is made possible because the discrete gradient is computed with respect to the discrete representation. Also the approach is suitable for reconstructing objects with sharp edges, having depth discontinuities or self-occlusions.

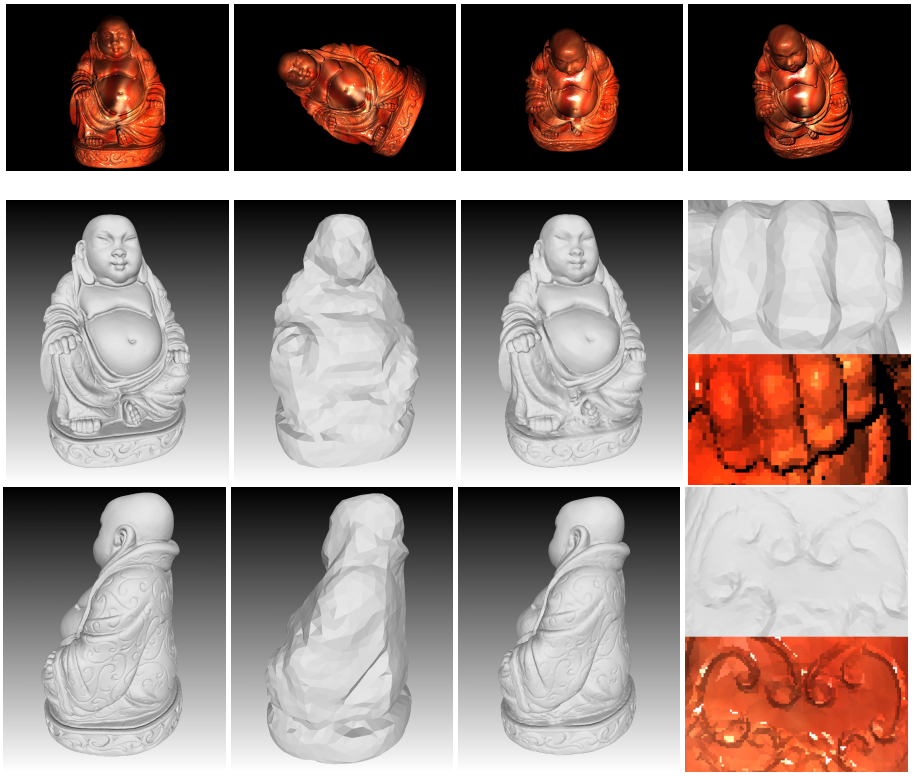

Fig. 4. Buddha dataset. Top row: 2 camera – light pairs out of the 32. Bottom row: ground truth surface; initial visual hull; estimated mesh; input image zoom and corresponding recovered mesh details.

The example in figure 4 shows that our method can be applied to reconstruct full and high quality object shapes. Even though the input resolution of the images is low (1024×768), the recovered surface nicely matches the ground truth model. The images were generated using a mixture of different specular models so that it looks realistic and

non Lambertian. Details are well recovered, and the quality of the mesh is good enough for further object relighting.

Following the evaluation presented in [24], we perform a quantitative evaluation of examples from Figures 3 and 4 where we consider the object in a 2m diameter bounding box. For Figure 3, the completeness at 10mm is 87.51%, and the accuracy at 95% is 9.301mm. For the Buddha, the completeness at 10mm is 95.766%, and the accuracy at 95% is 5.29mm.

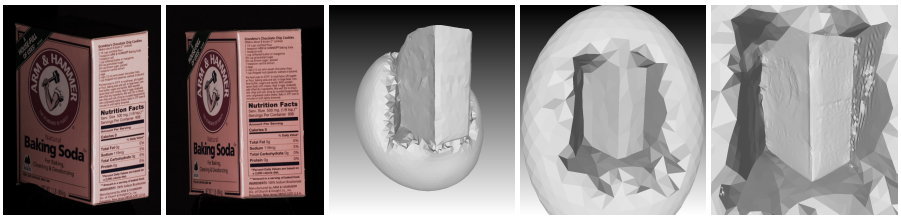

Fig. 5. Small box data: results of the proposed method with different initial conditions and mesh resolution. From left to right: 1 camera – light pair out of the 8 pairs; result from a small non-encompassing initial surface; result from an encompassing initial surface; result from an encompassing initial surface with a more dense resolution.

The next two figures 5 and 6 show real data used in previous work [13, 14], that have been capture only from one side. We then cannot reconstruct the full surface of the object since images behind the object are missing. Figure 5 shows the 3D reconstruction of a textured box. Since we can choose the mesh resolution, having large triangles compared to the image resolution allows to integrate the gradient over the triangle and have a correct gradient flow for the vertices of the triangle. Then we can reconstruct objects with textured or rough surfaces using Helmholtz reciprocity similarly as in [15]. This figure also show different optimization taken from different initial conditions that finally give similar results.

Figure 6 show results for two real datasets, the mask containing 18 reciprocal pairs, and the mannequin containing 8 reciprocal pairs.

Finally, the approach was tested on real dataset on full 3D objects (Figure 7). It consists of 18 reciprocal pairs (using 1104×828 resolution images) taken on a ring around the object slightly on top of it. The two data show a "Fish" reconstruction highly specular with fine changing surface structure, whereas the second one "Dragon" has strong self-occlusions and complex shape. Starting from the visual hull, we are able to recover details on the surface even though the input image resolution is not too high. Results are illustrated in Figure 7 and present the recovered full 3D surface. Some parts of the surface are not visible from the images and thus cannot be recovered. Camera calibration was performed using a checker board without distortion corrections and light positions were empirically positioned and calibrated. These datasets will be made available for comparisons purposes. Images are taken around the object so some parts are occluded from images and objects also contain self-occlusions. For those reasons,

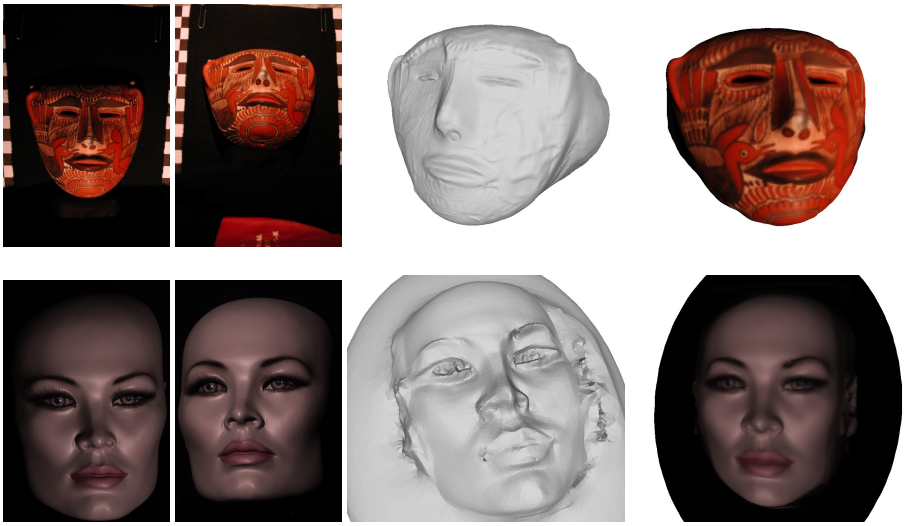

Fig. 6. Mannequin (8 reciprocal pairs) and Mask (18 reciprocal pairs) data: results of the proposed method for two real dataset containing varying complex appearance. From left to right: 1 camera / light pair out of the 18 (respectively 8); final result; final mesh textured with the mean of the reprojected colors from the cameras.

previous approaches using a camera-center view do not apply – since it requires depth continuity – whereas our surface-based approach can recover the full 3D shape.

4.1 Discussion

All these examples illustrated the advantage of having a mesh representation. It allows to preserve edges, and we show that having a coherent discrete gradient flow allows vertices to be placed at their correct locations. Possible extensions to the approach may use automatic adaptive meshes, where big triangles would fill planar surface parts and high curvature would have a more dense mesh.

Second order minimization is known to recover well the higher frequency of the surface, but poorly recover the lower ones. Also, due to the integration process, the optimization might be slow if the surface is too far from the solution. A good initial condition and a coarse to fine approach significantly help to prevent these problems, as illustrated in the experiments. In particular, the gradient flow tends to shrink the surface and introduce a minimal surface bias.

However, since the method minimizes a weighted area functional defined over the surface, one of its global minima (in addition to the real surface) is the empty set. To prevent from this choice, one can add an additional term, start closer to the solution, or fix boundary conditions. A more elegant way is to see the problem as a reprojection error. Instead of minimizing a weighted area functional, one can reformulate the problem

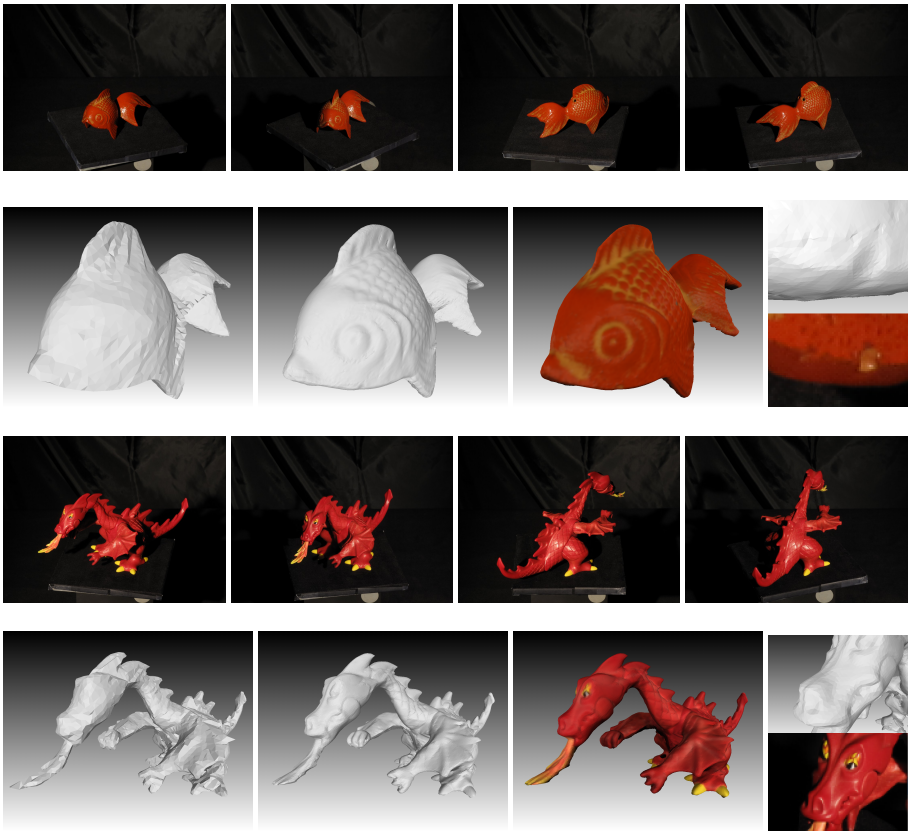

Fig. 7. Fish and Dragon data (18 reciprocal pairs of 1104×828): Results of the proposed method for two full 3D real world datasets. Top and third row: 2 camera / light pairs out of the 36. Second and last row: initial visual hull; recovered 3D mesh; final mesh textured with the mean of the reprojected colors from the cameras; input image zoom and corresponding recovered mesh details.

(2) by minimizing the following energy functional:

$$E(\mathcal{S}) = \int_{\mathcal{I}_c} \left(I_c - \left(I_l \circ \pi_{\mathcal{S},I_l} \circ \pi^{-1}_{\mathcal{S},I_c} \right) \frac{\hat{\mathbf{v}}_l \cdot \left(\mathbf{n} \circ \pi^{-1}_{\mathcal{S},I_c} \right)}{\hat{\mathbf{v}}_c \cdot \left(\mathbf{n} \circ \pi^{-1}_{\mathcal{S},I_c} \right)} \right)^2 \mathrm{d}\mathbf{u} \,, \tag{9}$$

where $\pi^{-1}_{\mathcal{S},I_c}(\mathbf{u})$ is the reprojection of an image point $\mathbf{u}$ of the image I_c on the surface $\mathcal{S}$. This formulation as been presented by [17], but has not been minimized as a reprojection error, like for instance in the case of [25] for continuous surfaces, and of [21] using deformable meshes. In fact one may rewrite Equation (9) as an energy over the visible surface [25] instead of the image in order to optimize the surface. Such formulation lets appear an additional term, that turns out to behave like a visual hull constrain on the

silhouette occluding contours, or as a contour matching term for self occlusions. Such a term will give boundary conditions to prevent from shrinkage, and will help the method to be more robust to initial conditions.

5 Conclusion

In this paper we have presented a surface-based method to estimate the 3D shape of objects from multiview Helmholtz stereo pairs. As far as we know, this is the first time Helmholtz stereopsis can be used to recover dense and full 3D models into a single framework. This is made possible thanks to the compact surface representation that allow to easily compute surface point visibility. Moreover, the mesh based representation allows to naturally exploit geometric relationships between a point of the scene and its surface normal. Tests on synthetic and real datasets demonstrate the benefit of our approach.

Acknowledgement. Research was supported by the Agence Nationale pour la Recherche within the Flamenco project (Grant ANR-06-MDCA-007). The authors would like to thank Todd Zickler for providing Helmholtz stereo data used in Figures 5 and 6.

References

1. Goesele, M., Curless, B., Seitz, S.M.: Multi-view stereo revisited. In: IEEE Conference on Computer Vision and Pattern Recognition. Volume 02. (2006) 2402–2409
2. Jin, H., Cremers, D., Yezzi, A.J., Soatto, S.: Shedding light on stereoscopic segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition. Volume 01. (2004) 36–42
3. Jin, H., Cremers, D., Wang, D., Prados, E., Yezzi, A., Soatto, S.: 3-d reconstruction of shaded objects from multiple images under unknown illumination. International Journal of Computer Vision **76** (2008)
4. Hernandez, C., Vogiatzis, G., Cipolla, R.: Multiview photometric stereo. IEEE Transactions on Pattern Analysis and Machine Intelligence **30** (2008) 548–554
5. Mallick, S.P., Zickler, T., Kriegman, D.J., Belhumeur, P.N.: Beyond lambert: Reconstructing specular surfaces using color. In: IEEE Conference on Computer Vision and Pattern Recognition. Volume 2. (2005) 619–626
6. Zickler, T., Mallick, S.P., Kriegman, D.J., Belhumeur, P.: Color subspaces as photometric invariants. International Journal of Computer Vision (2008)
7. Pons, J.P., Keriven, R., Faugeras, O.: Multi-view stereo reconstruction and scene flow estimation with a global image-based matching score. The International Journal of Computer Vision **72** (2007) 179–193
8. Vu, H., Keriven, R., Labatut, P., Pons, J.P.: Towards high-resolution large-scale multi-view stereo. In: IEEE Conference on Computer Vision and Pattern Recognition. (2009)
9. Jin, H., Soatto, S., Yezzi, A.J.: Multi-view stereo beyond lambert. Computer Vision and Pattern Recognition, IEEE Computer Society Conference on **1** (2003) 171
10. Yu, T., Xu, N., Ahuja, N.: Recovering shape and reflectance model of non-lambertian objects from multiple views. In: IEEE Conference on Computer Vision and Pattern Recognition. (2004) 226–233
11. Yoon, K.J., Prados, E., Sturm, P.: Joint estimation of shape and reflectance using multiple images with known illumination conditions. International Journal of Computer Vision (2009) to appear.

12. Hertzmann, A., Seitz, S.: Example-based photometric stereo: Shape reconstruction with general, varying brdfs. IEEE Trans. Pattern Anal. Mach. Intell. **27** (2005) 1254–1264
13. Zickler, T., Belhumeur, P.N., Kriegman, D.J.: Helmholtz stereopsis: Exploiting reciprocity for surface reconstruction. International Journal of Computer Vision **49** (2002) 215–227
14. Zickler, T., Ho, J., Kriegman, D., Ponce, J., Belhumeur, P.: Binocular helmholtz stereopsis. In: Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on. (2003) 1411–1417 vol.2
15. Guillemaut, J.Y., Drbohlav, O., Sara, R., Illingworth, J.: Helmholtz stereopsis on rough and strongly textured surfaces. In: 3DPVT '04: Proceedings of the 3D Data Processing, Visualization, and Transmission, 2nd International Symposium, Washington, DC, USA, IEEE Computer Society (2004) 10–17
16. Zickler, T.: Reciprocal image features for uncalibrated helmholtz stereopsis. In: Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on. Volume 2. (2006) 1801–1808
17. Tu, P., Mendon, P.R.S.: Surface reconstruction via helmholtz reciprocity with a single image pair. Computer Vision and Pattern Recognition, IEEE Computer Society Conference on **1** (2003) 541
18. Pons, J.P., Boissonnat, J.D.: Delaunay deformable models: Topology-adaptive meshes based on the restricted delaunay triangulation. In: IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, USA (2007)
19. Zaharescu, A., Boyer, E., Horaud, R.: Transformesh: a topology-adaptive mesh-based approach to surface evolution. In: Proceedings Asian Conference on Computer Vision, Tokyo, Japan (2007)
20. Eckstein, I., Pons, J.P., Tong, Y., Kuo, C.C.J., Desbrun, M.: Generalized surface flows for mesh processing. In: Eurographics Symposium on Geometry Processing. (2007)
21. Delaunoy, A., Prados, E., Gargallo, P., Pons, J.P., Sturm, P.: Minimizing the multi-view stereo reprojection error for triangular surface meshes. In: British Machine and Vision Conference, Leeds, UK. (2008)
22. Debreuve, ., Gastaud, M., Barlaud, M., Aubert, G.: Using the shape gradient for active contour segmentation: from the continuous to the discrete formulation. Journal of Mathematical Imaging and Vision (2007)
23. Meyer, M., Desbrun, M., Schröder, P., Barr, A.H.: Discrete differential-geometry operators for triangulated 2-manifolds. (2002)
24. Seitz, S.M., Curless, B., Diebel, J., Scharstein, D., Szeliski, R.: A comparison and evaluation of multi-view stereo reconstruction algorithms. In: IEEE Conference on Computer Vision and Pattern Recognition. (2006) 519–528
25. Gargallo, P.: Contributions to the Bayesian approach to Multi-view Stereo. PhD thesis, Institut National Polytechique de Grenoble, France (2008)

A Gradient Flows of Weighted Area Functionals using Triangular Meshes

Let $\mathcal{S}_j$ be the jth triangle of the mesh and let us consider a parametrization of the triangle $\mathcal{S}_j$ such that $\mathbf{x}(\mathbf{u}) = x_k^j + u\,\overrightarrow{x_k^j x_1^j} + v\,\overrightarrow{x_k^j x_2^j}$ where x_k^j and x_1^j and x_2^j are the three vertices associated to the triangle $\mathcal{S}_j$ and where $\mathbf{u}(u,v) \in T = \{(u,v) | u \in [0,1] \text{ and } v \in [0,u]\}$. (See Figure 2)
In this appendix, we detail the calculus of $\frac{d}{dt} E(\mathcal{S}_j[t])\Big|_{t=0}$ for energy functionals of the form:

$$E(\mathcal{S}) = \sum_i \int_{\mathcal{S}_j} g(\mathbf{x}, \mathbf{n}(\mathbf{x}))\, d\sigma = \sum_j 2\, A_j \int_T g(\mathbf{x}(\mathbf{u}), \mathbf{n}_j)\, \mathrm{d}\mathbf{u}\,, \tag{10}$$

where A_j is then the area of the triangle $\mathcal{S}_j$ and $\mathbf{n}_j$ the outward surface normal of triangle $\mathcal{S}_j$. Let us focus on the evolution of $E(S)$ under the induced velocity $\mathbf{V}$ on triangle $\mathcal{S}_j$ only (In other words when a vertex $\mathbf{x}_k$ is moved according to $\mathbf{x}_k[t] = \mathbf{x}_k^0 + t\mathbf{V}_k$). We have:

$$E(\mathcal{S}_j[t]) = 2\,A_j[t] \int_T g(\mathbf{x}(\mathbf{u}) + t\mathbf{V}(\mathbf{x}(\mathbf{u})), \mathbf{n}_j[t])\,\mathrm{d}\mathbf{u}\,, \tag{11}$$

where $A_j[t]$ is the area of the triangle $\mathcal{S}_j[t]$, $\mathbf{n}_j[t]$ is its normal and $\mathbf{V}(\mathbf{x})$ is the piecewise linear extension of the $\mathbf{V}$ vector field on the whole surface $\mathcal{S}$. By simple derivation we get:

$$\begin{aligned}\frac{d}{dt}E(\mathcal{S}_j[t])\Big|_{t=0} = \frac{A_j'[0]}{A_j} \int_{\mathcal{S}_j} g(\mathbf{x}, \mathbf{n}_j)\,\mathrm{d}s_j \\ + \int_{\mathcal{S}_j} \nabla_{\mathbf{x}} g(\mathbf{x}, \mathbf{n}_j)\cdot \mathbf{V}(\mathbf{x})\,\mathrm{d}s_j \;+ \int_{\mathcal{S}_j} \nabla_{\mathbf{n}} g(\mathbf{x}, \mathbf{n}_j)\cdot \mathbf{n}_j'[0]\,\mathrm{d}s_j\,,\end{aligned} \tag{12}$$

where $A_j'[0] = \frac{d}{dt}A_j[t]\Big|_{t=0}$ and $\mathbf{n}_j'[0] = \frac{d}{dt}\mathbf{n}_j[t]\Big|_{t=0}$ are detailed below. In order to rewrite $\frac{d}{dt}E(\mathcal{S}_j[t])\Big|_{t=0}$ as a scalar production of $\mathbf{V}$, we then need to detail $A_j'[0]$ and $\mathbf{n}_j'[0]$. Since for all $\mathbf{x} \in \mathcal{S}_j$, $\mathbf{V}(\mathbf{x}) = \sum_{k\in K_j} \mathbf{V}_k \phi_k(\mathbf{x})$ (K_j, ϕ_k and $\mathbf{e}_{j,k}$ being defined in section 3.2), one can show that: $A_j'[0] = \sum_{k\in K_j} \frac{1}{2}\,(\mathbf{n}_j \wedge \mathbf{e}_{j,k})\cdot \mathbf{V}_k$, and

$$\mathbf{n}_j'[0] = \frac{1}{2\,A_j}\left(\left(\sum_{k\in K_j} \mathbf{e}_{j,k}\wedge \mathbf{V}_k\right) - \left(\left(\sum_{k\in K_j} \mathbf{e}_{j,k}\wedge \mathbf{V}_k\right)\cdot \mathbf{n}_j\right)\mathbf{n}_j\right)\,. \tag{13}$$

It follows that

$$\begin{aligned}\frac{d}{dt}E(\mathcal{S}_j[t])\Big|_{t=0} = \sum_{k\in K_j} \mathbf{V}_k\cdot \Bigg\{ \frac{\mathbf{n}_j\wedge \mathbf{e}_{j,k}}{2A_j} \int_{\mathcal{S}_j} g(\mathbf{x}, \mathbf{n}_j)\,\mathrm{d}s_j \\ + \int_{\mathcal{S}_j} \nabla_{\mathbf{x}} g(\mathbf{x}, \mathbf{n}_j)\phi_k(\mathbf{x})\,\mathrm{d}s_j - \frac{\mathbf{e}_{j,k}}{2\,A_j}\wedge \int_{\mathcal{S}_j} g_{\mathbf{n}}(\mathbf{x}, \mathbf{n}_j)\,\mathrm{d}s_j \Bigg\}\,.\end{aligned} \tag{14}$$

Above, we have denoted $g_{\mathbf{n}} = \nabla_{\mathbf{n}} g(\mathbf{x}, \mathbf{n}_j) - \langle \nabla_{\mathbf{n}} g(\mathbf{x}, \mathbf{n}_j), \mathbf{n}_j\rangle\ \mathbf{n}_j$, where $\nabla_{\mathbf{n}} g(\mathbf{x}, \mathbf{n}_j)$ is the gradient of g with respect to the second variable (i.e. $\mathbf{n} \in \mathbb{R}^3$).

It then immediately follows that:

$$\begin{aligned}\frac{d}{dt}E(\mathcal{S}[t])\Big|_{t=0} = \sum_k \mathbf{V}_k \cdot \sum_{j\in \mathcal{J}_k} \Bigg\{ \int_{\mathcal{S}_j} \nabla_{\mathbf{x}} g(\mathbf{x}, \mathbf{n}_j)\phi_k(\mathbf{x})\,\mathrm{d}s_j \\ - \frac{\mathbf{e}_{j,k}}{2A_j}\ \wedge\ \int_{\mathcal{S}_j} g(\mathbf{x}, \mathbf{n}_j)\mathbf{n}_j + g_{\mathbf{n}}(\mathbf{x}, \mathbf{n}_j)\,\mathrm{d}s_j \Bigg\}\,,\end{aligned} \tag{15}$$

where $\mathcal{J}_k$ is the set of triangles containing vertex $\mathbf{x}_k$. By replacing $g(\mathbf{x}, \mathbf{n}(\mathbf{x}))$ by $(\mathbf{h}_i(\mathbf{x})\cdot \mathbf{n}(\mathbf{x}))^2$ in the equations below, where $\mathbf{h}_i(\mathbf{x})$ is the vector defined in the paper, we have:

$$\begin{aligned}\nabla_{\mathbf{x}} g(\mathbf{x}, \mathbf{n}_j) &= 2(\mathbf{h}(\mathbf{x})\cdot \mathbf{n}_j)\nabla_{\mathbf{x}}(\mathbf{h}(\mathbf{x})\cdot \mathbf{n}_j)\,, \\ g(\mathbf{x}, \mathbf{n}_j)\mathbf{n}_j &= (\mathbf{h}(\mathbf{x})\cdot \mathbf{n}_j)^2\,\mathbf{n}_j\,, \\ \text{and}\ \ g_{\mathbf{n}}(\mathbf{x}, \mathbf{n}_j) &= 2\,(\mathbf{h}(\mathbf{x})\cdot \mathbf{n}_j)\,\mathbf{h}(\mathbf{x}) - 2\,(\mathbf{h}(\mathbf{x})\cdot \mathbf{n}_j)^2\,\mathbf{n}_j\,.\end{aligned} \tag{16}$$