Multi-View Normal Estimation - Application to Slanted Plane-sweeping

Lilian Calvet, Nicolas Maignan, Baptiste Brument, Jean Mélou, Silvia Tozza, Jean-Denis Durou, Yvain Quéau

Multi-View Normal Estimation – Application to Slanted Plane-sweeping

Lilian Calvet[1,3]✉*, Nicolas Maignan[2*], Baptiste Brument[3], Jean Mélou[3], Silvia Tozza[4], Jean-Denis Durou[3], and Yvain Quéau[5]

[1] Research in Orthopedic Computer Science, Balgrist University Hospital, University of Zurich, 8008 Zurich, Switzerland

[2] Université de Lorraine CNRS Inria LORIA, 54000 Nancy, France

[3] IRIT, UMR CNRS 5505, 31000 Toulouse, France

[4] Department of Mathematics, Università di Bologna, 40126 Bologna, Italy

[5] Normandie Univ, UNICAEN, ENSICAEN, CNRS, GREYC, 14000 Caen, France

✉lilian.calvet@balgrist.ch

Abstract. In this paper, we show how to estimate the normals of a 3D surface from a minimum of two views, assuming that the poses of a calibrated camera are perfectly known. For each pair of image points, the normal at the corresponding 3D point is expressed in function of the local gradients of the grey level, whatever the type of image formation (orthogonal or perspective projection). As an application, this allows us to fully estimate the inter-image homography, which not only depends on the relative pose between views, but also on the local orientation of the surface. Hence, the photo-consistency between patches from two images, which is the basis of the so-called "plane-sweeping" method, is improved. Experiments on synthetic and real data validate our approach.

Keywords: Normal Estimation · Multi-view Stereo · Plane-sweeping.

1 Introduction

Image-based 3D reconstruction pipelines usually comprise three steps: 1) feature extraction and matching across the images; 2) structure-from-motion to estimate the camera poses and a sparse 3D point cloud; 3) multi-view stereo (MVS), which reconstructs a dense 3D geometry. A common approach to MVS is to match the pixels between the different views by maximizing the photo-consistency of a specific image, called *reference image*, with the others, called *control images*. To measure photo-consistency, the reference image is warped to the control images, assuming known poses and making a guess on the depth.

Assuming the surface is locally flat, the plane-sweeping method allows for a more robust comparison than pixel-to-pixel, as it allows for patch comparison. The change of point of view implies a distortion of a patch from one image to another, which takes the form of a homography depending on the normal to the tangent plane of the surface. However, this dependency is usually ignored, due to the difficulty of estimating this normal, which is considered to be collinear with the optical axis of the reference camera (see Figure 1-a).

*Lilian Calvet and Nicolas Maignan contributed equally to this work.

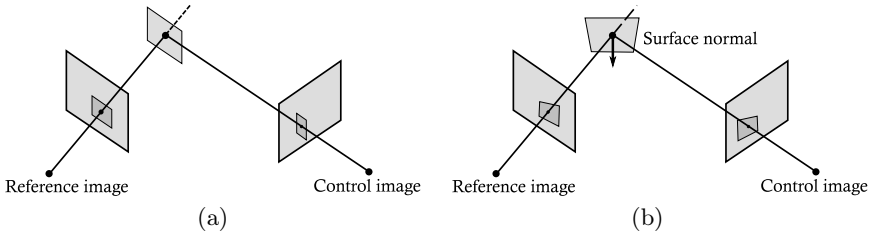


Fig. 1. (a) Standard plane-sweeping assumes that the scene is locally fronto-parallel to the reference image. (b) We propose to estimate the surface normal from image gradients and integrate this knowledge into a slanted plane-sweeping method.

In this paper, we present a new method for estimating the normal to a surface from two images, using the gradients of the grey levels in a pair of conjugate image points, which allows us to characterise the inter-image homography, and thus to improve the comparison between conjugate patches (see Figure 1-b). After a brief review of related approaches in Section 2, we present our method for normal estimation in Section 3. Preliminary experiments on both synthetic and real data are conducted in Section 4, which present the strengths and weaknesses of our approach. Our work is eventually summarised in Section 5.

2 Related Work

MVS methods can be divided in three main approaches. The first one exploits inter-image correspondences for multi-view 3D reconstruction [1, 8, 9, 25]. These methods typically estimate depth maps, fuse them into point clouds and optionally generate meshes [16]. The second one uses implicit representations and leverages differentiable rendering to reconstruct 3D geometry with appearance from image collections [15, 22, 24]. NeRF [22] and most of its follow-ups use volumetric representations and compute radiance by ray marching through a neurally encoded 5D light field. The third approach uses explicit surface representations and estimates an explicit 3D mesh from images [5, 6, 17, 19, 21, 26]. Most methods based on this approach assume a given, fixed mesh topology [5, 6, 21], but this assumption has been relaxed recently [17, 23, 26].

In this work, we focus on the method based on the first approach using inter-image correspondences. These methods commonly assume a fronto-parallel scene structure (see Figure 1-a). Gallup et al. [10] observed the distortion of the cost function caused by structures that deviate from this prior and combated it by using multiple sweeping directions deduced from the sparse reconstruction. Earlier approaches [3, 4, 28] similarly account for the surface normal in stereo matching. Bleyer et al. [27] use PatchMatch to estimate per-pixel normals to compensate for the distortion of the cost function. They initialize each pixel with a random plane, hoping that at least one pixel of the region, supposedly locally planar, carries a plane that is close to the correct one. The method has no guarantee of converging to the correct normal estimate and, in practice, its

success depends on the size of the regions that can be approximately modeled by the same plane. In addition, the method uses spatial and temporal propagations from "good" normal guesses, which is not desirable when aiming at reconstructing fine objects' details and working without video sequence. Schönberger et al. [25] follow the same approach while considering a variety of photometric and geometric priors. More recently, Liu et al. [20] proposed to automatically detect piecewise planar regions in the input images, and to compute the associated planes' parameters. They use a slanted plane-sweeping strategy. A set of three-dimensional slanted plane hypotheses is generated over both normal and depth values following uniform distributions of learned ranges.

In contrast to these approaches, we propose to estimate pixel-level normals from image gradients given camera parameters. We integrate this knowledge into a slanted plane-sweeping strategy (see Figure 1-b) in order to overcome distortions induced by deviation of the surface to the fronto-parallel scene structure assumption. The problem of surface normal estimation from image gradients was also tackled by Lindeberg et al. in [11, 18]. Therein, it is shown that the surface normal can be obtained from the transformation that relates the second moment matrices (computed from image gradients), in a scale-space framework. However, integrating this approach in a plane-sweeping strategy would require either solving a difficult two-parameters optimization problem, or resorting to a keypoint-based procedure which is not suitable when texture is lacking. On the contrary, the proposed approach requires solving a simpler one-parameter optimization problem, and it is suitable for poorly textured surfaces.

3 Slanted Plane-sweeping

3.1 Photo-consistency-based MVS

Let us first recall the principle of MVS, which estimates the depth map associated with the reference camera by browsing, for each pixel, a set of possible depths.

Let us consider an opaque surface observed by $n+1$ identical cameras providing a reference image I and n control images I_i, $i \in \{1, \dots, n\}$. The poses of these cameras are assumed to be known and expressed in a world frame aligned with the reference camera frame. Cameras intrinsics are also supposed known. Let $\mathbf{Q} = [x, y, z]^\top$ be a 3D point expressed in the reference camera frame. Since the camera parameters are known, we can note $\mathbf{q} = \pi(\mathbf{Q})$ the projection of $\mathbf{Q}$ in the reference image, whose coordinates $\mathbf{q} = [u, v]^\top$ are expressed in the reference image frame. We define the same way $\{\pi_i\}_{i \in \{1,\dots,n\}}$, the projections from 3D points to pixels in the control cameras.

The central projection π is invertible if the depth function z is known. In this case, there is a bijection between the visible 3D points of the scene and their images, which is written $\pi_z^{-1}(u, v)$, where the subscript z is used to indicate that, without knowledge of the function z, this writing would be ambiguous. Then, for a 3D point $\mathbf{Q}$ on the surface which is visible from all cameras, the Lambertian assumption gives:

$$I_i \circ \pi_i \circ \pi_z^{-1}(u, v) = I(u, v), \qquad i \in \{1, \dots, n\} \tag{1}$$

MVS searches for the depth function z corresponding to the reference image that maximises its **photo-consistency** with the n control images. Equations (1) are turned into a least squares problem, which has to be reformulated in discrete form (we do not know the grey level functions, but only their values in each pixel). The problem can then be solved separately in each pixel $\mathbf{q} = [u, v]^\top$:

$$\widehat{z}(u,v) = \underset{z \in \mathbb{R}}{\operatorname{argmin}} \ \frac{1}{n} \sum_{i=1}^{n} \left[I_i \circ \pi_i \circ \pi_z^{-1}(u,v) - I(u,v) \right]^2 \tag{2}$$

where $I_i \circ \pi_i \circ \pi_z^{-1}(u,v)$ has to be computed by interpolation.

For now, photo-consistency is reduced to the least squares comparison of two grey levels. In practice, photo-consistency is more complex [7, chapter 2]. The problem may then become nonlinear, non-smooth and non-convex, and the optimization tedious. Therefore, minimization is usually carried out using brute-force grid-search over the sampled depth space. This "winner-takes-all" strategy was first advocated in [14]. Despite its simplicity, it is remarkably efficient, and impressive depth map reconstructions of highly textured scenes have long been demonstrated [12].

3.2 Plane-sweeping

In practice, comparing pixel signals over a single pixel value, as shown in Equation (2), works very poorly due to the lack of information. To overcome this limitation, the photo-consistency is minimised over an image patch, namely over $\mathbf{v}(u,v)$ representing the pixel intensities in the vicinity of a pixel $\mathbf{q} = [u, v]^\top$. Considering the i-th control camera, $i \in \{1, \ldots, n\}$, the photo-consistency then measures the difference between vectors $I_i \circ \mathbf{v}_i \circ \pi_i \circ \pi_z^{-1}(u,v)$ and $I \circ \mathbf{v}(u,v)$, in the sense of a function ρ. Problem (2) then becomes:

$$\widehat{z}(u,v) = \underset{z \in \mathbb{R}}{\operatorname{argmin}} \ \frac{1}{n} \sum_{i=1}^{n} \rho \left(I_i \circ \mathbf{v}_i \circ \pi_i \circ \pi_z^{-1}(u,v), I \circ \mathbf{v}(u,v) \right) \tag{3}$$

As far as the reference camera is concerned, it seems natural to use the pixel grid to define a neighbourhood, and thus the $\mathbf{v}$ function. Now, we need to define the patch used in the control images. To do this, we are interested in the homography transforming $\mathbf{v}(u,v)$ into $\mathbf{v}_i \circ \pi_i \circ \pi_z^{-1}(u,v)$.

To go further, we need to explicit the coordinates $[u_i, v_i, 1]^\top$ of the projection on image plane i of a 3D point $\mathbf{Q} = [x, y, z]^\top$. Since the cameras are supposed to be identical, the calibration matrix $\mathbf{K}$ is independent of index i. The projection formula gives us:

$$[u_i, v_i, 1]^\top = \frac{1}{z_i} \mathbf{K} \left(\mathbf{R}_{0 \to i} \, \mathbf{Q} + \mathbf{t}_{0 \to i} \right) \tag{4}$$

where the rotation matrix $\mathbf{R}_{0 \to i}$ and the translation vector $\mathbf{t}_{0 \to i}$ charaterize the pose of camera i. Denoting $\mathbf{C}_i = -\mathbf{R}_{0 \to i}^{-1} \, \mathbf{t}_{0 \to i}$ the location of the optical center of camera i, (4) becomes:

$$[u_i, v_i, 1]^\top = \frac{1}{z_i} \mathbf{K} \, \mathbf{R}_{0 \to i} \left(\mathbf{Q} - \mathbf{C}_i \right) \tag{5}$$

which provides us with the following expression for the coordinates of the 3D point $\mathbf{Q}$:

$$\mathbf{Q} = z_i \, \mathbf{R}_{0\to i}^{-1} \, \mathbf{K}^{-1} [u_i, v_i, 1]^\top + \mathbf{C}_i \tag{6}$$

Putting (5) and (6) together, the movement of a point from camera i to camera j can be written:

$$[u_j, v_j, 1]^\top = \frac{1}{z_j} \, \mathbf{K} \, \mathbf{R}_{0\to j} \left(z_i \, \mathbf{R}_{0\to i}^{-1} \, \mathbf{K}^{-1} \, [u_i, v_i, 1]^\top + \mathbf{C}_i - \mathbf{C}_j \right)$$

which can be condensed in the following equation:

$$[u_j, v_j, 1]^\top = \frac{z_i}{z_j} \left(\mathbf{H}_{i,j}^{\infty} \, [u_i, v_i, 1]^\top + \frac{\mathbf{e}_{i,j}}{z_i} \right) \tag{7}$$

where $\mathbf{H}_{i,j}^{\infty} = \mathbf{K} \, \mathbf{R}_{i\to j} \, \mathbf{K}^{-1}$ is the homography which maps points at infinity from image i to image j, and $\mathbf{e}_{i,j} = \mathbf{K} \, \mathbf{R}_{0\to j} \, (\mathbf{C}_i - \mathbf{C}_j)$ is the *epipole* in image j.

The plane-sweeping method consists in assuming the surface to be locally flat during the exhaustive search for the depth z. The homography is then supported by the tangent plane to the surface, characterized by a unit-length normal $\mathbf{n}$ and located at a distance d_i from the optical centre of camera i, whose Cartesian equation is written:

$$\mathbf{n}^\top \mathbf{Q} = d_i \tag{8}$$

Plugging the expression (6) of $\mathbf{Q}$ in (8), we obtain:

$$\frac{1}{z_i} = \frac{\mathbf{n}^\top \mathbf{R}_{0\to i}^{-1} \, \mathbf{K}^{-1} \, [u_i, v_i, 1]^\top}{d_i - \mathbf{n}^\top \mathbf{C}_i} \tag{9}$$

and finally, combining (7) and (9):

$$[u_j, v_j, 1]^\top = \frac{z_i}{z_j} \left(\mathbf{H}_{i,j}^{\infty} + \frac{\mathbf{e}_{i,j} \, \mathbf{n}^\top \mathbf{R}_{0\to i}^{-1} \, \mathbf{K}^{-1}}{d_i - \mathbf{n}^\top \mathbf{C}_i} \right) [u_i, v_i, 1]^\top \tag{10}$$

Thus, the *inter-image homography* depends not only on the camera movement between two poses, but also on the normal vector $\mathbf{n}$ of the tangent plane. Facing the difficulty of estimating the normal, it is usual to assume that this plane is fronto-parallel to the image plane of the first camera i.e., to arbitrarily impose $\mathbf{n}$ colinear to its optical axis. In the next subsection, we show how to estimate this normal from the depth and the gradients of the grey level of a pair of images, which will allow us to use the inter-image homography expressed in (10).

3.3 Surface Normal Estimation

In this subsection, we establish the expression of the normal as a function of the depth and the gradients of the grey levels from two views of known poses. Whatever the type of camera projection, the normal is written, in the world frame:

$$\mathbf{n}(\mathbf{Q}) = \frac{1}{\sqrt{|\nabla z|^2 + 1}} \begin{bmatrix} \nabla z \\ -1 \end{bmatrix} \tag{11}$$

Thus, estimating the gradient $\nabla z = [\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y}]^\top$ suffices to characterize the surface normal. By derivation of (1) along the axes of the world frame, and introducing the notation $I_i(u_i, v_i) = I_i \circ \pi_i \circ \pi_z^{-1}(u, v)$, we get:

$$\nabla_{x,y} I_i(u_i, v_i) = \nabla_{x,y} I(u, v), \quad i \in \{1, \dots, n\} \tag{12}$$

According to the chain rule:

$$\nabla_{x,y} I_i(u_i, v_i) = \mathbf{J}_i^\top \, \nabla I_i(u_i, v_i) \tag{13}$$

where:

$$\mathbf{J}_i^\top = \begin{bmatrix} \frac{\partial u_i}{\partial x} & \frac{\partial v_i}{\partial x} \\ \frac{\partial u_i}{\partial y} & \frac{\partial v_i}{\partial y} \end{bmatrix} \tag{14}$$

On the other hand, since the reference camera is aligned with the world reference frame, and denoting α the number of pixels per meter, we have:

$$\nabla_{x,y} I(u, v) = \alpha \, \nabla I(u, v) \tag{15}$$

From (12), (13) and (15), we get:

$$\mathbf{J}_i^\top \, \nabla I_i(u_i, v_i) = \alpha \, \nabla I(u, v) \tag{16}$$

As we shall see next, the depth gradient can be deduced from this equation, for both orthogonal and perspective projections.

Case of Orthogonal Projection – The change of coordinates of a 3D point $\mathbf{Q}$ from the world frame to the camera frame $\mathcal{R}_i$ writes:

$$[x_i, y_i, z_i]^\top = \mathbf{R}_{0\to i} \, \mathbf{Q} + \mathbf{t}_{0\to i} \tag{17}$$

where, as already said, $\mathbf{R}_{0\to i}$ and $\mathbf{t}_{0\to i}$ characterize the pose of camera i.

Under the assumption of orthogonal projection, it is easy to deduce from (17):

$$\begin{cases} u_i = \alpha \, \left[r_i^{1,1} \, x + r_i^{1,2} \, y + r_i^{1,3} \, z + t_i^1 \right] + u_i^0 \\ v_i = \alpha \, \left[r_i^{2,1} \, x + r_i^{2,2} \, y + r_i^{2,3} \, z + t_i^2 \right] + v_i^0 \end{cases} \tag{18}$$

where $r_i^{j,k}$, $(j,k) \in \{1,2,3\}^2$, and t_i^j, $j \in \{1,2,3\}$, designate the current elements of matrix $\mathbf{R}_{0\to i}$ and of vector $\mathbf{t}_{0\to i}$, respectively, and $[u_i^0, v_i^0]^\top$ are the coordinates of the principal point in image i. By derivation of (18), we get:

$$\mathbf{J}_i^\top = \alpha \begin{bmatrix} r_i^{1,1} + r_i^{1,3} \frac{\partial z}{\partial x} & r_i^{2,1} + r_i^{2,3} \frac{\partial z}{\partial x} \\ r_i^{1,2} + r_i^{1,3} \frac{\partial z}{\partial y} & r_i^{2,2} + r_i^{2,3} \frac{\partial z}{\partial y} \end{bmatrix} \tag{19}$$

From (16) and (19), we finally obtain:

$$\left\{ \begin{bmatrix} r_i^{1,1} & r_i^{2,1} \\ r_i^{1,2} & r_i^{2,2} \end{bmatrix} + \begin{bmatrix} \frac{\partial z}{\partial x} \\ \frac{\partial z}{\partial y} \end{bmatrix} \begin{bmatrix} r_i^{1,3} & r_i^{2,3} \end{bmatrix} \right\} \nabla I_i(u_i, v_i) = \nabla I(u, v) \tag{20}$$

From this equation, we deduce the following expression for the depth gradient $\nabla z = [\frac{\partial z}{\partial x}, \frac{\partial z}{\partial y}]^\top$:

$$\nabla z = \frac{\nabla I(u,v) - \begin{bmatrix} r_i^{1,1} & r_i^{2,1} \\ r_i^{1,2} & r_i^{2,2} \end{bmatrix} \nabla I_i(u_i, v_i)}{\begin{bmatrix} r_i^{1,3} & r_i^{2,3} \end{bmatrix} \nabla I_i(u_i, v_i)} \tag{21}$$

Eq. (21) provides us with a closed-form expression for depth gradient, and hence the surface normal, at every point where the denominator does not vanish. This can happen in the following three cases:

- The vector $\begin{bmatrix} r_i^{1,3} & r_i^{2,3} \end{bmatrix}$ is null in the case of a pure rotation around the optical axis of the camera. In this case, the normal cannot be evaluated in any point on the surface. However, this type of camera movement is to be avoided in the context of multi-view stereo.

- The gradient $\nabla I_i(u_i, v_i)$ may be null at certain points in control image i, particularly if the surface is not sufficiently textured. This will happen in the case of a flat, untextured surface that is uniformly illuminated.

- Finally, it is possible that none of these vectors is null, but that this is the case for their scalar product. However, if we have several control images, it is very unlikely that this scalar product cancels for all of them.

Case of Perspective Projection – For the vast majority of real images, the projection onto the camera is no longer orthogonal but perspective. Extending the previous rationale to perspective projection is not difficult, however we skip the proof for space limitation reasons. We therefore content ourselves with giving the new expressions of $\frac{\partial z}{\partial x}$ and $\frac{\partial z}{\partial y}$, which still involve $\nabla I(u,v) = [\frac{\partial I}{\partial u}, \frac{\partial I}{\partial v}]^\top$ and $\nabla I_i(u_i, v_i) = [\frac{\partial I_i}{\partial u_i}, \frac{\partial I_i}{\partial v_i}]^\top$, but also z and z_i:

$$\begin{cases} \dfrac{\partial z}{\partial x} & = \dfrac{z_i^2\, z\, \frac{\partial I}{\partial u} + z^2 \left(w_i^{2,1}\, \frac{\partial I_i}{\partial u_i} - w_i^{1,1}\, \frac{\partial I_i}{\partial v_i} \right)}{z^2 \left(-w_i^{2,3}\, \frac{\partial I_i}{\partial u_i} + w_i^{1,3}\, \frac{\partial I_i}{\partial v_i} \right) + z_i^2 \left(x\, \frac{\partial I}{\partial u} + y\, \frac{\partial I}{\partial v} \right)} \\ \dfrac{\partial z}{\partial y} & = \dfrac{z_i^2\, z\, \frac{\partial I}{\partial v} + z^2 \left(w_i^{2,2}\, \frac{\partial I_i}{\partial u_i} - w_i^{1,2}\, \frac{\partial I_i}{\partial v_i} \right)}{z^2 \left(-w_i^{2,3}\, \frac{\partial I_i}{\partial u_i} + w_i^{1,3}\, \frac{\partial I_i}{\partial v_i} \right) + z_i^2 \left(x\, \frac{\partial I}{\partial u} + y\, \frac{\partial I}{\partial v} \right)} \end{cases} \tag{22}$$

In these expressions, the coefficients $w_i^{1,k}$, $w_i^{2,k}$ and $w_i^{3,k}$, $k \in \{1,2,3\}$, are defined by the following cross products, where (x_i, y_i, z_i) are already defined in (17):

$$\begin{bmatrix} w_i^{1,k} \\ w_i^{2,k} \\ w_i^{3,k} \end{bmatrix} = \begin{bmatrix} r_i^{1,k} \\ r_i^{2,k} \\ r_i^{3,k} \end{bmatrix} \wedge \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} = \begin{bmatrix} z_i\, r_i^{2,k} - y_i\, r_i^{3,k} \\ x_i\, r_i^{3,k} - z_i\, r_i^{1,k} \\ y_i\, r_i^{1,k} - x_i\, r_i^{2,k} \end{bmatrix}$$

4 Experiments

We conducted experiments with synthetic and real images to quantify the performance of the proposed method.

4.1 Implementation Details

In the following experiments, the size of the patch used for photo-consistency computation is fixed empirically to 9×9 pixels and the number of control views to four. As shown in Section 3, the surface normal can be computed from two views. In practice, we use the median of the set of normals estimated over all the pairs between the reference view and one of the four control views, which is the normal that minimizes the sum of the geodesic distances to these normals.

4.2 Synthetic Data

A synthetic sphere cap of radius 1, whose center is located at $(0, 0, 0)$, cut at the height $\sqrt{1 - 0.7^2}$, is viewed from five poses of the same camera with focal length equal to 1000. In each pose, the camera points towards the cap summit, at a distance equal to 4. In Figure 2-a, the reference camera is displayed in blue, while the four control cameras are displayed in red. The reference view and the four control views, which are generated using a uniform directional lighting parallel to the z-axis, are displayed in Figures 2-b and 2-c.

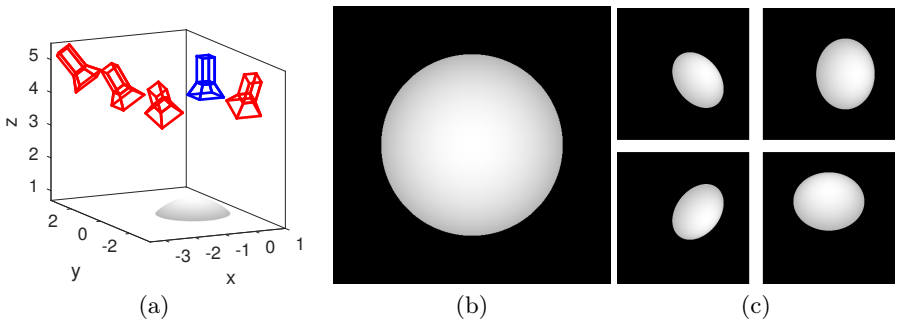


Fig. 2. (a) Synthetic dataset comprising one reference camera (in blue) and four control cameras (in red). The scene is a sphere cap illuminated by a uniform directional lighting parallel to the z-axis. (b) Reference view, whose depth is to be reconstructed. (c) Four control views.

The method described in Section 3 is applied to estimate the surface normal in each pixel of the reference view (see Figure 3-a). The angular errors are shown in Figure 3-b. The normal estimation method shows to perform well, except at the edge of the cap, where the grey level gradient may become infinite.

The slanted plane-sweeping algorithm, which makes use of the estimated normal, is applied to estimate the depth map. The depth errors of plane-sweeping under the fronto-parallel assumption, and of the slanted plane-sweeping method, are shown in Figures 3-c and 3-d. The proposed method is globally more accurate, but this highly depends on the local orientation of the surface.

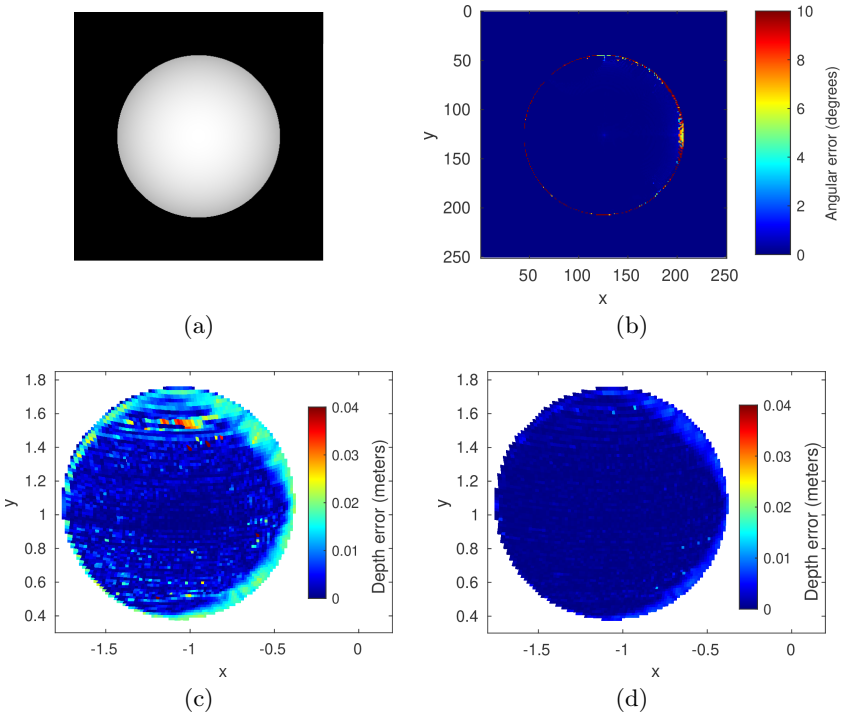


Fig. 3. (a) Reference view. (b) Angular error (in degrees) between the normal estimated according to the method described in Section 3 and the ground truth. (c) Depth error resulting from standard plane-sweeping, using a fronto-parallel patch. (d) Depth error resulting from the proposed slanted plane-sweeping, using a patch oriented according to the estimated normal. The proposed method is globally more accurate.

The spherical cap is then segmented into five regions, based on the angle between the surface normal and the optical axis of the reference camera, which are highlighted in color in Figure 4-a. The mean and median depth errors per region are shown in Figure 4-b, in the absence of noise in the images, and in Figure 4-c (resp. 4-d), by adding a uniform noise in the range $[-3, 3]$ (resp. $[-6, 6]$) to the images, whose grey level values are between 0 and 255. Depth errors shown by the proposed slanted plane-sweeping are overall lower than the ones obtained under the fronto-parallel assumption, but this is particularly true for areas nearby the cap edge, associated to the highest out-of-plane rotations of the tangent plane, strongly violating the fronto-parallel patch assumption.

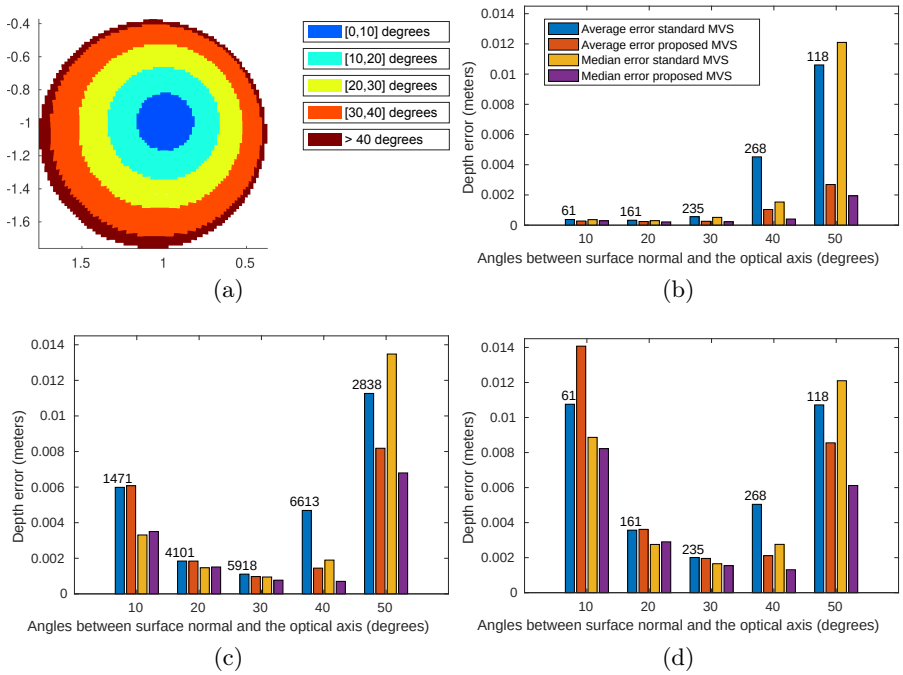
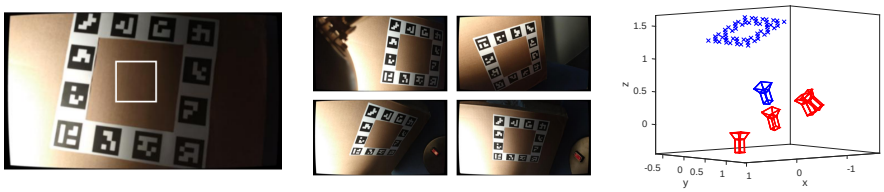


Fig. 4. (a) Segmentation of the reference view in five regions, based on the orientation of the surface normal. Histograms of the average error and of the median error on the estimated depth, using standard or slanted plane-sweeping methods: (b) in the absence of noise in the images; (c) adding to the images a uniform noise in the range $[-3, 3]$; (d) adding to the images a uniform noise in the range $[-6, 6]$.

4.3 Real Data

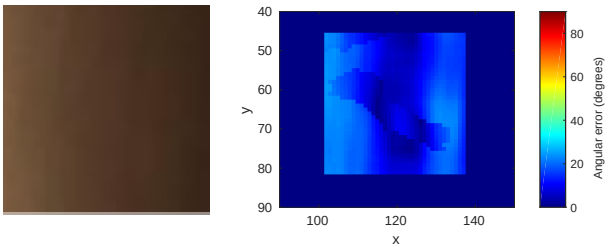
Then, the method has been evaluated on a real dataset, which is a set of five views of a plane. ArUco markers have been glued on the plane, in order to obtain its 3D reconstruction, along with the camera poses, using the structure-from-motion pipeline AliceVision Meshroom [13]. The normals of the plane are obtained by standard plane fitting from the marker locations, and used as ground truth. The input views are shown in Figures 5-a and 5-b, while the results of structure-from-motion are shown in Figure 5-c.

The angular errors, which are shown in Figure 6-a, show a mean of 13.9^o and a standard deviation of 6.75^o. In this real scenario, the errors obtained on the estimated normals are too large to be exploited by the proposed slanted plane-sweeping. This may partly be explained by a gradient computation very sensitive to noise or by inaccuracies in camera pose estimation.



(a) (b) (c)

Fig. 5. (a) Reference view. (b) Four control views. The depth errors will be evaluated within the white frame shown in (a). (c) Real camera and marker positions, computed using a standard structure-from-motion technique [13]. The reference camera is shown in blue, the control ones in red.



(a) (b)

Fig. 6. (a) Image area over which normal estimation is performed (highlighted in white in Figure 5-a). (b) Angular errors on the estimated normals, which show a mean and standard deviation of 13.9^o and 6.75^o, respectively, computed over 1296 pixels.

5 Conclusion and Perspectives

In this paper, we show how to estimate the normals of a surface from two views, provided that the poses of the camera, assumed calibrated, are perfectly known. For a pair of homologous points, we show that the normal at the corresponding 3D point can be computed unambiguously, as a function of the grey level gradient at each of these points, mentioning three cases for which this estimation is impossible. We detail the estimation method in the case of an orthogonal projection, and give its generalization to the perspective case.

Among the various applications of this new method of normal estimation, it makes it possible to estimate the inter-image homography of a surface portion, assumed to be locally planar, which depends not only on the change of pose, but also on the local orientation of the plane. This is therefore of interest for plane-sweeping matching, which is usually based on an approximate estimate of the inter-image homography. The tests carried out on synthetic images validate the theoretical part of our approach, and show that it is indeed worthwhile to take into account the local orientation of the surface in the criterion of photo-consistency. The tests on real data are less convincing, as the estimation of the normal by our approach gives too high angular errors, of the order of 10^o, to be able to claim an improvement of plane-sweeping matching.

Among the follow-ups to this work, we should first make the normal estimation method more robust. In addition to a possible inaccurate estimation of the camera poses (we used only five images in the real dataset), it is likely that the computation of the grey level gradients, as we did it, is grossly lacking in robustness. Moreover, a purely local estimate of the normal may be inherently too sensitive to noise. Another perspective is therefore to estimate the normals for a set of neighboring points, assumed to belong to a common tangent plane. It will then be appropriate to make the link with the work by Bartoli et al. on the estimation of normals from the deformations of a template [2].

References

1. Agarwal, S., Furukawa, Y., Snavely, N., Simon, I., Curless, B., Seitz, S.M., Szeliski, R.: Building Rome in a day. Commun. ACM **54**(10), 105–112 (2011)
2. Bartoli, A., Gérard, Y., Chadebecq, F., Collins, T., Pizarro, D.: Shape-from-template. PAMI **37**(10), 2099–2118 (2015)
3. Birchfield, S., Tomasi, C.: Multiway Cut for Stereo and Motion with Slanted Surfaces. In: ICCV (1999)
4. Burt, P.J., Wixson, L.E., Salgian, G.: Electronically Directed "Focal" Stereo. In: ICCV (1995)
5. Chen, W., Ling, H., Gao, J., Smith, E.J., Lehtinen, J., Jacobson, A., Fidler, S.: Learning to Predict 3D Objects with an Interpolation-based Differentiable Renderer. In: NeurIPS (2019)
6. Chen, W., Litalien, J., Gao, J., Wang, Z., Tsang, C.F., Khamis, S., Litany, O., Fidler, S.: DIB-R++: Learning to Predict Lighting and Material with a Hybrid Differentiable Renderer. In: NeurIPS (2021)
7. Furukawa, Y., Hernàndez, C.: Multi-view stereo: A tutorial. Foundations and Trends in Computer Graphics and Vision **9**(1–2) (2013)
8. Furukawa, Y., Ponce, J.: Accurate, Dense, and Robust Multiview Stereopsis. PAMI **32**(8), 1362–1376 (2010)
9. Galliani, S., Lasinger, K., Schindler, K.: Massively Parallel Multiview Stereopsis by Surface Normal Diffusion. In: ICCV. pp. 873–881 (2015)
10. Gallup, D., Frahm, J., Mordohai, P., Yang, Q., Pollefeys, M.: Real-Time Plane-Sweeping Stereo with Multiple Sweeping Directions. In: CVPR (2007)
11. Gårding, J., Lindeberg, T.: Direct computation of shape cues using scale-adapted spatial derivative operators. IJCV **17**(2), 163–191 (1996)
12. Goesele, M., Curless, B., Seitz, S.M.: Multi-view stereo revisited. In: CVPR. vol. 2, pp. 2402–2409 (2006)
13. Griwodz, C., Gasparini, S., Calvet, L., Gurdjos, P., Castan, F., Maujean, B., De Lillo, G., Lanthony, Y.: AliceVision Meshroom: An open-source 3D reconstruction pipeline. In: ACM Multimedia Systems Conference. pp. 241–247 (2021)
14. Hernández, C., Schmitt, F.: Silhouette and stereo fusion for 3D object modeling. CVIU **96**(3), 367–392 (2004)
15. Jiang, Y., Ji, D., Han, Z., Zwicker, M.: SDFDiff: Differentiable Rendering of Signed Distance Fields for 3D Shape Optimization. In: CVPR (2020)
16. Kazhdan, M.M., Hoppe, H.: Screened poisson surface reconstruction. ACM Trans. Graph. **32**(3), 29:1–29:13 (2013)
17. Liao, Y., Donné, S., Geiger, A.: Deep Marching Cubes: Learning Explicit Surface Representations. In: CVPR (2018)

18. Lindeberg, T., Gårding, J.: Shape-adapted smoothing in estimation of 3-D depth cues from affine distortions of local 2-d brightness structure. In: ECCV (1994)
19. Liu, H.D., Williams, F., Jacobson, A., Fidler, S., Litany, O.: Learning Smooth Neural Functions via Lipschitz Regularization. In: SIGGRAPH (2022)
20. Liu, J., Ji, P., Bansal, N., Cai, C., Yan, Q., Huang, X., Xu, Y.: PlaneMVS: 3D Plane Reconstruction from Multi-View Stereo. In: CVPR (2022)
21. Liu, S., Chen, W., Li, T., Li, H.: Soft Rasterizer: A Differentiable Renderer for Image-Based 3D Reasoning. In: ICCV (2019)
22. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: NeRF: representing scenes as neural radiance fields for view synthesis. Commun. ACM **65**(1), 99–106 (2022)
23. Munkberg, J., Chen, W., Hasselgren, J., Evans, A., Shen, T., Müller, T., Gao, J., Fidler, S.: Extracting Triangular 3D Models, Materials, and Lighting From Images. In: CVPR (2022)
24. Niemeyer, M., Mescheder, L.M., Oechsle, M., Geiger, A.: Differentiable Volumetric Rendering: Learning Implicit 3D Representations Without 3D Supervision. In: CVPR (2020)
25. Schönberger, J.L., Zheng, E., Frahm, J., Pollefeys, M.: Pixelwise View Selection for Unstructured Multi-View Stereo. In: ECCV (2016)
26. Shen, T., Gao, J., Yin, K., Liu, M., Fidler, S.: Deep Marching Tetrahedra: a Hybrid Representation for High-Resolution 3D Shape Synthesis. In: NeurIPS (2021)
27. Strecha, C., Fransens, R., Gool, L.V.: Wide-Baseline Stereo from Multiple Views: A Probabilistic Account. In: CVPR (2004)
28. Zabulis, X., Daniilidis, K.: Multi-Camera Reconstruction based on Surface Normal Estimation and Best Viewpoint Selection. In: 3DPVT (2004)