Combining PCA and LFA for Surface Reconstruction from a Sparse Set of Control Points

Reinhard Knothe, Sami Romdhani and Thomas Vetter
University of Basel, Computer Science Department, Basel, Switzerland
reinhard.knothe@unibas.ch, sami.romdhani@unibas.ch, thomas.vetter@unibas.ch

Abstract

This paper presents a novel method for 3D surface reconstruction based on a sparse set of 3D control points. For object classes such as human heads, prior information about the class is used in order to constrain the results. A common strategy to represent object classes for a reconstruction application is to build holistic models, such as PCA models. Using holistic models involves a trade-off between reconstruction of the measured points and plausibility of the result. We introduce a novel object representation that provides local adaptation of the surface, able to fit 3D control points exactly without affecting areas of the surface distant from the control points. The method is based on an interpolation scheme, opposed to approximation schemes generally used for surface reconstruction. Our interpolation method reduces the Euclidean distance between a reconstruction and its ground truth while preserving its smoothness and increasing its perceptual quality.

1. Introduction

The objective of this paper is to present a novel method for reconstructing a densely sampled 3D shape of an object given the 3D position of a sparse set of control points. In order to constrain the problem, the object is assumed to belong to a class of objects, such as human faces, for which prior knowledge is acquired. Hence, the aim is to estimate the 3D surface that matches optimally the given control points while being a plausible instance of the object class given the prior knowledge.

In mathematical terms, this is a problem of function estimation. A function may be estimated using two different mathematical tools: *approximation* or *interpolation*. The control points that are to drive the reconstructed shape may be acquired from different sources: either set manually or provided by some acquisition device. These points may suffer from some deviation from their true values which depends on the acquisition system. This can be considered as noise. When a large number of control points is available (large with respect to the number of degrees of freedom of the model), and the noise probability distribution is known, the noise can be canceled out by using an approximation method. For instance, if the control points are independent and the noise is Gaussian, a least square estimator provides an unbiased approximation of the function which tends to the true function when the number of control points tends to infinity. If the number of control points available is limited, an approximation method is not optimal because the resulting estimate is biased. In this case, an interpolation method that constrains the resulting function to match the control points exactly, is favorable. An example of function approximation using approximation and interpolation is shown in Figure 1.

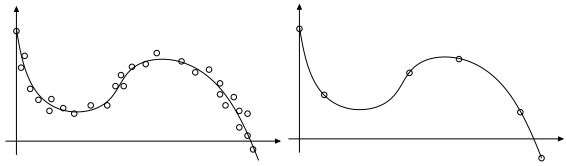


Figure 1: Example of an approximation and an interpolation [1]

When using an approximation method in the case of limited control points, as the noise does not cancels out, the estimated function may overfit the data, as is shown in Figure 2 (b). For instance, when the number of degrees of freedom of the model matches the number of control points, even an approximation method fits the control points are exactly. However, using a holistic model, e.g. a Principal Component Analysis (PCA) based model, which is usually utilized in this setting [2, 3, 4], the model is not adapted to the particular set of control points used. Hence, overfitting is very likely to result. This can be prevented by using a regularization which enforces the resulting function to be plausible according to prior knowledge (Figure 2 (c)). However, the estimate approximates poorly the control points. This is because the regularization does not depend on the noise of the control points and this noise is likely to be overestimated. This method is generally used in computer vision to tackle the problem of 3D surface reconstruction using prior knowledge. For instance, Blanz *et al.* [2] use a least square approximation with a Gaussian regularization to estimate the surface of a head given a small set of control points. In order to avoid overfitting, the Gaussian regularization parameter must be large, resulting in a head shape that approximates inadequately the control points.

Using an interpolation scheme, on the other hand, provides an estimated function that matches the control points perfectly and, with the method introduced in this paper, is also plausible according to the prior knowledge (Figure 2 (d)). An interpolation method requires to use a model that is somehow adapted to the set of control points used: Fitting one control point should not perturb

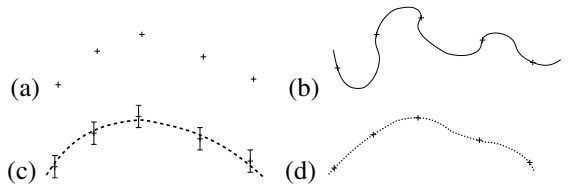


Figure 2: Schematic outline of the problem in 2D: Given a measurement of some vertices (a) our task is to reconstruct a surface. PCA-based reconstruction without regularization suffers from overfitting (b). Introducing a regularization helps to limit the overfitting, but the measured vertices are approximated (c). We present a method to interpolate the measured vertices without overfitting that passes exactly through the input points (d).

the fit of the other ones and should result in a plausible shape. The conventional method to interpolate a linear function is the cubic spline, which estimates a function as a sum of basis spline functions. To avoid the overfitting problem, these basis functions are defined on a local support. In our setting, matching the control points is not the only requirement, we also enforce the estimated function to belong to a class of objects, here, human faces. Therefore we propose a method to adapt the interpolating basis functions such as the estimated 3D surface is a plausible face. Similar to the spline scheme, our basis functions are defined on a local support and are smooth. In order to avoid overfitting, we require the estimated function to have similar spatial derivatives as typical examples of the object class modeled. The Euclidean distance between two surfaces is independent of the spatial derivatives. Therefore we detail in Section 2 a distance measure able to compare spatial derivatives of two surfaces. Using this distance measure, we show that a holistic PCA model is able to model the example class while preserving the spatial derivative characteristics *only at a coarse resolution*, using a few basis functions. In order to model the data at a higher resolution while still maintaining the spatial derivative, we use in Section 3 a Local Feature Analysis (LFA) introduced in Penev *et al.* [5]. Hence, the resulting model is a combination of a holistic PCA model and local LFA model introduced in Section 3.3 and 3.4. We evaluate our novel method on validation 3D heads in Section 4.

2. Derivative based Distance

To evaluate the quality of a reconstruction (with respect to its ground-truth), we need an appropriate distance measure to compare two 3D surfaces. The sum of the Euclidean distances over all vertices of the shapes is the most obvious choice:

$$\sum_{i \in V} \|\mathbf{v}_{r,i} - \mathbf{v}_{t,i}\|, \tag{1}$$

where V is the set of all vertices of the surfaces. $\mathbf{v}_{r,.}$ and $\mathbf{v}_{t,.}$ are corresponding vertices of the two surfaces. Euclidean distance treats all vertices independently and thus is not related to surface smoothness. As one can see in Figure 3, optimizing the Euclidean distance favors noisy, nearby surfaces over smooth and displaced surfaces. To get a good quality of surface reconstruction, a small Euclidean distance is necessary but not sufficient.

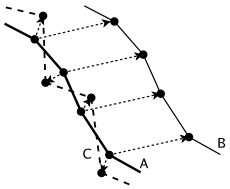
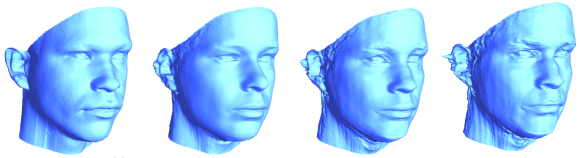


Figure 3: Illustration of the drawbacks of the Euclidean distance. Comparing reconstructed surfaces B and C of the original surface A, the Euclidean distance between B and A is much larger than between C and A, although C appears quite different from A.



N. of coeff.:	65	130	195
Euclid.:	2110	1630	1470
Derivative:	445.6	457.6	490.0

Figure 4: Projecting the leftmost surface to a PCA model that includes more dimensions reduces the Euclidean distance but degrades the perceptual quality. The number of basis vectors used and the projection errors are written below the surfaces.

The visual quality of a reconstruction depends not only on the Euclidean distance of the reconstructed surface from the control points but also on its spatial derivatives. Hence, we introduce a derivative based distance:

$$\frac{1}{|E|} \sum_{(i,j) \in E} \|(\mathbf{v}_{r,i} - \mathbf{v}_{r,j}) - (\mathbf{v}_{t,i} - \mathbf{v}_{t,j})\|_{L_1} \tag{2}$$

where E is the set of all neighboring vertices connected through edges in the surface mesh. Note that, as the mesh is parametrized over a 2D area and that a surface is sampled at equidistant increments of a cylindrical coordinate frame, (r, ϕ), this distance is in fact the norm of the difference of the numerical approximation of the spatial derivatives along r and ϕ. We chose a L_1 distance, to keep commensurate distance units. The derivative based distance is large, if either the reconstructed or the original surface has small wrinkles or spikes at locations where the other surface is flat. So it is a measure of smoothness similarity of two surfaces.

Let us now give an example to demonstrate the advantage of the derivative based distance. We project a novel head into a PCA-model learned from a training set. This model is discussed in details in Section 3.1.

As one can see in Figure 4, the projection gets spikier, noisier and less realistic when increasing the number of PCA-coefficients. However, using more coefficients decreases the Euclidean distance between the original and the reconstruction (see Figure 5, top left plot). This is because there is no smoothness constraint when minimizing the Euclidean distance. In a reconstruction, if the model is able to reduce the Euclidean distance even at the cost of generating a spiky surface,

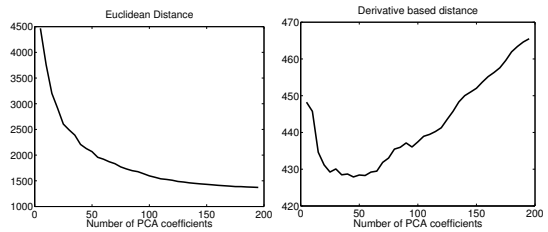


Figure 5: Plots of the average of Euclidean and derivative based reconstruction errors using all vertices of 70 novel heads into a PCA space.

it will do so. In fact, a PCA model does not preserve relationships between neighboring vertices, hence it does not preserve spatial derivatives of the training examples. In order to do so, we should use a model that encodes relationships between neighboring vertices. A method for doing this is presented in Section 3.3.

This experiment may be done for a set of validation shapes (not used for model building). Figure 5 shows a plot of the two distance measures as the number of PCA coefficients used for reconstruction is increased. Note that, in this experiment, all vertices of the mesh are used as control points. The derivative based distance has a minimum at around 60 PCA-coefficients. So, using 60 PCA coefficients gives the best reconstruction with respect to this distance measures. Modeling higher resolution behavior, while still preserving spatial derivatives, requires the use of another type of model as discussed in next section.

3. Class Specific Knowledge

The method we propose is based on a combination of a holistic PCA model, to represent coarse shape variations and a Local Feature Analysis model to account for high frequencies. The two models are first briefly discussed, then the way we used a combination of them to reconstruct a shape is detailed.

3.1. Principal Components Analysis

We use a Morphable model to describe the 3D shape of a linear object class [6]. We assume that we have m examples of human shapes, acquired with a Cyberware scanner. Each example is defined by the 3D position of n vertices. The shape data is arranged in a column vector for each example:

$$\mathbf{x_j} = (x_1, y_1, z_1, x_2, y_2, z_2, ...x_n, y_n, z_n)^T \qquad 1 \leq j \leq m \tag{3}$$

The shape vectors are in correspondence [6]: Each component represents the corresponding vertex of the object within the data set. The arithmetic mean is given by:

$$\bar{\mathbf{x}} = \frac{1}{m}\sum_{j=1}^{m} \mathbf{x}_j \tag{4}$$

The examples are centered around the mean and concatenated into a data matrix:

$$\mathbf{X} = [\mathbf{x}_1 - \bar{\mathbf{x}}, ..., \mathbf{x}_m - \bar{\mathbf{x}}] \in \mathbb{R}^{3n \times m} \tag{5}$$

To compute PCA, a Singular Value Decomposition (SVD) [7] is used. The matrix $\mathbf{X}$ is decomposed into:

$$\mathbf{X} = \mathbf{U}\mathbf{W}\mathbf{V}^{\mathbf{T}} = \sqrt{m}\mathbf{U}\,\mathrm{diag}(\sigma_i)\,\mathbf{V}^T \tag{6}$$

where $\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices, $\mathbf{W}$ is a diagonal matrix and $\sigma_i = \frac{w_{ii}}{\sqrt{m}}$ is the standard deviation of principal component i. The columns of $\mathbf{U} = [\mathbf{u}_1, ...\mathbf{u}_m]$ form an orthogonal set of eigenvectors of the covariance matrix $\Sigma_{\mathbf{X}} = \frac{1}{m}\mathbf{X}\mathbf{X}^T$. The eigenvectors can be used as a basis. Therefore the PCA-coefficients $\mathbf{c}^p$ of a vector $\mathbf{x} \in \mathbb{R}^{3n}$ are calculated by:

$$\mathbf{c}^p = \mathrm{diag}(\frac{1}{\sigma_i})\mathbf{U}^T(\mathbf{x} - \bar{\mathbf{x}}) \in \mathbb{R}^m \tag{7}$$

The vector $\mathbf{x}$ is reconstructed from its coefficients $\mathbf{c}^p$ yielding $\mathbf{x}^{rec} \in \mathbb{R}^{3n}$ as follows:

$$\mathbf{x}^{rec} = \mathbf{U}\cdot\mathrm{diag}(\sigma_i)\cdot\mathbf{c}^p + \bar{\mathbf{x}} = \mathbf{U}\mathbf{U}^{\mathbf{T}}\cdot(\mathbf{x} - \bar{\mathbf{x}}) + \bar{\mathbf{x}} \tag{8}$$

The coefficients $\mathbf{c}^p$ are decorrelated in the sense that the correlation matrix of the training example coefficients is the identity matrix.

3.2. Local Feature Analysis

The PCA representation is typically not topographic, meaning that nearby components of the PCA-coefficient vector $\mathbf{c}^p = (c_1, ..., c_i, c_{i+1}, ...)^T$ have no spatial relationship. Topography in this context means, that nearby components should have the same relationship as the corresponding components in the input vector $\mathbf{x} = (x_1, ..., x_i, x_{i+1}, ...)^T$. Thus, the coefficients should be labeled in the same way as the input vector $\mathbf{x}$. Penev and Atick [5] have introduced a model that preserves topography, Local Feature Analysis (LFA), which we now overview. The most general topographic kernel that projects signals to the subspace spanned by the eigenvectors of $\Sigma_{\mathbf{X}}$ is [5]:

$$\mathbf{c}^l = \frac{1}{m}\mathbf{U}\cdot\mathbf{Q}\cdot\mathbf{U}^T\cdot(\mathbf{x} - \bar{\mathbf{x}}) \tag{9}$$

where $\mathbf{U}$ is the matrix of the eigenvectors of $\Sigma_{\mathbf{X}}$ (see Equation 6), $\mathbf{Q}$ is an arbitrary $m \times m$ matrix, $\mathbf{x}$ is an input vector and $\mathbf{c}^l$ is the LFA-coefficients vector.

As $\mathbf{c}^l \in \mathbb{R}^{3n}$, we can no longer satisfy the condition of coefficients decorrelation. It can be shown [5] that the coefficients are as decorrelated as possible if

$$\mathbf{Q} = \mathrm{diag}(\frac{1}{\sigma_i}). \tag{10}$$

LFA provides a component for each dimension of the space modeled. This component is the mode of variation of one vertex. Its support is local and includes the vertices which are correlated with the corresponding vertex. Moreover, LFA spans the same subspace as PCA. Given the LFA-coefficients, the input vector is

reconstructed by:

$$\begin{aligned}\mathbf{x}^{rec} &= m\cdot\mathbf{U}\cdot\mathbf{Q}^{-1}\cdot\mathbf{U}^T\mathbf{c}^l+\bar{\mathbf{x}}\\ &= m\cdot\mathbf{U}\cdot\mathrm{diag}(\sigma_i)\cdot\mathbf{U}^T\mathbf{c}^l+\bar{\mathbf{x}}\\ &= \mathbf{U}\mathrm{diag}(\sigma_i)\mathbf{U}^T\mathbf{U}\mathrm{diag}(\tfrac{1}{\sigma_i})\mathbf{U}^T(\mathbf{x}-\bar{\mathbf{x}})+\bar{\mathbf{x}}\\ &= \mathbf{U}\mathbf{U}^\mathbf{T}(\mathbf{x}-\bar{\mathbf{x}})+\bar{\mathbf{x}}\end{aligned} \tag{11}$$

3.3. Adjusting a shape given one vertex

We now show how to use LFA for shape reconstruction given a sparse set of control points. For explanation purposes we first show how to adjust a shape to match a single vertex. In the next section, we will use this method to match a shape to a set of control points using a combination of PCA and LFA models. Starting from the shape of a head (e.g. the mean head or a coarse reconstruction or even any head from the database as is shown in Figure 6), we want to move one of its vertices to an arbitrary position so that the resulting head is a plausible head in terms of position of its vertices and their spatial derivatives. To meet these requirements, the modification of the head should have a local support. Changing the tip of the nose should have no effect on the ears (see Figure 6). This is achieved by moving in the direction of the LFA component associated, in this example, with the tip of the nose.

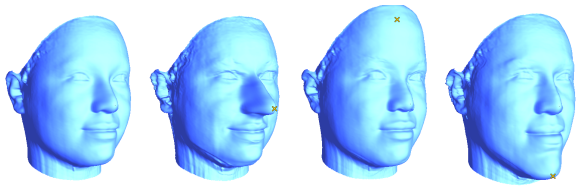


Figure 6: A face (left) and three deformations of it are shown, obtained by displacement of different control points marked by a cross (Equations (14) and (15)).

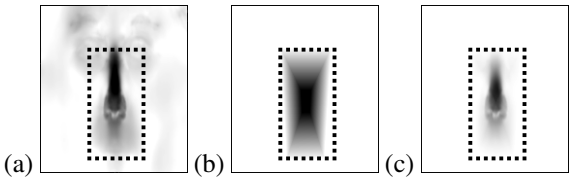


Figure 7: (a) z-component of the LFA vector for the tip of the nose, (b) its mask given by the Voronoi Tessellation and (c) their pointwise multiplication.

Let us denote by $\mathbf{r}$ the displacement between a measurement $\mathbf{v}$ of a vertex at position i and the corresponding vertex of an input shape that is to be modified:

$$\mathbf{r}=\mathbf{v}-\mathbf{x}|_P \qquad \mathbf{r},\mathbf{v}\in\mathbb{R}^3 \tag{12}$$

where $P=\{3i,3i+1,3i+2\}$ is the set of components in the shape vector $\mathbf{x}$ that represents the (x,y,z) coordinates of vertex i. We now project this displacement onto the LFA basis corresponding to the vertex i. Note that although the LFA is learnt on mean centered data, we can also legitimately project to it the difference between an input shape and not only the average shape but also any shape in the span of the PCA model. The corresponding rows of $\mathbf{U}$ are denoted by $\mathbf{U}|_P$. Equation (11) is now restricted to the components P. Since it is not reasonable to compute more coefficients than input dimensions, we also restrict the LFA-coefficients $\mathbf{c}^l$ to P:

$$\mathbf{r}=m\mathbf{U}|_P\cdot\mathrm{diag}(\sigma)\cdot(\mathbf{U}|_P)^T\cdot\mathbf{c}^l_P \tag{13}$$

This Equation is solved for the LFA-coefficients $\mathbf{c}^l_P$. Note that in this case, we can use Equation (11) directly (instead of Equation (9)) since the matrix $(\mathbf{U}|_P\cdot\mathrm{diag}(\sigma)\cdot(\mathbf{U}|_P)^T)\in\mathbb{R}^{3\times3}$ is non-singular:

$$\mathbf{c}^l_P=\frac{1}{m}\cdot(\mathbf{U}|_P\cdot\mathrm{diag}(\sigma)\cdot(\mathbf{U}|_P)^T)^{-1}\cdot\mathbf{r} \tag{14}$$

The reconstruction is obtained from the coefficients $\mathbf{c}^l_P$ only, as the other coefficients are zero.

$$\begin{aligned}\mathbf{x}^{rec}_i &= m\cdot\mathbf{U}\cdot\mathrm{diag}(\sigma_j)\cdot(\mathbf{U}|_P)^T\cdot\mathbf{c}^l_P+\mathbf{x}\\ &= m\cdot\mathbf{L}_i\cdot\mathbf{c}^l_P+\mathbf{x}\end{aligned} \tag{15}$$

where $\mathbf{L}_i=\mathbf{U}\cdot\mathrm{diag}(\sigma)\cdot(\mathbf{U}|_P)^T\in\mathbb{R}^{3n\times3}$.

This method is able to deform a shape so that the resulting shape passes through an input control point and is maximally correlated with the set of prototypical shapes used. We denote this by *Maximal Correlation Interpolation*. If several control points are to be interpolated, their influence on one another should be null. Therefore, the value of $\mathbf{L}_i$ for one control point should be $(0,0,0)$ for all the other ones. This is achieved by multiplying $\mathbf{L}_i$ by a factor that is 1 at the control point i and 0 at all other control points. This factor decreases linearly between the control point i and the neighboring ones, see figure 7. This is implemented easily using a Voronoi Tessellation.

3.4. Reconstruction from a few control points

In this section, we detail the usage of a combined PCA and LFA model to reconstruct the shape from a small set of control points. Because no unbiased noise estimate can be made from a small set of control points, an interpolation scheme is favored over an approximation method. Hence, the reconstructed surface is enforced to match the points exactly.

First, the shape is reconstructed using the coarse holistic PCA model with 60 coefficients (see Section 2). This is achieved using the approximation with regularization method of Blanz *et al.* [2], by minimizing the following cost function:

$$E=\|\mathrm{diag}(\sigma)\mathbf{U}|_F\mathbf{c}^p-\mathbf{x}|_F\|+\eta\|\mathbf{c}^p\|^2 \tag{16}$$

where F includes the indices of the (x,y,z) components of the set of control points and $\mathbf{x}|_F$ is their 3D position. The first part of the error term is the Euclidean reconstruction error of the control points and the second term is a regularization term to control the plausibility of the reconstruction that assumes that the control points are independent, identically and normally distributed. It can be shown that this cost function is minimized by [2]:

$$\mathbf{c}^p=\mathbf{V}'\cdot\mathrm{diag}(\frac{w_i}{w_i^2+\eta})\cdot\mathbf{U}'^T\cdot\mathbf{x}|_F \tag{17}$$

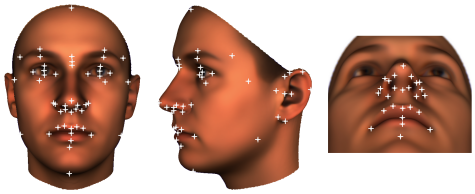


Figure 8: Positions of the 73 Farkas points.

where $\mathbf{U}'\mathbf{W}\mathbf{V}'^T$ is the singular value decomposition of the matrix $\mathbf{U}|_F \mathrm{diag}(\sigma)$.

We have already mentioned that overfitting is the main problem of PCA-based methods. The overfitting can be reduced by restricting the number of PCA eigenvectors in Equation (8) or by choosing a larger regularization term (see Equation (16)). However, this involves a larger reconstruction error. Here, we use PCA to obtain a coarse estimation of the shape with a relatively large error at the measured vertices (using a large regularization ensuring no overfitting). Then local models (one for each measured vertex $\mathbf{v}$) are utilized to compensate the approximation error. A shape is then reconstructed with Equation (8):

$$\sum_{i=1}^{m'} c_i^p \sigma_i \cdot \mathbf{u}_i + \bar{\mathbf{x}} \tag{18}$$

where the coefficients c_i^p are obtained using Equation (14), in which $\mathbf{r}$ is the residual between the coarse PCA approximation and each input control point. As a result, in the proposed combined method, the shape is represented as follows:

$$\sum_{i=1}^{m'} c_i^p \sigma_i \cdot \mathbf{u}_i + \sum_{i=1}^{|F|/3} \mathbf{L}_i \cdot \mathbf{c}_i^l + \bar{\mathbf{x}} \tag{19}$$

where the PCA-coefficients c_i^p are computed using Equation (17) and the LFA-coefficients $\mathbf{c}_i^l$ are computed using Equation (14).

4. Experiments on Face Data

In this experiment we choose the set of human faces as object class. To obtain a set of prototypical examples, a Cyberware laser scanner with a spatial resolution of less than $1mm$ was used. The surface is represented by $n = 75972$ vertices. The vertices are combined in a shape vector, as in Equation (3). Corresponding points, such as the tip of the nose, the corner of the eye, etc. are represented by the same vector component across all shape vectors. Hence, a linear combination of shape vectors produces realistic novel faces [6, 2]. A database of 270 individuals is used. It is split in a training set of $m = 200$ faces and a test set of 70 faces.

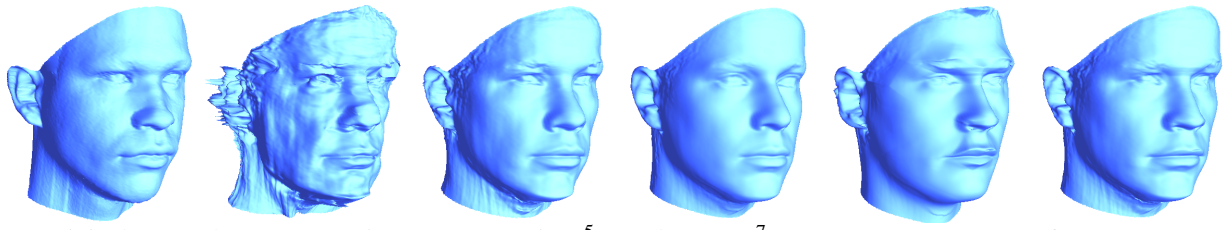
In the experiments detailed in this section, we use the *Farkas* points as input control points. Farkas [8] describes a set of 78 reference points for measurements of the human head. We use 73 of them as our face model covers only a portion of the entire head: from one ear to the other, see Figure 8. We have chosen the Farkas control points for two reasons: They are widely used in anthropology and medicine to describe a human head and they are relatively easy to locate. These control points are used as a fixed set of input points and we trained the LFA model, thereby producing $\mathbf{L}_i$ (Equation 19), on this particular set of points. This is not a limitation, since we could use any other set of points and automatically train our model accordingly. Since the heads are in correspondence, it suffices to specify the Farkas points on a single face, in order to get their positions on each face of the training and testing set.

Examples of reconstructions of a head of the test set from the 73 Farkas points using several methods are shown in Figure 9. It can be seen that using PCA, the regularization parameter η must be chosen by the user. A small value improves the Euclidean distance but degrades the derivative based distance and a large value does the opposite. Hence, there is a trade-off between position accuracy and smoothness accuracy. Using LFA only provides a head with small derivative based error but with large Euclidean error. The last rendering of this figure shows a reconstruction obtained by the proposed combined PCA/LFA model. This reconstruction minimizes both, *simultaneously* the Euclidean error (related to point position) as well as the derivative based error (related to spatial derivatives). Perceptually as well as numerically, this reconstruction is the closest to the original head. It should be noted that the combined model is more flexible than the PCA model: As advocated in Section 2, 60 PCA components were used. Using more components would favor overfitting and large derivative based error. The combined model also includes 60 PCA components as well as $3 \cdot 73$ LFA components. One of the advantage of the LFA model is that much more components can be used than with a PCA model (the upper limit of m components of PCA is lifted for LFA), while still preserving the smoothness of the training data.

Figure 10 provides quantitative evaluation of the method performed over the whole test set. Two sets of experiments are presented. The first one (top row) shows results obtained using the 73 Farkas points. On this experiment, more coefficients were used for the combined PCA/LFA model than for the PCA model. This is because the combined model permits the usage of more dimensions than the PCA model. It can be seen that using a large value of the regularization parameter η, the PCA model can provide a reconstruction similar to the combined model in terms of smoothness, but then the discrepancy between the original surface and the reconstructed surface provided by the PCA model is much larger than with the combined PCA/LFA model. The same conclusion can be drawn when the same number of coefficients for both models is used (second row), however to a lesser extent. In this experiment only 33 control points were used.

5. Conclusion

We have presented a novel method for 3D surface reconstruction from sparse data. It is based on an interpolation scheme enabled by a novel combined PCA/LFA model. It has been applied to estimate the 3D shape of human faces from a small set of control points. In contrast to the previous PCA-only-based approach, our method improves the results both in terms



(a) Original	(b) PCA: $\eta = 5000$	(c) $\eta = 5 \cdot 10^5$	(d) $\eta = 10^7$	(e) LFA	(f) PCA/LFA
N. of coeff:	159	159	159	$3 \cdot 73 = 219$	$60 + 3 \cdot 73 = 279$
Euclid. dist.:	4430	2840	3240	6620	2570
Derivative dist.:	941.8	507.5	462.0	468.5	462.1

Figure 9: Shapes (b) to (f) are reconstructions from 73 Farkas points of the ground truth shape (a). The first three reconstructions were obtained by PCA approximation with different values of the regularization parameter. The shape (e) uses LFA, and the shape (f) uses the novel combined PCA/LFA model.

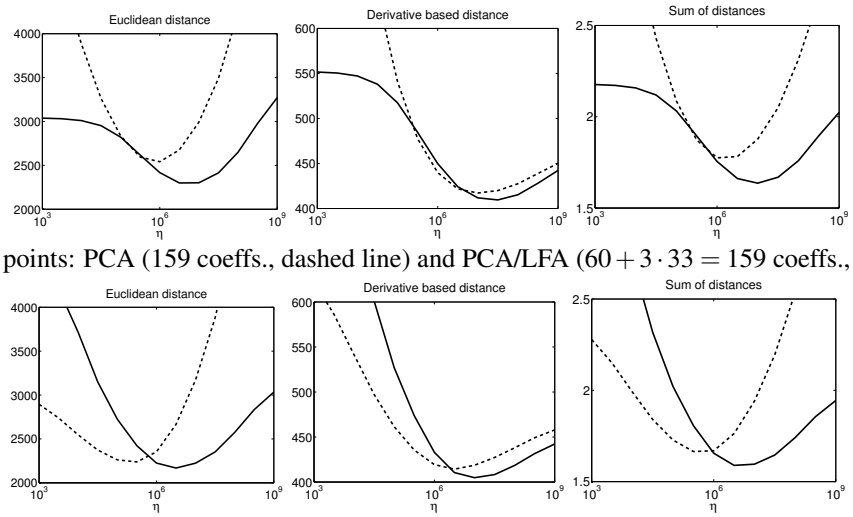


Figure 10: Comparison of the reconstruction errors (Euclidean, derivative based and the sum of both) using PCA (dashed line) and the combined PCA/LFA method (solid line) for different values of the regularization parameter (x-axis). On the top row, the reconstructions are computed using the 73 Farkas points, and on the bottom row, using 33 control points. The reconstruction error is computed over a complete shape and averaged over the 70 examples of the test set.

of point-wise accuracy and also in terms of smoothness accuracy: Not only the 3D position of the vertices but also the spatial derivatives of the reconstructed surface are closer to the original surface using the reconstruction of the combined PCA/LFA model than the one obtained by PCA. Smoothness accuracy is important as it is strongly related to how humans perceive shapes. In this paper we used Farkas points to control a shape reconstruction. However, it may be possible that a selection of other control points could decrease both Euclidean and derivative based errors. It is left for further research to investigate what this set of points could be and how to use this knowledge in an analysis of variations of human heads.

Acknowledgements R.K. would like to thank Beat Röthlisberger and Pascal Paysan for carefully reading the manuscript. This work was supported by grants from the Swiss National Science Foundation 200021-103814 and the NCCR CO-ME.

References

[1] H. R. Schwarz and N. Köckler. *Numerische Mathematik*. Teubner, 5th edition, 2004.

[2] Volker Blanz and Thomas Vetter. Reconstructing the complete 3D shape of faces from partial information. *it+ti Oldenburg Verlag*, 44(6):295–302, 2002.

[3] V. Blanz, A. Mehl, T. Vetter, and H.-P. Seidel. A statistical method for robust 3D surface reconstruction from sparse data. In *2nd Int. Symposium on 3D Data Processing, Visualization and Transmission*, pages 293–300, 2004.

[4] B. Allen, B. Curless, and Z. Popović. The space of human body shapes: Reconstruction and parameterization from range scans. *ACM Trans. Graph.*, 22(3):587–594, 2003.

[5] P. Penev and J. Atick. Local feature analysis: A general statistical theory for object representation. *Neural Systems*, 7:477–500, 1996.

[6] V. Blanz and T. Vetter. A morphable model for the synthesis of 3D faces. In *Siggraph 1999*, pages 187–194. Addison Wesley Longman, Reading, Massachusetts, 1999.

[7] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C / the Art of Scientific Computing*. Cambridge University Press, 1999.

[8] Leslie G. Farkas. *Anthropometry of the Head and Face*. Raven Press, 2nd edition, 1994.