Learning Spatially-Smooth Mappings in Non-Rigid Structure from Motion

Onur C. Hamsici[1], Paulo F.U. Gotardo[2], and Aleix M. Martinez[2]

[1]Qualcomm Research, San Diego, CA, USA
[2]The Ohio State University, Columbus, OH, USA
ohamsici@qualcomm.com,{gotardop,aleix}@ece.osu.edu

Abstract. Non-rigid structure from motion (NRSFM) is a classical underconstrained problem in computer vision. A common approach to make NRSFM more tractable is to constrain 3D shape deformation to be smooth over time. This constraint has been used to compress the deformation model and reduce the number of unknowns that are estimated. However, temporal smoothness cannot be enforced when the data lacks temporal ordering and its benefits are less evident when objects undergo abrupt deformations. This paper proposes a new NRSFM method that addresses these problems by considering deformations as spatial variations in shape space and then enforcing spatial, rather than temporal, smoothness. This is done by modeling each 3D shape coefficient as a function of its input 2D shape. This mapping is learned in the feature space of a rotation invariant kernel, where spatial smoothness is intrinsically defined by the mapping function. As a result, our model represents shape variations compactly using custom-built coefficient bases learned from the input data, rather than a pre-specified set such as the Discrete Cosine Transform. The resulting kernel-based mapping is a by-product of the NRSFM solution and leads to another fundamental advantage of our approach: for a newly observed 2D shape, its 3D shape is recovered by simply evaluating the learned function.

1 Introduction

Structure from motion (SFM) techniques have seen vast improvements over the past three decades by relying on the assumption of object rigidity [1]. However, computer vision applications often involve the observation of deformable objects such as the human face and body. When the assumption of object rigidity is relaxed, and in the absence of any prior knowledge on 3D shape deformation, computing non-rigid structure from motion (NRSFM) becomes a challenging, underconstrained problem. Given a set of corresponding 2D points, established over multiple images of a deformable object, the goal of NRSFM is to recover the object's 3D shape and 3D pose (relative camera position) in each image [2–15].

To make this largely underconstrained problem more tractable, recent research work has attempted to define new, general constraints for 3D shape deformation. A common approach to NRSFM is the matrix factorization method

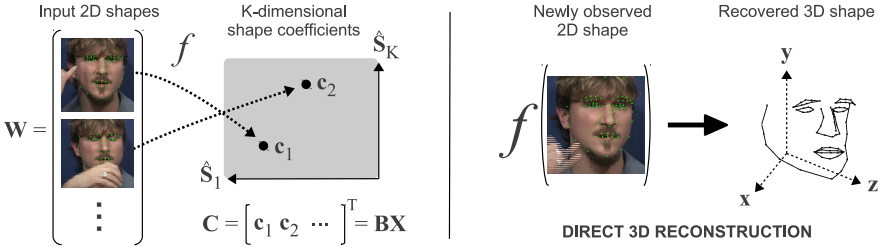

Fig. 1. Before solving NRSFM, a basis $\mathbf{B}$ is computed as to compactly represent a nonlinear mapping from the input data to the coefficients $\mathbf{C} = \mathbf{BX}$ of the linear shape model: (*left*) $\mathbf{B}$ is obtained by modeling each coefficient vector as a function $f(\cdot)$ of its input 2D shape; an RIK feature space is used to learn $f(\cdot)$ and $\mathbf{B}$ based on similarities in these input shapes; (*right*) with $f(\cdot)$ being a by-product of the NRSFM solution, the 3D reconstruction of a newly observed 2D shape is done by simply evaluating $f(\cdot)$.

of [2], which constrains all 3D shapes to lie within a low-dimensional linear shape space. In addition, many NRSFM techniques also enforce smoothness constraints on camera motion and object deformation, which are assumed to change only gradually over subsequent images [3, 6, 11–13, 15].

The recent Shape Trajectory Approach (STA) of [13], a generalization of [12], demonstrates how gradual 3D shape deformation can be seen as the smooth time-trajectory of a single point (object) within a low-dimensional shape space. As a result, a few low-frequency components of the Discrete Cosine Transform (DCT) can be used as basis vectors to define a compact representation of 3D shape deformation. Because the DCT basis is known *a priori*, the number of unknowns that need to be estimated is greatly reduced. STA has been shown to outperform a number of state-of-the-art NRSFM algorithms when applied to the 3D reconstruction of challenging datasets. However, it was also shown in [13] that sudden (high-frequency) deformations require the use of a large DCT basis, leading to less compact models. In addition, if the input 2D points come from a collection of images for which no temporal relation is known, the smoothness assumption does not hold and there is no gain in using the DCT basis.

This paper presents a novel NRSFM approach that addresses these problems by considering deformations as spatial variations in shape space and then enforcing spatial, rather than temporal, smoothness. Instead of using the DCT basis, we represent the coefficients of the linear shape model compactly using custom-built bases learned from the input data. These bases are obtained by expressing each 3D shape coefficient as a function of its input 2D shape, Fig. 1(left). This smooth function is learned in the feature space of a rotation invariant kernel (RIK) [16], in terms of the input data; more specifically, we learn a compact subspace using kernel principal component analysis (KPCA) [17]. The learned mapping becomes a by-product of our NRSFM solution and leads to another fundamental advantage of our approach: for a newly observed 2D shape, its 3D reconstruction is obtained via the simple evaluation of this function, Fig. 1(right).

Finally, we also propose a novel model fitting algorithm, based on iteratively-reweighted least squares (IRLS) [18], to extract local (sparse) modes of deformation – which are key features in applications that analyze 3D object deformation.

Our NRSFM model is derived in Section 3. Section 4 presents our IRLS-based algorithm, with experimental results in Section 5.

2 Related Work and Basic Formulation

We first summarize the notation used in the following: matrices and column vectors are denoted using upper-case and lower-case bold letters, respectively; $\mathbf{I}_n$ is the $n \times n$ identity matrix; $\mathbf{A} \otimes \mathbf{B}$ is the Kronecker product of two matrices; $\mathbf{A}^\dagger$ denotes the Moore-Penrose pseudo-inverse of $\mathbf{A}$; $\|\mathbf{A}\|_F$ is the Frobenius norm; $\mathbf{z}^*$ is the Hermitian of complex vector $\mathbf{z}$; and $\delta_{i,j}$ is the Kronecker delta.

For a NRSFM problem with T images (cameras), the n input 2D point tracks are given in an input matrix $\mathbf{W} \in \mathbb{R}^{2T \times n}$; $[x_{t,j}, y_{t,j}]^T$ is the 2D projection of the j^{th} 3D point observed on the t^{th} image, $t = 1, 2, \ldots, T$, $j = 1, 2, \ldots, n$. For clarity of presentation, assume for now that: (i) $\mathbf{W}$ is complete, meaning that no 2D points became occluded during tracking; and (ii) its mean column vector $\mathbf{t} \in \mathbb{R}^{2T}$ has been subtracted from all columns, making them zero-mean. With orthographic projection and a world coordinate system centered on the observed 3D object, $\mathbf{t}$ gives the observed 2D camera translations in each image.

The matrix factorization approach of [2] models $\mathbf{W} = \mathbf{MS}$ as a product of two matrix factors of low-rank $3K$, $\mathbf{M} \in \mathbb{R}^{2T \times 3K}$ and $\mathbf{S} \in \mathbb{R}^{3K \times n}$,

$$\underbrace{\begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,n} \\ y_{1,1} & y_{1,2} & \cdots & y_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{T,1} & x_{T,2} & \cdots & x_{T,n} \\ y_{T,1} & y_{T,2} & \cdots & y_{T,n} \end{bmatrix}}_{\mathbf{W}} = \underbrace{\begin{bmatrix} \widehat{\mathbf{R}}_1 & & & \\ & \widehat{\mathbf{R}}_2 & & \\ & & \ddots & \\ & & & \widehat{\mathbf{R}}_T \end{bmatrix}}_{\mathbf{D}} \left(\underbrace{\begin{bmatrix} c_{1,1} & \cdots & c_{1,K} \\ c_{2,1} & \cdots & c_{2,K} \\ \vdots & \ddots & \vdots \\ c_{T,1} & \cdots & c_{T,K} \end{bmatrix}}_{\mathbf{C}} \otimes \mathbf{I}_3 \right) \underbrace{\begin{bmatrix} \widehat{\mathbf{S}}_1 \\ \vdots \\ \widehat{\mathbf{S}}_K \end{bmatrix}}_{\mathbf{S}} \tag{1}$$

Factor $\mathbf{M} = \mathbf{D}\,(\mathbf{C} \otimes \mathbf{I}_3)$ comprises a block-diagonal rotation matrix $\mathbf{D} \in \mathbb{R}^{2T \times 3T}$ and a shape coefficient matrix $\mathbf{C} \in \mathbb{R}^{T \times K}$. Let $\mathbf{c}_t^T$ be the t^{th} row of $\mathbf{C}$. The unknown 3D shape of the t^{th} image is modeled as the matrix function

$$S(\mathbf{c}_t^T) = (\mathbf{c}_t^T \otimes \mathbf{I}_3)\mathbf{S} = \sum_{k=1}^{K} c_{t,k} \widehat{\mathbf{S}}_k, \tag{2}$$

that is, a *linear* combination of K basis shapes $\widehat{\mathbf{S}}_k \in \mathbb{R}^{3 \times n}$ as described by the shape coordinates $c_{t,k}$. The camera orientation (object pose) at image t is given by $\widehat{\mathbf{R}}_t \in \mathbb{R}^{2 \times 3}$, a 3D rotation followed by an orthographic projection to 2D.

The factors $\mathbf{M}$ and $\mathbf{S}$ are computed from the singular value decomposition (SVD) $\mathbf{W} = (\mathbf{U}\mathbf{\Sigma}^{\frac{1}{2}})(\mathbf{\Sigma}^{\frac{1}{2}}\mathbf{V}^T) = \overline{\mathbf{M}}\overline{\mathbf{S}}$, with all but the largest $3K$ singular values in $\mathbf{\Sigma}$ set to zero. This non-unique solution is defined only up to a rank-$3K$ ambiguity matrix $\mathbf{Q} \in \mathbb{R}^{3K \times 3K}$. To recover $\mathbf{D}$ and $\mathbf{C}$, an Euclidean upgrade step [11] finds a corrective $\mathbf{Q}$ for the solution $\mathbf{W} = (\overline{\mathbf{M}}\mathbf{Q})(\mathbf{Q}^{-1}\overline{\mathbf{S}}) = \mathbf{MS}$.

To further constrain the reconstruction process above, many authors assume that the observed 3D shape deformation is only gradual over time $t = 1, \ldots, T$ [3, 6, 12, 13]. Here, we summarize STA [13], which is closely related to our new method. STA considers $\mathbf{c}_t^T = c(t)$ as a single K-dimensional point describing a smooth time-trajectory within an unknown linear shape space. This means that each shape coordinate $c_{t,k}$ varies smoothly with t. The shape trajectory is then modeled compactly using a small number d of low-frequency DCT coefficients,

$$\mathbf{C} = \mathbf{\Omega}_d \left[\mathbf{x}_1, \ldots, \mathbf{x}_K\right] = \mathbf{\Omega}_d \mathbf{X}, \qquad \mathbf{x}_k \in \mathbb{R}^d. \tag{3}$$

With $d \ll T$, $\mathbf{X} \in \mathbb{R}^{d \times K}$ represents $\mathbf{C} \in \mathbb{R}^{T \times K}$ compactly in the domain of the truncated DCT basis matrix $\mathbf{\Omega}_d \in \mathbb{R}^{T \times d}$. The f^{th} column of $\mathbf{\Omega}_d$ is the f^{th}-frequency cosine wave [12, 13]. Because the DCT matrix is known *a priori*, the number of unknowns in $\mathbf{C}$ is significantly reduced with STA.

The optimization stage of STA considers that $\mathbf{S} = \mathbf{M}^\dagger \mathbf{W}$ is a function of $\mathbf{M}$ and $\mathbf{W}$. The goal is then to minimize the 2D reprojection error,

$$e(\mathbf{M}) = \|\mathbf{W} - \mathbf{W}^*\|_F^2, \quad \mathbf{W}^* = \mathbf{M}\mathbf{S} = \mathbf{M}\mathbf{M}^\dagger \mathbf{W}. \tag{4}$$

With $\mathbf{M} = \mathbf{D}(\mathbf{\Omega}_d \mathbf{X} \otimes \mathbf{I}_3)$, a coarse initial deformation model ($\mathbf{X} = \mathbf{I}_K$) [12] is first used to compute $\mathbf{D}$. Then higher-frequency DCT coefficients in $\mathbf{X}$ are estimated using a Gauss-Newton algorithm to minimize (4) in terms of $\mathbf{X}$ only.

3 NRSFM with RIKs

In this section, we propose a new kernel-based solution to NRSFM. Our goal is to derive a function that estimates the coefficient matrix $\mathbf{C}$ and is not restricted to cases of smooth deformations over time. As a result, we will also learn a custom-built basis $\mathbf{B}$ from the input data, providing a compact representation $\mathbf{C} = \mathbf{B}\mathbf{X}$. To this end, we first need to establish a relationship between $\mathbf{C}$ and the observed data in $\mathbf{W}$. More especially, we learn a function $f(\cdot)$ that estimates vector $\mathbf{c}_t^T$ – representing an unknown 3D shape as a point within the shape space – given the corresponding input 2D shape $\mathbf{w}^t \in \mathbb{R}^{2 \times n}$ observed on the t^{th} image,

$$\mathbf{c}_t^T = f(\mathbf{w}^t), \qquad \mathbf{w}^t = \begin{bmatrix} x_{t,1} & x_{t,2} & \ldots & x_{t,n} \\ y_{t,1} & y_{t,2} & \ldots & y_{t,n} \end{bmatrix}. \tag{5}$$

This mapping becomes a by-product of the NRSFM solution and leads to a fundamental advantage of our approach. Given a new image with a previously unseen 2D shape, the estimation of the corresponding 3D shape is readily achieved.

3.1 Defining a mapping using the kernel trick

Following the well-known kernel trick [17], we first consider a nonlinear mapping of each 2D shape $\mathbf{w}^t$ onto vector $\phi(\mathbf{w}^t)$, located within a high dimensional space where a final linear mapping can be learned. According to the Representer Theorem, the function $f(\cdot)$ that we seek can be expressed as a linear combination of a few representative $\phi(\mathbf{w}^t)$. Thus, we can model the k^{th} coefficient of $\mathbf{c}_t^T$ as

$$c_{t,k} = f_k(\mathbf{w}^t) = \sum_{i=1}^{d} \phi(\mathbf{w}^t)^T \phi(\mathbf{w}_b^i) x_{ik} \tag{6}$$

where x_{ik} are the coefficients of a linear combination of a few 2D basis shapes, $\mathbf{w}_b^i$. The number of basis elements d must be sufficient as to represent the relations between $\mathbf{C}$ and $\mathbf{W}$, as discussed below.

In general, explicitly evaluating the mapping $\phi(\cdot)$ can be computationally expensive or even impossible when the image is a function in an infinite dimensional space. Thus, we perform this mapping only implicitly by embedding it in the computation of a generalized inner product given by a kernel function $\kappa(\cdot,\cdot)$,

$$c_{t,k} = f_k(\mathbf{w}^t) = \sum_{i=1}^{d} \kappa(\mathbf{w}^t, \mathbf{w}_b^i) x_{ik}. \tag{7}$$

The kernel function above must provide a similarity measure for two 2D shapes observed from different points of view (*i.e.*, poses); its proper definition is discussed in Section 3.3. Considering all K coefficients of $\mathbf{c}_t^T$, $\forall t$, from (7) we obtain

$$\mathbf{C} = \mathbf{\Phi}(\mathbf{W})^T \mathbf{\Phi}(\mathbf{W}_b)\mathbf{X} = \mathbf{K}_{\mathbf{W}\mathbf{W}_b}\mathbf{X} \stackrel{\text{def}}{=} \mathbf{B}\mathbf{X}, \tag{8}$$

where $\mathbf{X} \in \mathbb{R}^{d\times K}$ is a coefficient matrix; $\mathbf{B} \stackrel{\text{def}}{=} \mathbf{K}_{\mathbf{W}\mathbf{W}_b} \in \mathbb{R}^{T\times d}$ is a custom-built basis matrix that has the inner product values for all pairings of a 2D shape in $\mathbf{W}$ and a 2D basis shape in $\mathbf{W}_b$.

Unfortunately, selecting the best set of basis shapes with d out of the T observed 2D shapes is an NP-complete problem. We therefore define a simple, alternative solution based on kernel principal component analysis (KPCA) [17]. We first pre-compute a complete kernel matrix $\mathbf{K}_{\mathbf{W}\mathbf{W}} \in \mathbb{R}^{T\times T}$ and its eigenvector matrix $\mathbf{V}$ associated with the d largest eigenvalues in the diagonal matrix $\mathbf{\Lambda}$, *i.e.*, $\mathbf{K}_{\mathbf{W}\mathbf{W}}\mathbf{V} = \mathbf{V}\mathbf{\Lambda}$. In the range space of mapping $\phi(\cdot)$, we have d eigenfunctions given by $\mathbf{\Phi}(\mathbf{W})\mathbf{V}\mathbf{\Lambda}^{-1/2}$. By projecting each observation $\phi(\mathbf{w}^t)$ onto this eigenfunction subspace, we can then define our new basis matrix $\mathbf{B}$ of $\mathbf{C}$ as,

$$\mathbf{C} = \mathbf{\Phi}(\mathbf{W})^T \mathbf{\Phi}(\mathbf{W})\mathbf{V}\mathbf{\Lambda}^{-1/2}\mathbf{X} = \underbrace{\mathbf{K}_{\mathbf{W}\mathbf{W}}\mathbf{V}\mathbf{\Lambda}^{-1/2}}_{\mathbf{B}}\mathbf{X}. \tag{9}$$

The number of eigenfunctions d must be large enough as to provide a subspace that captures a sufficient amount of the variation in the kernel matrix.

Finally, we obtain our new NRSFM model with $\mathbf{M} = \mathbf{D}(\mathbf{B}\mathbf{X} \otimes \mathbf{I}_3)$. A solution is achieved by estimating the rotation matrix $\mathbf{D}$ and the $d \times K$ coefficient matrix $\mathbf{X}$ as to minimize the reprojection error in (4). This optimization procedure is detailed in Section 4. Once the optimal $\mathbf{M}$ and $\mathbf{S} = \mathbf{M}^\dagger\mathbf{W}$ have been found, we can use (2) to recover the 3D shape for the t^{th} image as

$$S(\mathbf{c}_t^T) = S(f(\mathbf{w}^t)) = \left(f(\mathbf{w}^t) \otimes \mathbf{I}_3\right)\mathbf{M}^\dagger\mathbf{W}, \quad \text{with} \tag{10}$$

$$\mathbf{c}_t^T = f(\mathbf{w}^t) = \boldsymbol{\kappa}(\mathbf{w}^t, \mathbf{W})\mathbf{V}\mathbf{\Lambda}^{-1/2}\mathbf{X}, \quad \text{and} \tag{11}$$

$$\boldsymbol{\kappa}(\mathbf{w}^t, \mathbf{W}) = \left[\ \kappa(\mathbf{w}^t, \mathbf{w}^1)\ \kappa(\mathbf{w}^t, \mathbf{w}^2)\ \ldots\ \kappa(\mathbf{w}^t, \mathbf{w}^T)\ \right]. \tag{12}$$

This new approach is referred to as NRSFM with RIKs.

3.2 Recovering the 3D shape from a newly seen 2D shape

An important advantage of NRSFM with RIKs is the ability to easily reconstruct the 3D shape from a newly observed image that was not considered in the optimization above; let $\mathbf{w}^\tau$ $(\tau > T)$ denote this newly observed 2D shape. Notice from (10) that the 3D shape $S(f(\mathbf{w}^\tau))$ associated with $\mathbf{w}^\tau$ can be easily estimated given $\mathbf{W}$, $\mathbf{M}$, $f(\cdot)$ from the optimization above.

Once the 3D shape has been recovered, the associated rotation (pose) matrix $\widehat{\mathbf{R}}_\tau$ can also be readily estimated by solving two simple systems of linear equations, $\mathbf{w}^\tau = \widehat{\mathbf{R}}_\tau\ S(f(\mathbf{w}^\tau))$, then using the SVD of $\widehat{\mathbf{R}}_\tau$ to enforce orthogonality.

3.3 Rotation invariant kernels

The kernel function must provide a similarity measure for two 3D shapes based on their 2D projections, $\mathbf{w}^t$ and $\mathbf{w}^{t'}$, taken from different points of view. One possible choice of the kernel function $\kappa(\cdot,\cdot)$ is the RIK of [16]. This RIK calculates the rotation invariant similarity between two scale-normalized 2D shapes represented in the complex domain, vectors $\mathbf{z}_t$ and $\mathbf{z}_{t'}$ with $\mathbf{z}_t^*\mathbf{z}_t = \mathbf{z}_{t'}^*\mathbf{z}_{t'} = 1$,

$$\kappa(\mathbf{z}_t, \mathbf{z}_{t'}) = \exp\left(\frac{-1 + |\mathbf{z}_t^*\mathbf{z}_{t'}|}{\sigma^2}\right), \quad \mathbf{z}_t = \frac{(\mathbf{w}^t)^T}{\|\mathbf{w}^t\|_F}\begin{bmatrix}1\\ \sqrt{-1}\end{bmatrix} \in \mathbb{C}^n. \tag{13}$$

The scale (smoothness) of this RIK is defined by parameter σ. The 2D rotation invariance property ensures that $k(\mathbf{z}_t, \mathbf{z}_{t'}) = k(e^{\theta\sqrt{-1}}\mathbf{z}_t, \mathbf{z}_{t'})$ for any rotation angle θ in the complex plane. This is a property of the inner product in the complex domain. Although the kernel above is not invariant to the 3D orientation of the observed shapes, we can still use it to learn the mapping in (11) because the input 2D shapes are highly correlated with the underlying 3D shapes – we can even use *appearance features* that are correlated with 3D shape [19].

Here we also propose a new kernel dubbed the *affine structure from motion* (aSFM) kernel. The aSFM RIK is defined in terms of the reprojection error $r_{t,t'}^2$ of an affine, rigid SFM solution obtained from the two observations $\mathbf{w}^t$ and $\mathbf{w}^{t'}$,

$$\kappa(\mathbf{w}^t, \mathbf{w}^{t'}) = \exp\left(\frac{-r_{t,t'}^2}{\sigma^2}\right) + \alpha\delta_{t,t'}, \quad r_{t,t'} = \left\| \begin{bmatrix}\mathbf{w}^t\\ \mathbf{w}^{t'}\end{bmatrix} - \begin{bmatrix}\mathbf{A}^t\\ \mathbf{A}^{t'}\end{bmatrix}\mathbf{S}_a \right\|_F \tag{14}$$

where σ is the kernel scale and parameter α regulates how similar the 3D shapes are in general, while also ensuring that the kernel matrix is positive semi-definite. The affine cameras $\mathbf{A}^t$ and $\mathbf{A}^{t'} \in \mathbb{R}^{2\times 3}$ and the affine 3D shape $\mathbf{S}_a \in \mathbb{R}^{3\times n}$ are obtained from a rank-3 approximation to $\mathbf{w}^t$ and $\mathbf{w}^{t'}$ using SVD. If these 2D shapes are projections of two dissimilar 3D shapes, then the rigid SFM solution will provide a large reprojection error and the aSFM kernel value will be small.

3.4 Model Analysis

Parameter setting: With the rank parameter K assumed to be known, the number of unknowns in $\mathbf{X} \in \mathbb{R}^{d\times K}$ depends on the number of columns d of

$\mathbf{B} \in \mathbb{R}^{T\times d}$. A rank-$3K$ solution $\mathbf{M}$ requires $d \geq K$. If $d = K$, then $\mathbf{X}$ must be full-rank (*i.e.*, $\mathbf{X}^{-1}$ exists) and the non-unique solution $\mathbf{M} = \mathbf{D}(\mathbf{BX} \otimes \mathbf{I}_3)$ has an equivalent form $\overline{\mathbf{M}} = \mathbf{M}(\mathbf{X}^{-1} \otimes \mathbf{I}_3) = \mathbf{D}(\mathbf{B}\mathbf{I}_K \otimes \mathbf{I}_3)$, with a constant $\overline{\mathbf{X}} = \mathbf{I}_K$. By assuming $d > K$, we allow the rank-$3K$ solution to consider other important variations in the kernel matrix, leading to better results.

The discussion above suggests a deterministic initialization $\mathbf{X}_0 = [\ \mathbf{I}_K\ \mathbf{0}\ ]^T$ in which the coefficients associated with less important principal components are initially zero. Not surprisingly, the same initialization is used in STA, with high-frequency DCT coefficients set to zero. Note that the DCT and PCA bases are known to be closely related for certain types of random processes.

To select d, a common approach in PCA is to choose a d-dimensional subspace that captures about 99% of the total variance in the dataset, discarding small variations assumed as noise. In NRSFM with RIKs, we note that d is closely related to the RIK scale parameter σ: the larger σ is, the more smoothness is applied to the shape similarity values and the more compact is the KPCA space. Therefore, we consider d as a user-supplied parameter that defines the desired compactness of the model; then σ is easily chosen, automatically, as to yield a d-dimensional KPCA space with about 99% of the data variance. In the aSFM RIK, α is also set automatically as to yield a positive semi-definite kernel matrix.

Comparison to related work: There are two main differences between our model above and that of the kernel NRSFM approach in [15]. First, NRSFM with RIK models 3D shapes within a linear space; the approach in [15] defines a non-linear model. Second, in NRSFM with RIK the inputs to the kernel-based mapping are observed 2D shapes; in [15], the inputs are the coefficients of the non-linear model. Nevertheless, the two approaches are complementary: future work can use RIKs to define a mapping from observed 2D shapes onto the coefficients of the non-linear model of [15].

4 Model Fitting

Having obtained the basis matrix $\mathbf{B}$ through an RIK and KPCA, as described above, the next step is to estimate $\mathbf{D}$ and $\mathbf{X}$ in $\mathbf{M} = \mathbf{D}(\mathbf{BX} \otimes \mathbf{I}_3)$ as to minimize the reprojection error in (4). Two alternative algorithms are presented in this section. Here, we will assume that the rotation matrix $\mathbf{D}$ has been estimated by an initialization algorithm (*e.g.*, using rigid SFM if some points are known to remain in a rigid configuration, or using the procedure of STA). Thus, we focus on the iterative process for fitting our new model $\mathbf{C} = \mathbf{BX}$. If necessary, we can later refine $\mathbf{D}$ and $\mathbf{X}$ in an alternated manner, by fixing one of these matrices.

Algorithm 1 (NRSFM with RIK): We first consider an optimization procedure in which the computation of $\mathbf{X}$ is carried out using the iterative Gauss-Newton method proposed in [14], with the DCT basis replaced by our new basis $\mathbf{B}$. This procedure is summarized in Algorithm 1.

Algorithm 2 (Iteratively-Reweighted NRSFM with RIK): With a linear shape model, the 3D shape of a non-rigid object can be seen as comprising two

main components: a rigid (average) 3D shape and $K-1$ modes of deformation. For typical objects, these modes should reflect localized (sparse) deformations involving a small subset of points (sub-shapes). Also, different parts of an object often present different amounts of deformation. For instance, consider facial shapes that present larger deformation for the mouth in comparison to the nose; other shapes may even present points that remain in a rigid configuration.

NRSFM algorithms in general estimate shape deformation using a globally uniform least squares criterion; the objective function is automatically tuned to points with large deformation and is not sensitive to local deformations. Furthermore, the global solution does not allow for the modeling of local deformations with different complexities (ranks). This usually results in an inaccurate extraction of the rigid component and associated modes of deformation.

To address these problems, we propose a new method based on iteratively-reweighted least squares (IRLS) [18]. The algorithm iteratively minimizes the residual error resulting from Algorithm 1 above. The initial step extracts the rigid shape component of the observed object; the following steps are targeted at modeling localized modes of deformation. While IRLS has been used to implement robustness against outliers (whose errors are allowed to remain large), our goal here is to focus on columns that have a similar error pattern, corresponding to a mode of deformation that was not yet reconstructed properly.

More specifically, let $\mathbf{W} \approx \mathbf{M}_1\mathbf{S}_1$ be the output of Algorithm 1 with $K=1$. The single 3D basis shape recovered in iteration 1 describes the rigid component of the object shape. Next, we calculate the error matrix $\mathbf{E}_1 = \mathbf{W} - \mathbf{M}_1\mathbf{S}_1$ whose columns capture modes of shape deformation. To extract local (sub-shape) deformation, we focus on a subset of the columns of $\mathbf{E}_1$ corresponding to 2D points with similar motion. This is done by specifying a weight matrix that emphasizes columns (points) with a similar pattern of error (deformation). Let $\mathbf{e}_{i,j}$ be the j^{th} column of $\mathbf{E}_i$ (in the i^{th} iteration). We then define a Gaussian weighting mask with nonzero diagonal elements,

$$\mathbf{G}_i(j,j) = \exp\left(-\frac{\|\mathbf{e}_{i,j} - \mathbf{e}_{i,j_{max}}\|_2^2}{\sigma_e^2}\right), \quad j_{max} = \arg\max_j \|\mathbf{e}_{i,j}\|_2 \qquad (15)$$

where σ_e^2 is the average distance between $\mathbf{e}_{i,j_{max}}$ and its $0.1n$ (10%) nearest neighbors. This mask $\mathbf{G}_i$ assigns weight 1 to the column with largest error, $\mathbf{e}_{i,j_{max}}$, and slightly smaller weights to other similar columns. It is used to project $\mathbf{E}_i$ onto a subspace of large error, $\widetilde{\mathbf{E}}_1 = \mathbf{E}_1\mathbf{G}_1$.

The following iterations uses Algorithm 1 to factorize $\widetilde{\mathbf{E}}_i \approx \mathbf{M}_{i+1}\mathbf{S}_{i+1}$, always using $K=1$. The error matrix $\mathbf{E}_i$ is updated and the iterations continue until the error $\|\mathbf{E}_i\|_F$ is sufficiently small. Note that rotation matrix $\mathbf{D}$ remains constant during this iterative process and, therefore, the recovered deformation components are aligned in 3D space. The Iteratively-Reweighted NRSFM with RIK algorithm is summarized in Algorithm 2.

To recover the 3D shape for a new image whose 2D shape $\mathbf{w}^\tau$ has now being detected, we now follow the iterative procedure in Algorithm 3. Each iteration estimates the coefficient $c_{\tau,i} = f_i(\mathbf{w}^\tau)$ associated with the i^{th} 3D basis shape $\mathbf{S}_i$.

Algorithm 1 NRSFM with RIK

Input: 2D shapes in $\mathbf{W}$, basis size d, rank parameter K.
Compute the RIK matrix $\mathbf{K}_{\mathbf{WW}}$ with σ^2 and α as described in the text.
Find d-dimensional KPCA subspace with 99% of data variance.
Define basis matrix $\mathbf{B}$ as in Eq.(9).
Estimate rotation matrix $\mathbf{D}$.
Estimate $d \times K$ matrix $\mathbf{X}$ s.t. $\mathbf{M} = \mathbf{D}(\mathbf{B}\mathbf{X} \otimes \mathbf{I}_3)$ minimizes Eq.(4).
Refine $\mathbf{D}$ and $\mathbf{X}$ in alternation as to minimize Eq.(4).
Output: $\mathbf{D}$, $\mathbf{B}$, $\mathbf{X}$, and $f(\cdot)$ as in Eq.(11).

Algorithm 2 Iteratively-Reweighted NRSFM with RIK

Input: 2D shapes in $\mathbf{W}$, basis size d, rank parameter $K = 1$, level of accuracy ϵ.
Initialize $i = 0$, $\mathbf{E}_0 = \mathbf{W}$, and $\mathbf{G}_0 = \mathbf{I}_n$.
repeat
    Calculate projected error matrix $\widetilde{\mathbf{E}}_i = \mathbf{E}_i \mathbf{G}_i$.
    Compute the factorization $\widetilde{\mathbf{E}}_i \approx \mathbf{M}_{i+1}\mathbf{S}_{i+1}$ using Algorithm 1.
    Update the error matrix $\mathbf{E}_{i+1} = \mathbf{E}_i - \mathbf{M}_{i+1}\mathbf{S}_{i+1}$.
    Calculate the weighting mask $\mathbf{G}_{i+1}$ as in Eq.(15).
    $i = i + 1$.
until $\|\mathbf{E}_i\|_F < \epsilon$
Compute the final, recovered 3D shapes as $\mathbf{S}_{3D} = \sum_i (\mathbf{B}_i \mathbf{X}_i \otimes \mathbf{I}_3)\, \mathbf{S}_i$.
Output: $\mathbf{S}_{3D}$, $\mathbf{D}$, $\mathbf{B}_i$, $\mathbf{X}_i$, $\mathbf{S}_i$, and $f_i(\cdot)$.

Algorithm 3 Iterative 3D Reconstruction for a newly seen 2D shape

Input: newly observed 2D shape $\mathbf{w}^\tau$.
for $i = \{1, \ldots, N\}$ **do**
    Restore $\mathbf{S}_i$, and $f_i(\cdot)$, as previously computed with Algorithm 2.
    Evaluate $c_{\tau,i} = f_i(\mathbf{w}^\tau) = \boldsymbol{\kappa}_i(\mathbf{w}^\tau \mathbf{G}_i, \widetilde{\mathbf{E}}_i)\mathbf{V}_i \boldsymbol{\Lambda}_i^{-1/2} \mathbf{X}_i$.
    Update the current 3D shape estimate, $S(\mathbf{c}_\tau^T) = \sum_{l \leq i} c_{\tau,l} \mathbf{S}_l$
    Update the 3D pose matrix $\mathbf{R}_\tau$ s.t. $\mathbf{w}^\tau \approx \mathbf{R}_\tau S(\mathbf{c}_\tau^T)$.
    Compute the 2D error $\mathbf{w}^\tau = \mathbf{w}^\tau - \mathbf{R}_\tau S(\mathbf{c}_\tau^T)$.
end for
Output: shape coefficients $\mathbf{c}_\tau^T$, 3D pose $\mathbf{R}_\tau$, and 3D shape $S(\mathbf{c}_\tau^T)$.

5 Experimental Results

We evaluate the proposed methods in three different applications. First, we compare the solutions of NRSFM with RIK against those of STA with its fixed DCT basis (see [14] for a comparison of STA against other NRSFM methods). Second, we provide experiments that show the generalization performance of our NRSFM solutions to newly seen 2D shapes. Finally, we illustrate and analyze the local modes of deformation extracted with Algorithm 2. Additional results are also available with the supplementary material at *http://cbcsl.ece.ohio-state.edu*.

We consider a variety of motion capture 3D datasets, with the number of frames and 3D points indicated as (T, n) after the dataset name: *face1* (74,37) [9]; *stretch* (370,41), *pick-up* (357,41), *yoga* (307,41), *dance* (264,75) [12]; and *walking* (260,55) [3]. The input $\mathbf{W}$ is obtained via 2D orthographic projection.

NRSFM with RIK *versus* STA: Temporal smoothness, enforced by STA, does not hold when the observed shape undergoes abrupt deformation, or when the data lacks temporal ordering. NRSFM with RIK does not suffer such limitations because it enforces spatial smoothness of $f(\cdot)$ in the RIK space. From (7),

Table 1. Average 3D error (standard deviation) of NRSFM solutions on temporally ordered and randomly permuted (π) datasets. Parameters (d, K or N) are also shown.

Algorithm	face1	stretch	pick-up	yoga	dance	walking
STA	0.056 (0.037)	0.068 (0.043)	0.228 (0.176)	0.147 (0.119)	0.172 (0.171)	0.105 (0.141)
A1	0.067 (0.041)	0.087 (0.062)	0.229 (0.175)	0.150 (0.120)	0.174 (0.164)	0.133 (0.203)
A1_{aSFM}	0.069 (0.049)	0.086 (0.053)	0.231 (0.173)	0.152 (0.120)	0.173 (0.163)	0.104 (0.120)
A2	0.063 (0.050)	0.118 (0.103)	0.231 (0.164)	0.163 (0.129)	0.212 (0.223)	0.180 (0.230)
A2_{aSFM}	0.084 (0.059)	0.120 (0.089)	0.223 (0.158)	0.168 (0.125)	0.215 (0.237)	0.177 (0.248)
STA^{π}	0.130 (0.098)	0.384 (0.346)	0.424 (0.281)	0.366 (0.303)	0.396 (0.312)	0.323 (0.445)
A1^{π}	0.067 (0.041)	0.087 (0.062)	0.229 (0.175)	0.150 (0.120)	0.174 (0.164)	0.133 (0.203)
$\text{A1}^{\pi}_{\text{aSFM}}$	0.069 (0.049)	0.086 (0.053)	0.231 (0.173)	0.152 (0.120)	0.173 (0.163)	0.104 (0.120)
A2^{π}	0.063 (0.050)	0.118 (0.103)	0.231 (0.164)	0.163 (0.129)	0.212 (0.223)	0.180 (0.230)
$\text{A2}^{\pi}_{\text{aSFM}}$	0.084 (0.059)	0.120 (0.089)	0.223 (0.158)	0.168 (0.125)	0.215 (0.237)	0.177 (0.248)
STA (d, K)	$0.3T$, 5	$0.1T$, 8	$0.1T$, 3	$0.1T$, 7	$0.1T$, 7	$0.3T$, 5
A1 (d, K)	$0.3T$, 5	$0.2T$, 8	$0.2T$, 3	$0.2T$, 7	$0.2T$, 7	$0.2T$, 5
A1_{aSFM}	$0.3T$, 5	$0.2T$, 8	$0.2T$, 3	$0.2T$, 7	$0.1T$, 7	$0.1T$, 5
A2 (d, N)	$0.4T$, 26	$0.3T$, 26	$0.3T$, 26	$0.3T$, 26	$0.2T$, 26	$0.1T$, 26
A2_{aSFM}	$0.3T$, 26	$0.1T$, 26	$0.1T$, 26	$0.1T$, 26	$0.1T$, 26	$0.2T$, 26

Table 2. Average 3D error (standard deviation) of new shapes using cross-validation.

Algorithm	face1	stretch	pickup	yoga	dance	walking
A1	0.098 (0.101)	0.090 (0.059)	0.233 (0.174)	0.160 (0.125)	0.179 (0.180)	0.108 (0.123)
A2	0.125 (0.080)	0.126 (0.110)	0.245 (0.166)	0.167 (0.128)	0.216 (0.232)	0.278 (0.299)

Fig. 2. Reconstruction errors of A2 versus the number of iterations: 2D RIK (left) and aSFM RIK (right). Final reconstructions are obtained with approximately 15 iterations.

note that the same function $f(\cdot)$ can be learned regardless of the temporal order of the input 2D shapes. The following experiment illustrates this property.

STA, Algorithm 1 (A1), and Algorithm 2 (A2) are first used to reconstruct 3D shapes from temporally ordered 2D shapes $\mathbf{w}^t$ in $\mathbf{W}$. Then, 3D reconstructions are computed from an unordered matrix $\mathbf{W}^{\pi}$, obtained with a random permutation $\pi(t)$ of the input 2D shapes. To focus on the evaluation of the different 3D shape models, all algorithms are run with the same rotation matrix, $\mathbf{D}$ or $\mathbf{D}^{\pi}$, obtained from the original $\mathbf{W}$ as in [11].

Table 1 shows the 3D reconstruction error for each algorithm – *i.e.*, average Euclidean distance to the 3D points of the ground truth shapes, normalized by average shape size [13]. Note that the performance of the RIK-based methods is unaffected by permutations in the input data, while the performance of STA decreases significantly. When temporal smoothness holds, the three algorithms show similar performance, with compact solutions (small d). The similar performance presented by the aSFM and the 2D RIK shows that the 2D RIK

adequately captures shape variations in the input data. Overall, the aSFM RIK often leads to more compact solutions while the 2D RIK is faster to evaluate. Table 1 shows the best results of STA and A1 with $K = 1, 2, \ldots, 26$. While the results of NRSFM methods in general degenerate as K increases (*i.e.*, as the low-rank constraint is gradually relaxed), the reconstructions obtained with the IRLS-based A2 are less sensitive to the choice of this parameter. Fig. 2 shows that the solutions of A2 on each dataset stabilized after approximately 15 iterations. A2 also computes sparse modes of deformation with more meaningful information to computer vision applications, as discussed later in this section.

Reconstruction of newly observed 2D shapes: Another key advantage of NRSFM with RIKs is the capability of recovering 3D shapes of newly observed 2D shapes using the learned function $f(\cdot)$. Considering this scenario, we illustrate the performance of A1 and A2 using 30-fold cross-validation: the 2D shapes in $\mathbf{W}$ are randomly permuted and divided into 30 validation sets. In each fold, one validation set $\mathcal{S}_{\mathbf{W}}$ with nearly 3% of the 2D shapes is left out of the input data $\mathbf{W}$ and $f(\cdot)$ is learned from the remaining 2D shapes, with (d, K or N) set as in Table 1. Then the 3D reconstruction of each 2D shape $\mathbf{w}^\tau \in \mathcal{S}_{\mathbf{W}}$ is obtained using (10) or Algorithm 3. This process is repeated for each validation set. The average 3D error of all these reconstructions is shown in Table 2, for each dataset. These errors are similar to those obtained on the complete datasets (Table 1), indicating that the learned functions correctly reconstructed the new 2D shapes.

We also performed a similar experiment using 2D face shapes of a single person, taken from the real video sequence *ASL* (114,77) of [14]. First, A1_{aSFM} ($K = 4, d = 0.3T$) was used to recover the 3D shapes of all 114 input 2D faces, Fig. 3(left). Then a second 3D reconstruction was computed for each 2D shape, this time using 30-fold cross-validation as above. Comparing these two sets of 3D shapes, we observed a very small average 3D difference of 0.025 (0.034), relative to the average face size. As an additional experiment, we also evaluated the learned $f(\cdot)$ on input 2D shapes from a separate dataset with faces of the same person and also faces of other people. This is an example application in transfer of facial expression across subjects, which is very useful in computer graphics and animation. Note that, in cases of occlusion, the kernel is evaluated only on the subset of points that are observed on both 2D shapes being compared. Fig. 3(right) shows that the recovered 3D shapes do capture the learned deformations even when expressed by other people. As expected, the recovered 3D shapes can only express the identity and modes of deformation learned during the NRSFM (training) stage, using the data illustrated in Fig. 3(left). Nevertheless, this is not a limitation of our approach because, with the removal of the temporal smoothness assumption, the NRSFM stage can consider multiple datasets depicting different identities and shape variations (deformations). Naturally, if the newly observed 2D shapes differ considerably from the training shapes, 3D reconstruction may be inaccurate due to the limitations of the shape model when used for extrapolation. Future work will develop this capability further, considering new constraints such as ensuring $\mathbf{c}^\tau = f(\mathbf{w}^\tau)$ remains in the vicinity of the training samples within the learned shape space.

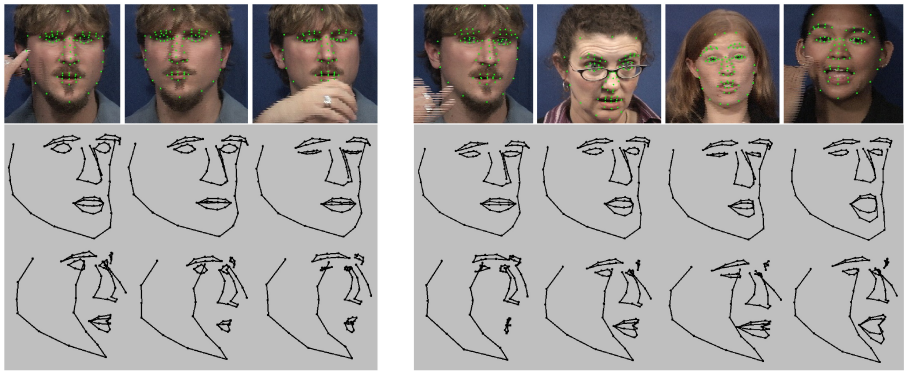

Fig. 3. Using the mapping $f(\cdot)$ learned from a real dataset: (*left*) sample 2D face shapes (green dots) of a same person and NRSFM solution of A1_{aSFM}, in two views; (*right*) result of evaluating the learned $f(\cdot)$ on newly seen 2D face shapes from different people.

Recovered modes of local 3D deformation: A limitation of most kernel methods is the use of a unique parameter σ, defining the smoothness of the estimated function globally. The Gaussian weighting masks $\mathbf{G}_i$ of A2 can be seen as altering (customizing) σ for each column on the input error matrix $\mathbf{E}_i$. This is important in NRSFM because the observed objects often present localized deformations with different spatial smoothness (*e.g.*, mouth shapes of a talking face present larger variations than nose shapes). A2 can model these local deformations by extracting a set of functions that correspond to sub-shape variations. The property described above is illustrated by the extracted modes of deformation shown in Fig. 4. For *face1*, the local deformation $\mathbf{S}_2$ represents mouth opening and closing (correlated with chin movement), $\mathbf{S}_3$ eye-nose distance, $\mathbf{S}_4$ right side jaw, $\mathbf{S}_5$ left side jaw, and $\mathbf{S}_6$ chin movements. For *stretch*, the deformations are: $\mathbf{S}_2$ left arm, $\mathbf{S}_3$ right arm, $\mathbf{S}_4$ head and waist, $\mathbf{S}_5$ right hand, and $\mathbf{S}_6$ left hand movements. In comparison to the standard model in NRSFM, the basis shapes above describe more meaningful, local deformations that can be combined in different ways as to better extrapolate new 3D shapes. Future work on A2 will explore this fact to further improve the generalization of the learned function $f(\cdot)$ to shapes largely different than those seen in the NRSFM stage.

6 Conclusion

We propose a new kernel-based solution to NRSFM that is not restricted to cases of smooth deformations over time. The main idea is to use a spatial, rather than temporal, smoothness constraint. Using a RIK and KPCA, we derive a smooth function that outputs 3D shape coefficients directly from an input 2D shape. As a result, we learn a custom-built basis to model the shape coefficient compactly while solving NRSFM. The learned mapping becomes a by-product of our NRSFM solution and leads to another fundamental advantage of our approach: for a newly observed 2D shape, its 3D reconstruction is obtained via

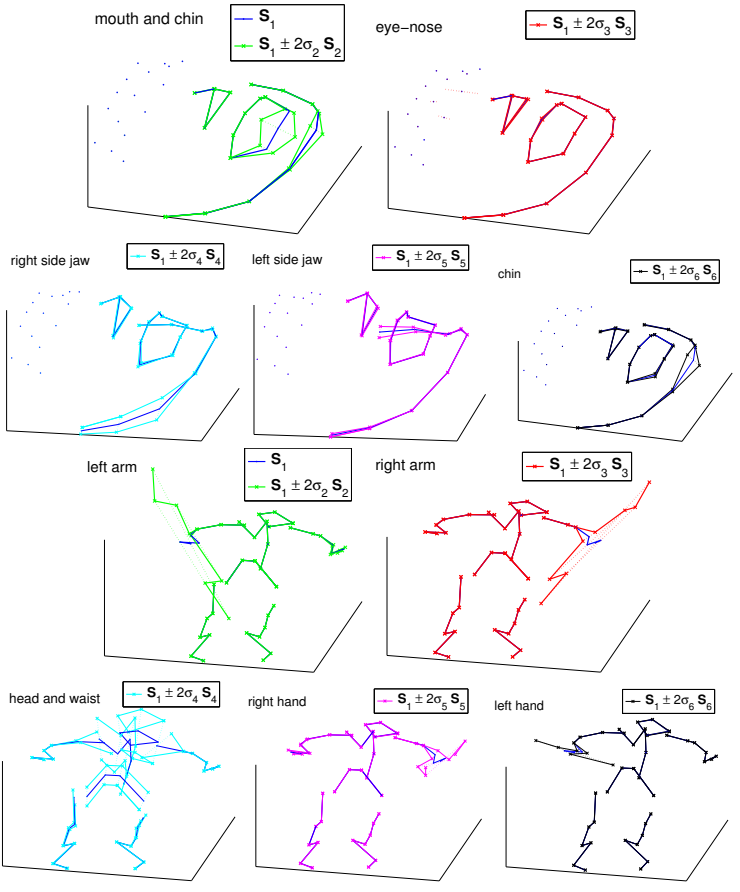

Fig. 4. The 3D shape bases obtained in the first 6 iterations of Algorithm 2 on *face1* and *stretch*. The 3D basis shapes $\mathbf{S}_2, ..., \mathbf{S}_6$ correspond to sub-shape deformations around the rigid shape component $\mathbf{S}_1$ of the first iteration. These deformations are shown as $\mathbf{S}_1 \pm 2\sigma_i \mathbf{S}_i$, with σ_i the standard deviations of the corresponding coefficients. Note that the original motion capture markers on *stretch* were not located along straight lines.

the simple evaluation of this function. Finally, we also propose a novel model fitting algorithm based on IRLS that computes localized modes of deformation carrying meaningful information to computer vision applications.

NRSFM with RIK is a generic new approach that can make use of customized RIKs to build mappings that even exploit correlations between object appearance and 3D shape. Our approach can potentially combine the functionalities of NRSFM and 3D active appearance models with RIKs [19]: while NRSFM is seen as the training stage, "testing" corresponds to the evaluation of the learned mapping with a previously unseen 2D shape. These new capabilities allow for learning deformable models in a studio, reliably (*e.g.*, with known camera positions in $\mathbf{D}$), to reconstruct the 3D shapes of objects observed elsewhere.

Acknowledgements. This research was supported by the National Institutes of Health, grants R01 EY 020834 and R21 DC 011081.

References

1. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press (2003)
2. Bregler, C., Hertzmann, A., Biermann, H.: Recovering non-rigid 3d shape from image streams. In: Proc. IEEE CVPR. Volume 2. (2000) 690–696
3. Torresani, L., Hertzmann, A., Bregler, C.: Nonrigid structure-from-motion: Estimating shape and motion with hierarchical priors. IEEE Trans. PAMI **30** (2008) 878–892
4. Bartoli, A., Gay-Bellile, V., Castellani, U., Peyras, J., Olsen, S., Sayd, P.: Coarse-to-fine low-rank structure-from-motion. In: IEEE CVPR. Number 1 (2008) 1–8
5. Yan, J., Pollefeys, M.: A factorization-based approach for articulated nonrigid shape, motion and kinematic chain recovery from video. IEEE Trans. PAMI **30** (2008) 865–877
6. Rabaud, V., Belongie, S.: Rethinking nonrigid structure from motion. In: Proc. IEEE CVPR. Volume 1. (2008) 1–8
7. Rabaud, V., Belongie, S.: Linear embeddings in non-rigid structure from motion. In: Proc. IEEE CVPR. (2009)
8. Del Bue, A., Llado, X., Agapito, L.: Non-rigid metric shape and motion recovery from uncalibrated images using priors. In: Proc. IEEE CVPR. Volume 1. (2006) 1191–1198
9. Paladini, M., Del Bue, A., Stošić, M., Dodig, M., Xavier, J., Agapito, L.: Factorization for non-rigid and articulated structure using metric projections. In: Proc. IEEE CVPR. (2009) 2898–2905
10. Fayad, J., Russell, C., Agapito, L.: Automated articulated structure and 3d shape recovery from point correspondences. In: Proc. IEEE ICCV. (2011)
11. Akhter, I., Sheikh, Y., Khan, S.: In defense of orthonormality constraints for nonrigid structure from motion. In: Proc. IEEE CVPR. (2009) 1534–1541
12. Akhter, I., Sheikh, Y.A., Khan, S., Kanade, T.: Trajectory space: A dual representation for nonrigid structure from motion. IEEE Trans. PAMI **33** (2011) 1442–1456
13. Gotardo, P.F.U., Martinez, A.M.: Computing smooth time-trajectories for camera and deformable shape in structure from motion with occlusion. IEEE Trans. PAMI **33** (2011) 2051–2065
14. Gotardo, P.F.U., Martinez, A.M.: Non-rigid structure from motion with complementary rank-3 spaces. In: Proc. IEEE CVPR. (2011) 3065–3072
15. Gotardo, P.F.U., Martinez, A.M.: Kernel non-rigid structure from motion. In: Proc. IEEE ICCV. (2011)
16. Hamsici, O.C., Martinez, A.M.: Rotation invariant kernels and their application to shape analysis. IEEE Trans. PAMI **31** (2009) 1985–1999
17. Scholkopf, B., Smola, A.J.: Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press (2002)
18. Hastie, T., Tibshirani, R., Friedman, J.H.: The elements of statistical learning: data mining, inference, and prediction. 2 edn. Springer (2008)
19. Hamsici, O.C., Martinez, A.M.: Active appearance models with rotation invariant kernels. In: Proc. IEEE ICCV. (2009)