Laplacian Meshes for Monocular 3D Shape Recovery

Jonas Östlund, Aydin Varol, Dat Tien Ngo and Pascal Fua*

EPFL – CVLab
1015 Lausanne, Switzerland

Abstract. We show that by extending the Laplacian formalism, which was first introduced in the Graphics community to regularize 3D meshes, we can turn the monocular 3D shape reconstruction of a deformable surface given correspondences with a reference image into a well-posed problem. Furthermore, this does not require any training data and eliminates the need to pre-align the reference shape with the one to be reconstructed, as was done in earlier methods.

1 Introduction

Shape recovery of deformable surfaces from single images is inherently ambiguous, given that many different configurations can produce the same projections. In particular, this is true of template-based methods, in which a *reference* image of the surface in a known configuration is available and point correspondences between this reference image and an *input* image in which the shape is to be recovered are given.

It has been shown that solving this problem amounts to solving a degenerate linear system, which requires either reducing the number of degrees of freedom or imposing additional constraints [1]. The first can be achieved by various dimensionality reduction techniques [2, 3] while the second often involves assuming the surface to be either developable [4–6] or inextensible [7–9, 3]. These two approaches are sometime combined and augmented by introducing additional sources of information such as shading or textural clues [10, 11]. The resulting algorithms usually require solving a fairly large optimization problem and, even though it is often well behaved or even convex [9, 7, 12, 13, 3], it remains computationally demanding. Closed-form solutions have been proposed [14, 11] but they also involve solving very large equation systems and to make more restrictive assumptions than the optimization-based ones, which can lower the performance [3].

Here, we show that, by extending the Laplacian formalism first introduced in the Graphics Community [15, 16], we can turn the degenerate linear system mentioned above into a smaller non-degenerate one. More specifically, we express all vertex coordinates as linear combinations of those of a small number of

* This work was supported in part by the Swiss National Science Foundation

control vertices and inject this expression into the large system to produce the small one we are looking for. Well-posedness stems from an implicit and novel regularization term, which penalizes deviations away from the reference shape, which does not have to be planar.

In other words, instead of performing a minimization involving many degress of freedom or solving a large system of equations, we end up simply solving a very compact linear system. This yields an initial 3D shape estimate, which is not necessarily accurate, but whose 2D projections are. In practice, this is extremely useful to quickly and reliably eliminate erroneous correspondences. We can then achieve better accuracy than the method of [3], which has been shown to outperform other recent ones, by enforcing the same inextensibility constraints. This is done by optimizing with respect to far fewer variables, thereby lowering the computational complexity, and without requiring training data as [3] does.

Furthermore, unlike in other recent approaches that also rely on control points [9, 17], our implicit regularization term is orientation invariant. As a result, we do not have to pre-align the reference shape with the one to be recovered or to precompute a rotation matrix. To explore this aspect, we replaced our Laplacian approach to dimensionality reduction by a simpler one based on the same linear deformation modes as those of [14, 11] and again enforced the inextensibility constraints of [3]. We will show that without pre-alignment, it performs much worse. With manual pre-alignment it can yield the same accuracy but an automated version would necessitate an additional non-linear optimization step. Furthermore, computing the deformation modes requires either training data or a stiffness matrix, neither of which may be forthcoming.

In short, our contribution is a novel approach to reducing the dimensionality of the surface reconstruction problem. It removes the need both for an estimate of the rigid transformation with respect to the reference shape and for access to training data or material properties, which may be unavailable or unknown. Furthermore, this is achieved without sacrificing accuracy.

In the remainder of this paper, we first review existing approaches and remind the reader of how the problem can be formulated as one of solving a linear but degenerate system of equations, as was done in earlier approaches [1]. We then introduce our extended Laplacian formalism, which lets us transform the degenerate linear system into a smaller well-posed one. Finally, we present our results and compare them against state-of-the-art methods.

2 Related Work

Reconstructing the 3D shape of a non-rigid surface from a single input image is a severely under-constrained problem, even when a reference image of the surface in a different but known configuration is available, which is the problem we address here.

When point correspondences can be established between the reference and input images, shape-recovery can be formulated in terms of solving an ill-conditioned linear-system [1] and requires the introduction of additional constraints to make

it well-posed. The most popular ones involve preserving Euclidean or Geodesic distances as the surface deforms and are enforced either by solving a convex optimization problem [9, 7, 8, 12, 13, 3] or by solving in closed form sets of quadratic equations [14, 11]. The latter is typically done by linearization, which results in very large systems and is no faster than minimizing a convex objective function, as is done in [3] which is one of the best representatives of this class of techniques.

The complexity of the problem can be reduced using a dimensionality reduction technique such as Principal Component Analysis (PCA) to create morphable models [2, 18], modal analysis [11, 12], or Free Form Deformations (FFDs) [9]. As shown in the result section, PCA and modal analysis can be coupled with constraints such as those discussed above to achieve good accuracy. However, this implies using training data or being able to build a stiffness matrix, which is not always possible. The FFD approach [9] is particularly relevant since, like ours, it relies on parameterizing the surface in terms of control points. However, it requires placing the control points on a regular grid, whereas ours can be arbitrarily chosen for maximal accuracy. This makes our approach closer to earlier ones to fitting 3D surfaces to point clouds that rely on Dirichlet Free Form Deformations and also allow arbitrary placement of the control points [19].

Furthermore, none of these dimensionality reduction methods allow orientation-invariant regularization because deformations are minimized with respect to a reference shape or a set of control points that must be properly oriented. Thus, global rotations must be explicitly handled to achieve satisfactory results. To overcome this limitation, we took our inspiration from the Laplacian formalism presented in [15] and the rotation invariant formulation of [16], which like ours involves introducing virtual vertices. In both these papers, the mesh Laplacian is used to define a regularization term that tends to preserve the shape of a non-planar surface. Furthermore, vertex coordinates can be parameterized as a linear combination of those of control vertices to solve specific minimization problems [16]. In Section 4, we will go one step further by providing a generic linear expression of the vertex coordinates as a function of those of the control vertices, independently of the objective function to be minimized.

3 Linear Problem Formulation

As shown in [1], given point correspondences between a reference image in which the 3D shape is known and an input image, recovering the new shape in that image amounts to solving the linear system

$$\mathbf{M}\mathbf{x} = \mathbf{0}\ , \text{where } \mathbf{x} = \begin{bmatrix} \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_{N_v} \end{bmatrix}\ , \tag{1}$$

where $\mathbf{v}_i$ contains the 3D coordinates of the i^{th} vertex of the N_v-vertex triangulated mesh representing the surface, and $\mathbf{M}$ is a matrix that depends on the coordinates of correspondences in the input image and on the camera internal parameters. A solution of this system defines a surface such that 3D feature points

that project at specific locations in the reference image reproject at matching locations in the input image. Solving this system in the least-squares sense therefore yields surfaces, up to a scale factor, for which the overall reprojection error is small.

The difficulty comes from the fact that, for all practical purposes, $\mathbf{M}$ is rank deficient, with at least one third of its singular values being extremely small with respect to the other two thirds even when there are many correspondences. This is why the inextensibility constraints discussed in Section 2 were introduced.

A seemingly natural way to address this issue is to introduce a linear subspace model and to write surface deformations as linear combinations of relatively few basis vectors. This can be expressed as

$$\mathbf{x} = \mathbf{x}_0 + \sum_{i=1}^{N_s} w_i \mathbf{b}_i = \mathbf{x}_0 + \mathbf{B}\mathbf{w} \;, \tag{2}$$

where $\mathbf{x}$ is the coordinate vector or Eq. 1, $\mathbf{B}$ is the matrix whose columns are the $\mathbf{b}_i$ basis vectors typically taken to be the eigenvectors of a stiffness matrix, and $\mathbf{w}$ is the associated vectors of weights w_i. Injecting this expression into Eq. 1 and adding a regularization term yields a new system

$$\begin{bmatrix} \mathbf{MB} & \mathbf{M}\mathbf{x}_0 \\ \lambda_r \mathbf{L} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{w} \\ 1 \end{bmatrix} = \mathbf{0} \;, \tag{3}$$

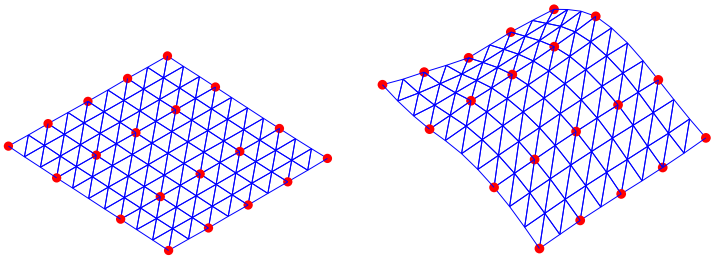
which is to be solved in the least squares sense, where $\mathbf{L}$ is a diagonal matrix whose elements are the inverse values of the eigenvalues associated to the basis vectors, and λ_r is a regularization weight. This favors basis vectors that correspond to the lowest-frequency deformations and therefore enforces smoothness.

In practice, the linear system of Eq. 3 is less poorly conditioned than the one of Eq. 1. But, because there usually are several *smooth* shapes that all yield virtually the same projection, its matrix still has a number of near zero singular values. As a consequence, additional constraints still need to be imposed for the problem to become well-posed. An additional difficulty is that, because rotations are strongly non-linear, this linear formulation can only handle small ones. As a result, the rest shape defined by $\mathbf{x}_0$ must be roughly aligned with the shape to be recovered, which means that a global rotation must be computed before shape recovery can be attempted.

In the remainder of this paper, we will show that we can reduce the dimensionality in a different way, which allows us to minimize the reprojection error by solving a small and well-conditioned linear system.

4 Laplacian Formulation

In the previous section, we introduced the linear system of Eq. 1, which is so badly conditioned that we cannot minimize the reprojection error by simply solving it. In this section, we show how to turn it into a much smaller and well-conditioned system.



(a) (b)

Fig. 1. Linear parameterization of mesh vertices using control points (a) Vertices and control points in their rest state. (b) Control points and vertices in a deformed state. Every vertex is a linear combination of the control points.

To this end, let us assume we are given a reference shape, which may be planar or not and let $\mathbf{x}_{\text{ref}}$ be the coordinate vector of its vertices, as the ones highlighted in Fig. 1. We first show that we can define a regularization matrix $\mathbf{A}$ such that $\|\mathbf{Ax}\|^2 = 0$, with $\mathbf{x}$ being the coordinate vector of Eq. 1, when $\mathbf{x}_{\text{ref}} = \mathbf{x}$ up to a rigid transformation. In other words, $\|\mathbf{Ax}\|^2$ penalizes non-rigid deformations away from the reference shape but not rigid ones. We then show that, given a subset of N_c mesh vertices whose coordinates are

$$\mathbf{c} = \begin{bmatrix} \mathbf{v}_{i_1} \\ \vdots \\ \mathbf{v}_{i_{N_c}} \end{bmatrix} , \tag{4}$$

if we force the mesh both to go through these coordinates and to minimize $\|\mathbf{Ax}\|^2$, we can define a matrix $\mathbf{P}$ that is independent of the control vertex coordinates and such that

$$\mathbf{x} = \mathbf{Pc} \ . \tag{5}$$

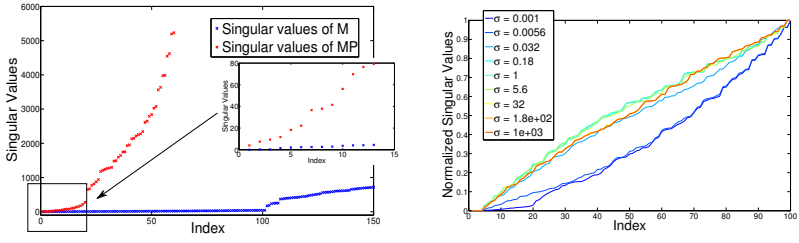
In other words, we can linearly parameterize the mesh as a function of the control vertices' coordinates, as illustrated in Fig. 1. Injecting this parameterization into Eq. 1, yields

$$\mathbf{MPc} = \mathbf{0} \, , \tag{6}$$

where $\mathbf{MP}$ is a matrix without any zero singular values for sufficiently small numbers of control points as demonstrated by Fig. 2(a). This system can therefore be solved in the least-square sense up to a scale factor by finding the eigenvector corresponding to its smallest singular value. In practice, because the correspondences can be noisy, we further regularize by solving

$$\min_{\mathbf{c}} \quad \|\mathbf{MPc}\|^2 + w_r \|\mathbf{APc}\|^2, \text{ s. t. } \|\mathbf{c}\| = 1 \, , \tag{7}$$

where w_r is a scalar coefficient and which can still be done in closed form. We will show in the results section that this yields accurate reconstructions, even with relatively small numbers of control points.



(a) (b)

Fig. 2. Singular values **(a)** of **MP** and **M** drawn with red and blue markers respectively for a planar model for a given set of correspondences. The smallest singular value of **MP** is about four whereas it is practically zero for the first N_v singular values of **M**. Normalized singular values **(b)** of regularization matrix **A** computed on a non-planar model for various values of σ. Note that the curves are almost superposed for σ values greater than one.

4.1 Regularization Matrix

We now turn to building the matrix **A** such that $\mathbf{A}\mathbf{x}_{\text{ref}} = \mathbf{0}$ and $\|\mathbf{A}\mathbf{x}\|^2 = \|\mathbf{A}\mathbf{x}'\|^2$ when $\mathbf{x}'$ is a rigidly transformed version of $\mathbf{x}$. We first propose a very simple scheme for the case when the reference shape is planar and then a similar, but more sophisticated one, when it is not.

Planar Rest Shape Given a planar mesh that represents a reference surface in its rest shape, consider every pair of facets that share an edge, such as those depicted by Fig. 3(a). They jointly have four vertices $\mathbf{v}_{1\text{ref}}$ to $\mathbf{v}_{4\text{ref}}$. Since they lie on a plane, whether or not the mesh is regular, one can always find a unique set of weights w_1, w_2, w_3 and w_4 such that

$$\begin{aligned} \mathbf{0} &= w_1\mathbf{v}_{1\text{ref}} + w_2\mathbf{v}_{2\text{ref}} + w_3\mathbf{v}_{3\text{ref}} + w_4\mathbf{v}_{4\text{ref}} \,, \\ 0 &= w_1 + w_2 + w_3 + w_4 \,, \\ 1 &= w_1^2 + w_2^2 + w_3^2 + w_4^2 \,, \end{aligned} \tag{8}$$

up to a sign ambiguity, which can be resolved by simply requiring the first weight to be positive. To form the **A** matrix, we first generate a matrix $\mathbf{A}'$ for 1-dimensional vertex coordinates by considering every facet pair configuration i of four vertices j_1, j_2, j_3, j_4 and set the elements of $\mathbf{A}'$ taken to be $a'_{ij} = w_j$ for

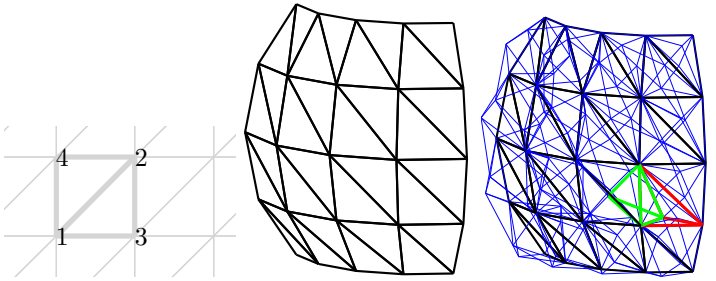


Fig. 3. Building the $\mathbf{A}$ regularization matrix of Section 4.1. (a) Two facets that share an edge. (b) Non-planar reference mesh. (c) Non-planar reference mesh with virtual vertices and edges added. One of the pairs of tetrahedra used to construct the matrix is shown in red.

$j \in \{j_1, j_2, j_3, j_4\}$. All remaining elements are set to zero and the matrix $\mathbf{A}$ is $\mathbf{A} = \mathbf{A}' \otimes \mathbf{I}_3$.

It is easy to verify that $\mathbf{Ax} = \mathbf{0}$ as long as $\mathbf{x}$ represents a planar mesh and that $\mathbf{Ax}$ is invariant to rotations and translations, and in fact to all affine deformations. The first equality of Eq. 8 is designed to enforce planarity while the second guarantees invariance. The third equality is there to prevent the weights from all being zero.

Non-Planar Rest Shape When the rest shape is non-planar, there will be facets for which we cannot solve Eq. 8. As in [16], we extend the scheme described above by introducing virtual vertices. As shown in Fig. 3(c), we create virtual vertices at above and below the center of each facet as a distance controlled by the scale parameter σ. Formally, for each facet $\mathbf{v}_i$, $\mathbf{v}_j$ and $\mathbf{v}_k$, its center $\mathbf{v}_c = \frac{1}{3}(\mathbf{v}_i + \mathbf{v}_j + \mathbf{v}_k)$ and its normal $\mathbf{n} = (\mathbf{v}_j - \mathbf{v}_i) \times (\mathbf{v}_k - \mathbf{v}_i)$, we take the virtual vertices to be

$$\mathbf{v}^+_{ijk} = \mathbf{v}_c + \sigma \frac{\mathbf{n}}{\sqrt{\|\mathbf{n}\|}} \qquad \text{and} \qquad \mathbf{v}^-_{ijk} = \mathbf{v}_c - \sigma \frac{\mathbf{n}}{\sqrt{\|\mathbf{n}\|}} \,, \tag{9}$$

where the norm of $\mathbf{n}/\sqrt{\|\mathbf{n}\|}$ is approximately equal to the average edge-length of the facet's edges. These new vertices, along with the original ones, are used to build the blue tetrahedra of Fig. 3(c).

Let $\mathbf{x}^{\mathrm{V}}$ be the coordinate vector of the virtual vertices only, and $\mathbf{x}^{\mathrm{A}} = \left[\mathbf{x}^T, \mathbf{x}^{\mathrm{V}T}\right]^T$ the coordinate vector of both real and virtual vertices. Let $\mathbf{x}^{\mathrm{A}}_{\mathrm{ref}}$ be similarly defined for the reference shape. Given two tetrahedra that share a facet, such as the red ones in Fig. 3(c), and the five vertices $\mathbf{v}_{1^{\mathrm{A}}_{\mathrm{ref}}}$ to $\mathbf{v}_{5^{\mathrm{A}}_{\mathrm{ref}}}$ they

share, we can now find weights w_1 to w_5 such that

$$\begin{aligned} \mathbf{0} &= w_1 \mathbf{v}_1{}^{\mathrm{A}}_{\mathrm{ref}} + w_2 \mathbf{v}_2{}^{\mathrm{A}}_{\mathrm{ref}} + w_3 \mathbf{v}_3{}^{\mathrm{A}}_{\mathrm{ref}} + w_4 \mathbf{v}_4{}^{\mathrm{A}}_{\mathrm{ref}} + w_5 \mathbf{v}_5{}^{\mathrm{A}}_{\mathrm{ref}} \ , \\ 0 &= w_1 + w_2 + w_3 + w_4 + w_5 \ , \\ 1 &= w_1^2 + w_2^2 + w_3^2 + w_4^2 + w_5^2 \ . \end{aligned} \tag{10}$$

The three equalities of Eq. 10 serve the same purpose as those of Eq. 8. We form a matrix $\mathbf{A}^{\mathrm{A}}$ by considering all pairs of tetrahedra that share a facet, computing the w_1 to w_5 weights that encode local linear dependencies between real and virtual vertices, and using them to add successive rows to the matrix, as we did to build the $\mathbf{A}$ matrix in the planar case. It is again easy to verify to $\mathbf{A}^{\mathrm{A}}\mathbf{x}^{\mathrm{A}}_{\mathrm{ref}} = \mathbf{0}$ and that $\mathbf{A}^{\mathrm{A}}\mathbf{x}^{\mathrm{A}}$ is invariant to affine deformations of $\mathbf{x}^{\mathrm{A}}$. In our scheme, the regularization term can be computed as

$$C = \left\| \mathbf{A}^{\mathrm{A}} \mathbf{x}^{\mathrm{A}} \right\|^2 = \left\| \hat{\mathbf{A}}\mathbf{x} + \tilde{\mathbf{A}}\mathbf{x}^{\mathrm{V}} \right\|^2 \ , \tag{11}$$

if we write $\mathbf{A}^{\mathrm{A}}$ as $\left[\hat{\mathbf{A}}\ \tilde{\mathbf{A}}\right]$ where $\hat{\mathbf{A}}$ has as three times as many columns as there are real vertices and $\tilde{\mathbf{A}}$ as there are virtual ones. Given the real vertices $\mathbf{x}$, the virtual vertex coordinates that minimize the C term of Eq. 11 is

$$\begin{aligned} \mathbf{x}^{\mathrm{V}} &= -\left(\tilde{\mathbf{A}}^T \tilde{\mathbf{A}}\right)^{-1} \tilde{\mathbf{A}}^T \hat{\mathbf{A}}\mathbf{x} \\ \Rightarrow C &= \left\| \hat{\mathbf{A}}\mathbf{x} - \tilde{\mathbf{A}} \left(\tilde{\mathbf{A}}^T \tilde{\mathbf{A}}\right)^{-1} \tilde{\mathbf{A}}^T \hat{\mathbf{A}}\mathbf{x} \right\|^2 = \|\mathbf{A}\mathbf{x}\|^2 \ , \end{aligned} \tag{12}$$

where $\mathbf{A} = \hat{\mathbf{A}} - \tilde{\mathbf{A}} \left(\tilde{\mathbf{A}}^T \tilde{\mathbf{A}}\right)^{-1} \tilde{\mathbf{A}}^T \hat{\mathbf{A}}$. In other words, this matrix $\mathbf{A}$ is the regularization matrix we are looking for and its elements depend only on the coordinates of the reference vertices and on the distance σ chosen to build the virtual vertices.

To study the influence of the scale parameter σ of Eq. 9, which controls the distance of the virtual vertices from the mesh, we computed the $\mathbf{A}$ matrix and its singular values for a sail shaped mesh and many σ values. As can be seen in Fig. 2(b), there is a wide range for which the distribution of singular values remains practically identical. This suggests that σ has limited influence on the numerical character of the regularization matrix. In all our experiments, we set σ to 1 and were nevertheless able to obtain accurate reconstructions of many different surfaces with very different physical properties.

4.2 Linear Parameterization

As before, let N_v be the total number of vertices and $N_c < N_v$ the number of control points we want to use to parameterize the mesh. Given the coordinate vector $\mathbf{c}$ of these control vertices, the coordinates of the minimum energy mesh that goes through the control vertices and minimizes the regularization energy can be found by minimizing

$$\|\mathbf{A}\mathbf{x}\|^2 \ \text{subject to} \ \mathbf{P}_{\mathrm{c}}\mathbf{x} = \mathbf{c} \ , \tag{13}$$

where $\mathbf{A}$ is the regularization matrix introduced above and $\mathbf{P}_c$ is a $3N_c \times 3N_v$ matrix containing ones in columns corresponding to the indices of control vertices and zeros elsewhere. We prove in the supplementary material that, then, there exists a matrix $\mathbf{P}$ whose components can be computed from $\mathbf{A}$ and $\mathbf{P}_c$ such that

$$\forall \mathbf{c} \ , \ \mathbf{x} = \mathbf{P}\mathbf{c} \ , \tag{14}$$

if $\mathbf{x}$ is solution of the minimization problem of Eq. 13.

4.3 Reconstruction

Given that we can write the coordinate vector $\mathbf{x}$ as $\mathbf{Pc}$ and given potentially noisy correspondences between the reference and input images, recall that we can find $\mathbf{c}$ as the solution of the linear least-squares problem of Eq. 7.

As shown in the result section, this yields a mesh whose reprojection is very accurate but whose 3D shape may not be because our regularization does not penalize affine deformations away from the rest shape. In practice, we use this initial mesh to eliminate erroneous correspondences. We then refine it by solving

$$\min_{\mathbf{c}} \quad \|\mathbf{MPc}\|^2 + w_r\|\mathbf{APc}\|^2, \text{ s. t. } C\left(\mathbf{Pc}\right) \leq 0 \ , \tag{15}$$

where $C\left(\mathbf{Pc}\right)$ are inextensibility constraints that prevent distances between neighboring vertices to grow beyond a bound, such as their Euclidean distance in their rest shape when it is planar. These are exactly the same constraints as those used in [3], against which we compare ourselves below. The inequality constraints are reformulated as equality constraints with additional slack variables whose norm is penalized to prevent the solution from shrinking to the origin.

4.4 Complexity and Robustness

To handle outliers in the correspondences, we iteratively perform the unconstrained optimization of Eq. (7) starting with a relatively high regularization weight w_r and reducing it by half at each iteration. Given a current shape estimate, we project it on the input image and disregard the correspondences with higher reprojection error than a pre-set radius and reduce it by half for the next iteration. Repeating this procedure a fixed number of times results in an initial shape estimate and provides inlier correspondences for the more computationally demanding constrained optimization that follows. This decoupled outlier rejection mechanism brings us a computational advantage against the state-of-the-art reconstruction method [3] which relies on a similar iterative method but solves the nonlinear optimization at each iteration. Furthermore, while dealing with the same number of constraints, our method typically uses ten times fewer degrees-of-freedom (i.e. $N_c < (\frac{1}{10} \times N_v)$), and thus solves the constrained least square problem of Eq 15 much faster.

5 Results

We compare our approach to surface reconstruction from a single input image given correspondences with a reference image against the recent method of [3], which has been shown to outperform closed-form solutions based on a global deformation model and inextensibility constraints [14] and solutions that rely on a nonlinear local deformation models [20]. In all these experiments, we set the initial regularization weight w_r of Eq. 7 to $2e4$, the scale parameter σ of Eq. 9 to 1 and the initial radius for outlier rejection to 150 pixels.

To complete this comparison, we also implemented and tested an approach to reducing dimensionality using a linear subspace model as discussed in Section 3 and solving in the least squares sense the system of Eq. 3 under the same inextensibility constraints as those of Eq. 15. We took our basis vectors to be the eigenvectors of a stiffness matrix [21–23] and chose their number so that the number of degrees of freedom using either the subspace approach or ours would be the same. For both approaches, we report the results with and without performing a rigid alignment of the reference model with the image prior to starting the minimization.

In all of these experiments we used SIFT to compute correspondences between a reference image in which the the 3D shape is known and individual other images. We assume the internal camera parameters are known and reason in camera coordinate frame. In the remainder of the section, we first evaluate the accuracy of our reconstruction method against the competing methods on images for which we have ground truth. We then present results on images of surfaces made of different materials.

5.1 Quantitative Results

To provide quantitative results both in the planar and non-planar cases, we acquired images of a deforming piece of paper using a Kinect$^{\text{tm}}$ camera and of a deforming sail using a pair of cameras.

Paper Sequence. Our qualitative results are depicted by Fig. 4. The rest shape is flat and modeled by a 11×9 rectangular mesh and parameterized by 20 regularly spaced control points such as those of Fig. 1. For reconstruction purposes, we ignored the depth maps and used only the color images, along with SIFT correspondences with the reference image. For evaluation purposes, we measured the distance of the recovered vertices to the depth map.

We report the 3D reconstruction errors for different methods in Fig. 6(a). Our approach yields the highest average accuracy, and it is followed closely by the subspace method provided that the model is properly pre-aligned by computing global rotation and translation.

Sail Images. We applied our reconstruction algorithm on sail images such as the one of Fig. 5. Since images taken by a second camera are available we used a simple triangulation method to generate 3D points corresponding to the circular

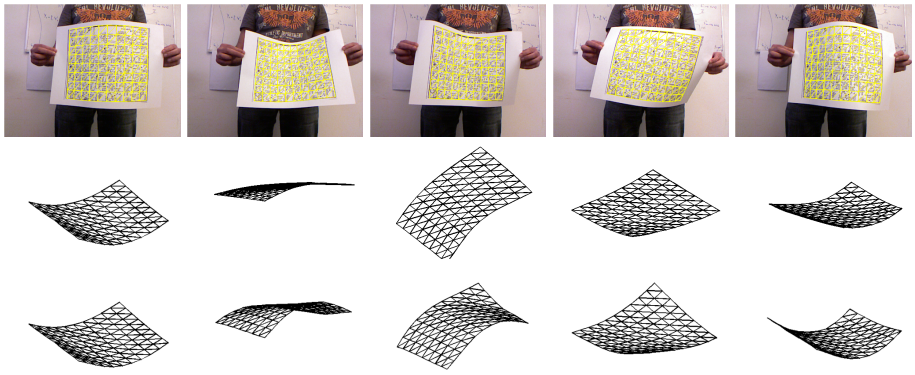


Fig. 4. Paper sequence with ground-truth: Top row: Reprojection of the constrained reconstructions on the input images **Middle row:** Unconstrained solutions seen from a different view-point **Bottom row:** Constrained solutions seen from the same view-point. We supply the corresponding video sequence as supplementary materials.

markers on the sail which serve as ground-truth measurements. We compare the reconstructions against the scaled ground-truth points such that the mean distance between the predicted marker positions and their ground-truth positions is minimized. In this case, as shown in Fig. 6(b) , our method achieves the highest accuracy against the competing methods.

Our C++ implementation relies on the Ferns keypoint classifier [24] to match feature points and average computation times per frame are given in Fig. 6(c) as a function of the numbers of control points. As stated above, the accuracy results of Figs. 6(a,b) were obtained using 20 control points.

5.2 Qualitative Results

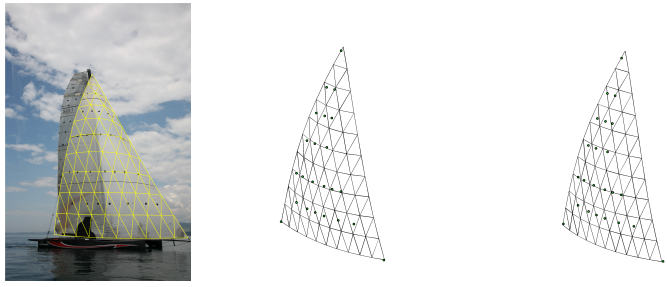
Here we present our results on real-images of surfaces made of three different materials, paper, cloth, and plastic.

The paper model is a triangulated mesh of size 11×8 which is parameterized by 20 regularly placed control points. Similarly, we model a t-shirt using a 7×10 mesh parametrized by 12 control points. Our reconstruction results for the paper and t-shirt sequences are depicted by Figs. 7 and 8, respectively.

Our final example involves a plastic ball that we cut in half, leaving the half shell shown in Fig. 9. We used our Kinect camera to produce the *concave* reference shape of Fig. 9. Our method can then be used to recover its shape from input images whether it is concave or *convex*. This demonstrates our method's ability to recover truly large non planar deformations.

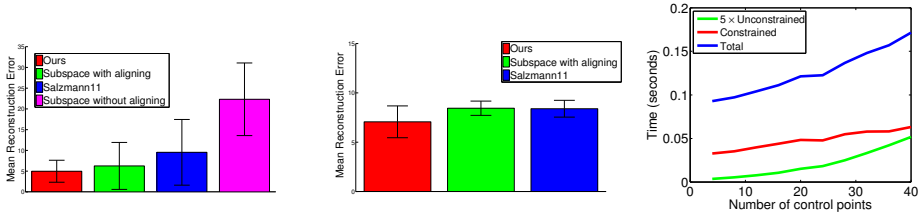
6 Conclusion

We have presented a novel approach to parameterizing the vertex coordinates of a mesh as a linear combination of a subset of them. In addition to reducing the



(a) (b) (c)

Fig. 5. Sail reconstructions (a) Reprojection of the unconstrained reconstruction on the input image. **(b)-(c)** Triangulations seen from a different view-point with the ground-truth points shown in green for unconstrained and constrained methods, respectively. Note that the two reconstructions are visually the same.



(a) (b) (c)

Fig. 6. Reconstruction errors for different methods on (a) paper sequence **(b)** sail images. We omit the results of the subspace method without aligning in **(b)** as it is a very high compared to the others in the plot. We are reporting all the distances in millimeters. **Computation times (c)** for C++ implementation with unconstrained outlier rejection (green), constrained optimization (red) and total time including keypoint matching and extraction (blue).

dimensionality of the monocular 3D shape recovery problem, it yields a rotation-invariant regularization term that lets us achieve good results without training data and or having to explicitly handle global rotations.

In our current implementation, the subset of vertices was chosen arbitrarily and we applied constraints on every single edge of the mesh. In future work, we will explore ways to optimally chose this subset and the constraints we enforce to further decrease computational complexity without sacrificing accuracy.

References

1. Salzmann, M., Fua, P.: Deformable Surface 3D Reconstruction from Monocular Images. Morgan-Claypool Publishers (2010)
2. Blanz, V., Vetter, T.: A Morphable Model for the Synthesis of 3D Faces. In: SIGGRAPH. (1999) 187–194

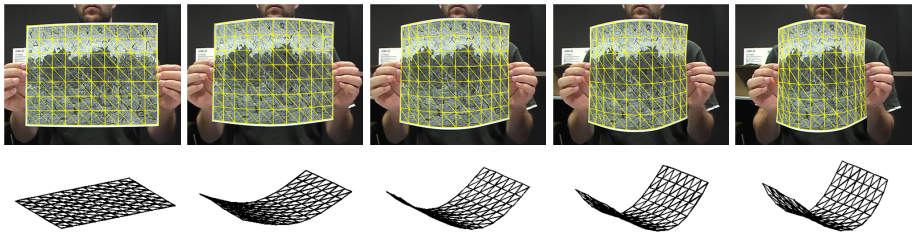


Fig. 7. Paper sequence. Top row: Reprojections of our reconstructions on the input images. **Bottom row:** Reconstructions seen from a different view-point. We supply the corresponding video-sequences as supplementary material.

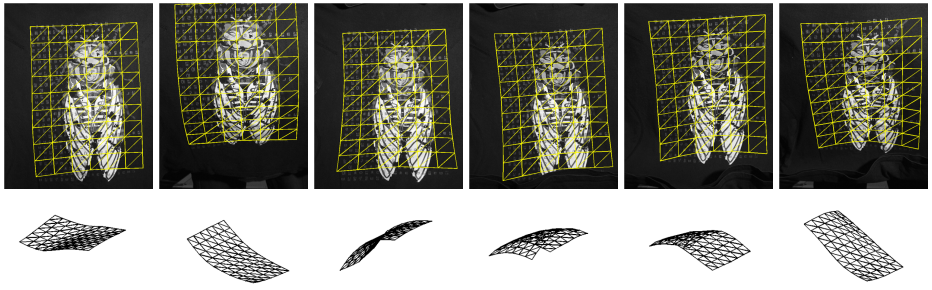


Fig. 8. T-shirt sequence. Top row: Reprojections of our reconstructions on the input images. **Bottom row:** Reconstructions seen from a different view-point. We supply the corresponding video-sequences as supplementary material.

3. Salzmann, M., Fua, P.: Linear Local Models for Monocular Reconstruction of Deformable Surfaces. PAMI **33** (2011) 931–944
4. Gumerov, N., Zandifar, A., Duraiswami, R., Davis, L.: Structure of Applicable Surfaces from Single Views. In: ECCV. (2004)
5. Liang, J., Dementhon, D., Doermann, D.: Flattening Curved Documents in Images. In: CVPR. (2005) 338–345
6. Perriollat, M., Bartoli, A.: A Quasi-Minimal Model for Paper-Like Surfaces. In: BenCos: Workshop Towards Benchmarking Automated Calibration, Orientation and Surface Reconstruction from Images. (2007)
7. Ecker, A., Jepson, A., Kutulakos, K.: Semidefinite Programming Heuristics for Surface Reconstruction Ambiguities. In: ECCV. (2008)
8. Shen, S., Shi, W., Liu, Y.: Monocular 3D Tracking of Inextensible Deformable Surfaces Under L2-Norm. In: ACCV. (2009)
9. Brunet, F., Hartley, R., Bartoli, A., Navab, N., Malgouyres, R.: Monocular Template-Based Reconstruction of Smooth and Inextensible Surfaces. In: ACCV. (2010)
10. White, R., Forsyth, D.: Combining Cues: Shape from Shading and Texture. In: CVPR. (2006)
11. Moreno-Noguer, F., Salzmann, M., Lepetit, V., Fua, P.: Capturing 3D Stretchable Surfaces from Single Images in Closed Form. In: CVPR. (2009)

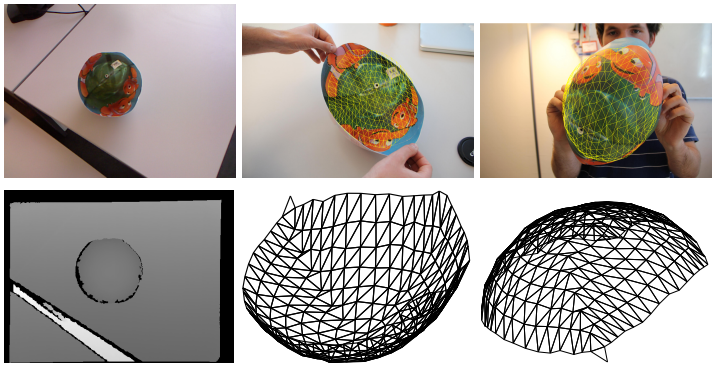


Fig. 9. Large deformations. Top row: The reference image with a concave reference shape and two input images. In the first, the surface is concave and in the other convex. **Bottom row:** The depth image used to build the reference mesh, reconstructions of the concave and convex versions of the surface.

12. Moreno-Noguer, F., Porta, J., Fua, P.: Exploring Ambiguities for Monocular Non-Rigid Shape Estimation. In: ECCV. (2010)
13. Perriollat, M., Hartley, R., Bartoli, A.: Monocular Template-Based Reconstruction of Inextensible Surfaces. IJCV **95** (2011)
14. Salzmann, M., Moreno-Noguer, F., Lepetit, V., Fua, P.: Closed-Form Solution to Non-Rigid 3D Surface Registration. In: ECCV. (2008)
15. Sorkine, O., Cohen-Or, D., Lipman, Y., Alexa, M., Rössl, C., Seidel, H.P.: Laplacian Surface Editing. In: Symposium on Geometry Processing. (2004)
16. Summer, R., Popovic, J.: Deformation Transfer for Triangle Meshes. TOG (2004)
17. Bue, A.D., Bartoli, A.: Multiview 3D Warps. In: ICCV. (2011)
18. Dimitrijević, M., Ilić, S., Fua, P.: Accurate Face Models from Uncalibrated and Ill-Lit Video Sequences. In: CVPR. (2004)
19. Ilić, S., Fua, P.: Using Dirichlet Free Form Deformation to Fit Deformable Models to Noisy 3D Data. In: ECCV. (2002)
20. Salzmann, M., Urtasun, R., Fua, P.: Local Deformation Models for Monocular 3D Shape Recovery. In: CVPR. (2008)
21. Terzopoulos, D., Witkin, A., Kass, M.: Symmetry-Seeking Models and 3D Object Reconstruction. IJCV **1** (1987) 211–221
22. Cohen, L., Cohen, I.: Finite-Element Methods for Active Contour Models and Balloons for 2D and 3D Images. PAMI **15** (1993)
23. Metaxas, D., Terzopoulos, D.: Constrained Deformable Superquadrics and Non-rigid Motion Tracking. PAMI **15** (1993)
24. Ozuysal, M., Calonder, M., Lepetit, V., Fua, P.: Fast Keypoint Recognition Using Random Ferns. PAMI **32** (2010)