

3D Face Alignment Without Correspondences

Zsolt Sánta and Zoltan Kato^(✉)

Institute of Informatics, University of Szeged, PO. Box 652, Szeged 6701, Hungary
{santazs,kato}@inf.u-szeged.hu

Abstract. A novel correspondence-less approach is proposed to find a thin plate spline map between a pair of 3D human faces represented by triangular surface meshes. The proposed method works without landmark extraction and feature correspondences. The aligning transformation is simply found by solving a system of nonlinear equations. Each equation is generated by integrating a non-linear function over the surfaces represented as fuzzy sets of triangles. We derive approximating formulas for the efficient computation of these integrals. Based on a series of comparative tests on a standard 3D face dataset, our triangular mesh-based algorithm outperforms state of the art methods in terms of computing time while maintaining accuracy.

Keywords: 3D face · Alignment · Thin plate spline · Correspondence-less

1 Introduction

The analysis of 3D faces is an important task in many applications, like face comparison, face motion capture, facial expression recognition [5], biometric identification [16] as well as several medical related problems, *e.g.* surgical planning [8, 26] and craniofacial dysmorphology [29]. A core component of many 3D face analysis tasks is the geometric alignment of faces. Most of the alignment algorithms are relying on the extraction of well-defined feature points or landmarks. Detection of face landmarks in 2D images has a long research track in the literature [4, 10]. With new 3D image acquisition devices and technologies, 3D face scans are becoming widespread [6, 14, 16, 20, 29]. One of the main advantage of the 3D scans over 2D images is that they are not affected by viewpoints and lighting conditions. Unfortunately, landmark localization in 3D data is a hard task even by user interaction. Current methods are relying on either geometric (*e.g.* curvature of the face surface) or color information only. For example, nose tip is usually detected as peak in the curve of the face [14, 19, 29], while color-based features are typically extracted by adopting 2D feature point detector and descriptor techniques such as SIFT [16]. Training-free landmark localization methods are using carefully designed rules [29]. However, these rules are usually not independent from each other, where estimation errors are propagated to subsequent stages [29]. On the contrary, training-based methods are more flexible but require a sufficiently large training dataset [29]. These type of methods

are built upon *e.g.* statistical models [19] or techniques originated from machine learning [6, 29].

In this paper, we propose a different strategy for 3D face alignment, which works without point correspondences. Given a pair of *template* and *observation* 3D face scans, we trace back the alignment to the solution of a system of non-linear equations, which directly provides the parameters of the aligning transformation. What kind of transformation should we consider? Obviously, face alignment requires a generic elastic transformation. The most generic mapping between the faces could be given as a vector field $\phi : \mathbb{R}^3 \rightarrow \mathbb{R}^3$, but such a transformation has too many degree of freedom, which makes its estimation difficult. Therefore, to reduce the complexity of the problem, the general mapping is often replaced by a parametric deformation model. Following [9] and [28], the deformation models could be organized into three main groups: physical model based, models derived from interpolation and approximation theory, and knowledge-based geometric transformations. In our approach, we propose to model the deformation by thin plate splines (TPS) [3, 31, 32]. TPS models are often used whenever a parametric free-form registration is required but the underlying physical model of the object deformation is unknown or too complex.

Observing the possible deformations in face alignment task, we can either consider intra- or inter-person registration [30]. In the former case we register different scans from the same person, therefore the deformation usually caused by various facial expressions. In the latter case scans from different people are registered, thus we have to deal with large non-rigid deformations caused by the variation of faces in size and shape from one person to another. Moreover, in intra-person registration the deformation is acting more locally with respect to the inter-person case, hence it is easier to find similar areas to restrict the space of possible deformations. In the current proposal, our aim is to solve the inter-person registration problem.

From a methodological point of view, registration methods can be classified into two main groups: geometric (or landmark-based) methods and iconic (or area-based) approaches [28]. The fundamental difference is that geometric methods rely on extracted landmarks placed in salient image locations while iconic methods are using the whole image domain to determine the transformation. Geometric methods are challenged by the correspondence problem, which is particularly difficult to solve in the case of non-linear deformations. Iconic approaches are typically relying on the availability of rich radiometric information which is used to construct a similarity measure based on some kind of intensity correlations. The aligning transformation is then found by maximizing similarity between the objects, which usually yields a complex non-linear optimization procedure. Hybrid methods are also available, combining the bests properties of both worlds [28]. Unfortunately, none of these methods are well adapted to 3D face registration problems, as radiometric information is either missing or low quality.

In practice, 3D surfaces are given as triangular surface meshes. Current methods are either using the whole triangular surface [24] or focusing on registering

the vertex set only [11, 18]. In the latter case, a popular way to deal with unstructured point sets is to represent each set by a suitable, problem specific model (*e.g.* Gaussian mixtures): In [18], a probabilistic model is proposed where a Gaussian mixture with centroids corresponding to the *template* set is fit to the *observation* set by maximizing the likelihood. Thus, an energy function composed of the negative log-likelihood and an additional regularization term is minimized using the Expectation Maximization algorithm. The transformation is represented by parametric radial basis functions using a Gaussian kernel. In [11], both point sets are represented as a Gaussian Mixture Model and then the L2 distance of the two mixtures is minimized. The authors use a closed-form expression to calculate the distance between the Gaussian mixtures efficiently. The underlying deformation is modeled using thin plate splines. Both approaches reported to be robust against occlusions, however they are inefficient for large point sets, due to the computational cost.

In this paper, a TPS-based method is proposed for aligning triangulated facial 3D scans based on a recent registration framework [7, 24]. The method relies on geometric information only and the parameters of the TPS is obtained directly as a solution of a properly generated system of equations. The main idea behind the generation is to integrate a set of non-linear functions over the domains of the input data in order to eliminate the need for individual point correspondences. Unlike [24], which works only for closed surfaces (*i.e.* volumetric objects) and [7], which works only for 2D planar regions, the proposed approach is specifically developed for open 3D surfaces. The performance of the algorithm is evaluated on a subset of the Bosphorus Dataset [25]. We remark, that by aligning 3D face scans, one can transfer landmarks from one scan to the other. Although this is not a usual way to solve landmark extraction, similar approaches could be found in the literature. For example, a training-free method is proposed in [14], where the initial landmark estimation is refined by a deformable registration approach [1], using a few extracted landmark location as constraints on the final result.

2 3D Face Alignment

The proposed method has a very limited assumption about the input: faces are represented as triangulated surface meshes without any radiometric data. Obviously, the spatial resolution of the input mesh determines the possible alignment precision, hence good quality input is critical for precise alignment. Let us now consider two faces represented as 3D surfaces: one is called the *template* and the other one is called the *observation*, denoted by $\mathcal{F}_t \subset \mathbb{R}^3$ and $\mathcal{F}_o \subset \mathbb{R}^3$, respectively. We are looking for the aligning transformation $\varphi : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ such that for all $\mathbf{x} \in \mathcal{F}_t$ there exists a $\mathbf{y} \in \mathcal{F}_o$ satisfying the so called *identity relation*

$$\varphi(\mathbf{x}) = \mathbf{y} \quad (1)$$

In classical landmark based approaches, a large number of corresponding landmarks are extracted from \mathcal{F}_t and \mathcal{F}_o , giving sufficiently many constraints through

Eq. (1) to find the parameters of the transformation. An alternative solution has been proposed in [7, 24], which relies on segmented regions instead of point correspondences: The idea is to integrate out individual point pairs in Eq. (1) over the foreground domains of the objects yielding the following equation:

$$\int_{\mathcal{F}_o} \mathbf{y} d\mathbf{y} = \int_{\varphi(\mathcal{F}_t)} \mathbf{z} d\mathbf{z}. \quad (2)$$

While in landmark-based approaches each point correspondence will generate a new equation of the form Eq. (1); Eq. (2) provides only three equations ($\mathbf{y}, \mathbf{z} \in \mathbb{R}^3$)! As a consequence, Eq. (2) alone is not enough to solve for the transformation parameters. In order to generate more equations, observe that Eq. (1) (hence Eq. (2)) remains valid when a non-linear $\omega : \mathbb{R}^3 \rightarrow \mathbb{R}$ function is acting on both sides [7, 24]. Thus applying a set of independent non-linear functions $\{\omega_i\}_{i=1}^\ell$ yields a system of ℓ equations [24]:

$$\int_{\mathcal{F}_o} \omega_i(\mathbf{y}) d\mathbf{y} = \int_{\varphi(\mathcal{F}_t)} \omega_i(\mathbf{z}) d\mathbf{z} \quad i = 1, \dots, \ell. \quad (3)$$

Since the *template* and *observation* surfaces are represented by their triangular surface meshes, let us denote them by T_Δ and $O_\Delta \subset \mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{R}^3$, respectively. They are a piecewise linear approximation of the true surfaces \mathcal{F}_t and \mathcal{F}_o , hence

$$\begin{aligned} T_\Delta &\approx \mathcal{F}_t \\ O_\Delta &\approx \mathcal{F}_o \\ \varphi(T_\Delta) &\approx \varphi(\mathcal{F}_t) \end{aligned} \quad (4)$$

and thus the integrals over the triangular surfaces will be approximations of the integrals over the true surfaces as well. The integrals over the triangular surfaces can be expressed as sums of integrals over the triangles of each mesh:

$$\sum_{o \in O_\Delta} \int_o \omega_i(\mathbf{y}) d\mathbf{y} \approx \sum_{\pi \in \varphi(T_\Delta)} \int_\pi \omega_i(\mathbf{z}) d\mathbf{z}. \quad (5)$$

Theoretically, any integrable $\{\omega_i\}_{i=1}^\ell$ function set could be used, but for computational efficiency we will use low-order power functions

$$\omega_i(\mathbf{x}) = x_1^{n_i} x_2^{m_i} x_3^{o_i}, \quad (6)$$

where $\{(n_i, m_i, o_i)\}_{i=1}^\ell = \{(a, b, c) \mid a + b + c = O\}$ and $O \in \{0, \dots, M\}$. The M value corresponds to the maximal degree of the polynomial set and it is chosen to provide the necessary amount of functions to determine the parameters of the transformation. Note that, using these ω_i functions the integrands will be various geometric moments of each triangle, which can be efficiently computed by making use of the methods proposed in [13, 22].

2.1 Determining the Integration Domains

One can consider Eq. (5) as an object level *identity relation*, because here we only require, that the object domains \mathcal{F}_t and \mathcal{F}_o be in correspondence as a whole. How to ensure this region-level correspondence for 3D faces? This is by far not a trivial question, because facial scans typically focus on the frontal face, but depending on the actual setting, other parts of the head are also visible. Therefore the scanned surfaces will not match as a whole! Moreover, the exact segmentation of corresponding parts is a rather complex problem as there are no clearly defined borders of a face in a 3D scan. Instead of solving a hard 3D face segmentation problem, let us define the integration domains in Eq. (5) as fuzzy sets [33] with a $W_{\lambda_1 \lambda_2} : (\mathbf{A}, \mathbf{B}, \mathbf{C}) \rightarrow [0, 1]$ membership function giving the weight of each triangle in the integrals of Eq. (5):

$$\sum_{o \in N_o(O_\Delta)} W_{\lambda_1 \lambda_2}(o) \int_o \omega_i(\mathbf{y}) d\mathbf{y} \approx \sum_{\pi \in \varphi(N_t(T_\Delta))} W_{\lambda_1 \lambda_2}(\pi) \int_\pi \omega_i(\mathbf{z}) d\mathbf{z}, \quad (7)$$

The membership function $W_{\lambda_1 \lambda_2}$ is governed by three parameters:

1. λ_1 is the upper threshold of the inner parts (where $W_{\lambda_1 \lambda_2} = 1$)
2. λ_2 is the lower threshold of the outer parts (where $W_{\lambda_1 \lambda_2} = 0$)
3. the interpolation method for the area between λ_1 and λ_2 , which is either a linear or a step function. For the *step* method, $W_{\lambda_1 \lambda_2} = 0.5$ between λ_1 and λ_2 , while the *linear* method sets $W_{\lambda_1 \lambda_2}$ between 0 and 1 proportionally to the distance.

The thresholds λ_1 and λ_2 are determined based on geodesic distances and by making use of a training set with ground truth landmark locations. First, we choose an origin point on the surface for the geodesic distance calculation. This point should be easy and stable to detect, therefore we recommend to use the nose tip. Moreover, this point is approximately at the middle of all important facial landmarks. Next, we determine the maximal geodesic distance with respect to the nose tip. We tried two different maximal values: (1) the true maximal geodesic distance as well as (2) the mean of the top 5% geodesic distances from the nose tip. In the next step, we calculate the geodesic distance to each ground truth landmark location and normalize the calculated distances with the previously determined maximal value in order to achieve head size invariance. In our experiments, we observed that taking the mean of the normalized distances on all training data will give an area containing most of the landmarks. Therefore, we define λ_1 as this mean value. Similarly, for λ_2 we simply add the standard deviation of the normalized distances to λ_1 . Thus the thresholds are expressed as the percentage of the maximal geodesic distance within the face. An example is shown in Fig. 1.

The main advantage of this fuzzy representation is that we can focus the equations on the more stable, inner parts of faces without the need of a high precision segmentation of the 3D meshes.

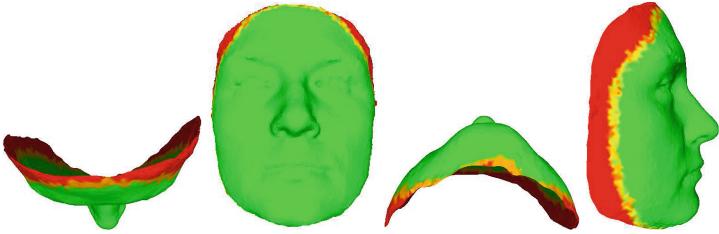


Fig. 1. An example for the membership functions. The green area denotes the inner parts having a membership value of 1, while the yellow and red regions denote the areas between λ_1 and λ_2 , and the areas above λ_2 , respectively. (Color figure online)

2.2 Computing the Integrals over Triangular Meshes

Now, we will derive an efficient numerical schemes for the calculation of the integrals, which uses a linear approximation of the ω_i functions yielding a fast algorithm. Following [22], we will use barycentric parametrization of the triangles. Considering an arbitrary triangle $o = (\mathbf{A}, \mathbf{B}, \mathbf{C})$, every \mathbf{p} point of the triangle can be expressed as a weighted sum of the vertices:

$$\mathbf{p} = u\mathbf{A} + v\mathbf{B} + w\mathbf{C}, \quad (8)$$

with $u, v, w \geq 0$ and $u + v + w = 1$. Thus the above formula has two free parameters as one weight can be expressed by the other two, e.g. $w = 1 - u - v$. Since we apply the above transformation, the area changes $\|(\mathbf{A} - \mathbf{C}) \times (\mathbf{B} - \mathbf{C})\| = 2 \text{area}(o)$ induced by this reparametrization has to be taken into account in the integrals over a triangle o . Thus the first integral in Eq. (7) becomes

$$\begin{aligned} \sum_{o \in O_\Delta} W_{\lambda_1 \lambda_2}(o) \int_o \omega_i(\mathbf{y}) d\mathbf{y} &= \\ \sum_{o \in O_\Delta} 2 \text{area}(o) W_{\lambda_1 \lambda_2}(o) \int_0^1 \int_0^{1-u} \omega_i(u\mathbf{A} + v\mathbf{B} + w\mathbf{C}) dv du, \end{aligned} \quad (9)$$

where $i = 1, \dots, \ell$ and $w = 1 - u - v$. A similar formula can be derived for the other integral in Eq. (7). Considering the $\{\omega_i\}$ set from Eq. (6) and the barycentric parameterization from Eq. (8), we can linearly interpolate the $\{\omega_i\}$ functions using the vertices of each triangle $o = (\mathbf{A}, \mathbf{B}, \mathbf{C})$ as

$$\omega_i(\mathbf{p}) \approx u\omega_i(\mathbf{A}) + v\omega_i(\mathbf{B}) + w\omega_i(\mathbf{C}) \quad i = 1, \dots, \ell. \quad (10)$$

Substituting the above approximation into Eq. (9), we get the following approximation for the integral:

$$\begin{aligned} \sum_{o \in O_\Delta} W_{\lambda_1 \lambda_2}(o) \int_o \omega_i(\hat{\mathbf{y}}) d\hat{\mathbf{y}} &\approx \\ \sum_{o \in O_\Delta} 2 \text{area}(o) W_{\lambda_1 \lambda_2}(o) \int_0^1 \int_0^{1-u} u\omega_i(\mathbf{A}) + v\omega_i(\mathbf{B}) + w\omega_i(\mathbf{C}) dv du, \end{aligned} \quad (11)$$

where $w = 1 - u - v$ and $i = 1, \dots, \ell$. Since ω_i is independent from the integration variables, we get the closed form expression

$$\int_o \omega_i(\mathbf{y}) d\mathbf{y} \approx \text{area}(o) \frac{\omega_i(\mathbf{A}) + \omega_i(\mathbf{B}) + \omega_i(\mathbf{C})}{3}, \quad (12)$$

which is simply the mean of the function values in the vertices weighted by the area of the triangle. Note that this formula will be exact for every linear functions, thus the precision of the approximation depends on the local linearity of the function and the size of the triangle. The complexity of this algorithm is $\mathcal{O}(M^3)$, where M is the maximal order of the ω_i functions.

2.3 Transformation Model

While the proposed method works with almost any parametric transformation model ϕ , what is the optimal choice for 3D face alignment? Obviously, we need an elastic transformation to warp faces between different persons. From a computation point of view, the transformation should have a minimal number of parameters, because the number of equations in Eq. (12) directly depends on this number. Thin plate splines (TPS) [3, 31, 32] are broadly used for such alignment problems.

In 3D, a TPS transformation $\varphi : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ can be decomposed as three coordinate functions $\varphi(\mathbf{x}) = [\varphi_1(\mathbf{x}), \varphi_2(\mathbf{x}), \varphi_3(\mathbf{x})]^T$, $\forall \varphi_i(\mathbf{x}) : \mathbb{R}^3 \rightarrow \mathbb{R}$. Given a set of control points $c_k \in \mathbb{R}^3$ and associated mapping coefficients $a_{ij}, w_{ki} \in \mathbb{R}$ with $i = 1, \dots, 3$, $j = 1, \dots, 4$ and $k = 1, \dots, K$, the TPS functions are

$$\varphi_i(\mathbf{x}) = a_{i1}x_1 + a_{i2}x_2 + a_{i3}x_3 + a_{i4} + \sum_{k=1}^K w_{ki}U(\|c_k - \mathbf{x}\|), \quad (13)$$

where $U(r) = -r$ [31] is called radial basis function. The transformation has $N = 3(K + 4)$ parameters: 12 affine parameters a_{ij} and 3 local coefficients w_{ki} for each control point c_k satisfying the following additional constraints [31, 32], which ensure that the TPS at infinity behaves according to its affine term:

$$\sum_{k=1}^K w_{ki} = 0 \quad \text{and} \quad \sum_{k=1}^K c_{kj} w_{ki} = 0 \quad i, j = 1, 2, 3. \quad (14)$$

When correspondences are available, the exact mapping of the control points are also known which, using Eq. (13), provides constraints on the unknown parameters. Thus in classical correspondence based approaches, control points are placed at extracted point matches, and the deformation at other positions is interpolated by the TPS. Therefore in such cases, a TPS can be regarded as an optimal *interpolating* function whose parameters are usually recovered via a complex optimization procedure [3, 32]. However, in the current approach, TPS is used as a parametric model to *approximate* the true deformation [7]. How to place the control points in such cases? A trivial way, also explored in [7], is

to have them equally sampled over the whole surface. However, this leads to a high number of parameters, which increases computational complexity. This has been pointed out in [17], where application-specific control point locations have been derived. In the same spirit, we propose to pick control points using the *Farthest Point Sampling* method [21], which uniformly samples the *template* surface maximizing the geodesic distance between the closest points. Note that, this approach also guarantees that control points are placed on the surface only.

3 Experimental Results

In this section we summarize our experimental results obtained on the Bosphorus Dataset [25]. This dataset consists of 4666 face scans with 2D color images and 3D surface points. The ground truth landmark locations are also marked in at most 24 points, giving the opportunity to evaluate our approach using ground truth data. The tests have been made on a randomly generated subset containing 153 pairs of faces with neutral facial expression from different people (*i.e.* we performed *inter-person* registration).

The triangular surface meshes have been created from the Bosphorus 3D point cloud using the *Poisson algorithm* [12] implemented using the CGAL Library [2]. The library implements a *Delaunay refinement* algorithm, allowing to construct meshes at different resolutions [23, 27]. In our tests, we controlled the resolution of the triangular meshes by the maximal radius r of the corresponding Delaunay sphere for each triangle. For the TPS model, we used 64 control points placed on the surface using the *Farthest Point Sampling* strategy. This model has 204 parameters, therefore we constructed a system of 220 equations by choosing M , the maximal order of polynomials, to be 9 for the function set from Eq. (6). Note that, since we have an overdetermined system, it is solved in the least squares sense by the *Levenberg-Marquardt* algorithm. The solver has been initialized by the identity transformation and the integrals have been normalized according to the algorithm described in [24].

The algorithm have been implemented in C++ using the *Levenberg-Marquardt* implementation `levmar` of Lourakis [15]. All tests have been ran on a laptop with Core i5 3.1 GHz architecture.

The results have been quantitatively evaluated based on the average landmark distances between the transformed and the ground truth locations:

$$D_{GT} = \frac{1}{N} \sum_{i=1}^N \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|, \quad (15)$$

where N is the number of available landmarks for a pair of scans, $\mathbf{x}_i \in O_\Delta$ is the ground truth and $\hat{\mathbf{x}}_i = \varphi(\mathbf{z}_i)$, $\mathbf{z}_i \in T_\Delta$ is the transformed landmark position of the i^{th} landmark using the estimated aligning transformation φ . The overall surface alignment accuracy has been also characterized by the maximal root mean square (RMS) error between the closest points of the triangular meshes:

$$D_{RMS} = \max\{RMS(O_\Delta, \varphi(T_\Delta)), RMS(\varphi(T_\Delta), O_\Delta)\}, \quad (16)$$

where

$$RMS(S_1, S_2) = \sqrt{\frac{1}{|V(S_1)|} \sum_{\mathbf{p} \in V(S_1)} \inf_{\mathbf{q} \in S_2} \|\mathbf{p} - \mathbf{q}\|^2},$$

and S_1, S_2 are the corresponding surfaces, while $V(S_1)$ denotes the set of all vertices of S_1 . The RMS function basically estimates the distance between each vertex of the first triangular surface and the closest (not necessarily vertex) point of the second triangular surface. This measure is more accurate than the simple vertex-wise distances and by taking the maximum of the values computed with swapped arguments leads to a symmetric measure between the surfaces. Note that, this value is only determined for the inner parts of the membership functions. In our first experiment, we tried each membership function with meshes generated by $r = 10$ as maximal radius of the enclosing Delaunay sphere. Note that each membership function depends on two parameters: the value of the thresholds (λ_1 and λ_2) and the interpolation method for the areas between the thresholds. For the possible combinations, see Table 1. As we mentioned in Sect. 2.1, the geodesic distance computation needs a specific origin point, which is the nose tip. The coordinate system of the scans in the Bosphorus dataset is established such a way, that the point having the maximal value on the Z axis is a good estimate for the nose tip, thus we used this point in the geodesic distance calculations. The results of this test is shown in Fig. 2. According to the D_{GT} error metric in Eq. (15), the best alignment is provided by using the thresholds from the top 5 % distances. Moreover, we have noticed that the interpolation method has no strong influence on the outcome of the algorithm, therefore we recommend to use the step interpolation.

Table 1. The weight functions used for our experiments

Test Case	Interpolation	λ_1	λ_2
top	linear, step	74.84 %	80.95 %
top 5 %	linear, step	71.07 %	75.82 %

For our next experiments we tried several triangular surface resolutions with $r \in \{2, 3, 5\}$ as maximal radius for the enclosing Delaunay sphere of each triangle. This is necessary because the triangulation will eventually reduce the resolution of the input data by removing points and smoothing the surface. An approximation for the loss in the resolution can be found in Table 2. In this table, we measured the average distance for each neighborhood containing 6 points. The resolution loss effect could be reduced by extracting a more detailed mesh from the input, but this will increase the amount of input data and lead to higher running times. The aim of the current experiment is to find the best runtime over quality ratio. The outline of this test is shown in Fig. 3. We achieved the best results on the $r = 3$ case, with respect to the D_{GT} metric and the runtime of the algorithm. The average D_{GT} error for the test dataset is 6.71mm

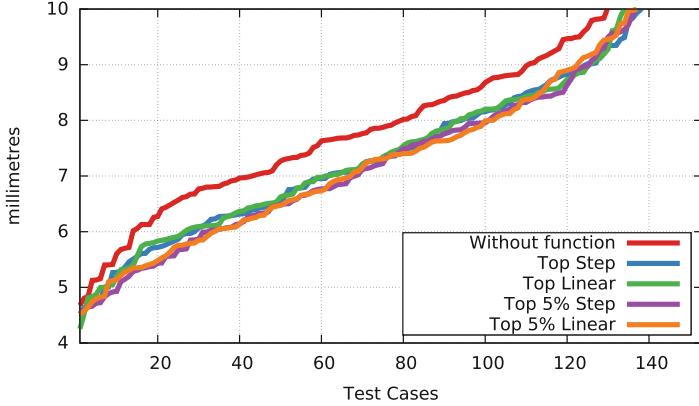


Fig. 2. The average landmark localization error in the first experiment. Each plot shows results with different weight functions. We achieved the best outcome using the thresholds defined by top 5 % distances.

with average running time of 16.4 seconds. Some results are presented in Fig. 5. For the surface alignment accuracy we got $D_{RMS} = 1.76\text{mm}$ (see Fig. 4 for an example). From the results we conclude that the algorithm achieves good results near the areas with significant curvature changes (*e.g.* nose and mouth), however performs poorly near the noisy eyebrows.

Table 2. The resolutions of the input sets expressed as the average Euclidean-distance between the 5 closest points in millimeters.

Point Cloud [25]	$r = 10$	$r = 5$	$r = 3$	$r = 2$
0.78	3.48	3.73	3.10	2.17

In the last experiment, we have compared our results to the point-based registration frameworks in [11] (GMMREG) and [18] (CPD). We used the C++ implementation of these methods available from <http://code.google.com/p/gmmreg> and set the parameters to their default values (within the given Matlab framework). Since the runtime of these algorithms are enormously high for the full original point sets (the average is around 7–8 h for one pair of faces), we ran the experiments using the vertices of the same surface meshes as we used for our algorithm. The results can be found in Fig. 6. We used the top 5 % thresholds with the step interpolation method for this test. The GMMREG algorithm has achieved very good results, slightly outperforming our method and the CPD algorithm on the D_{GT} error, but CPD gives inferior alignment compared to our approach. However, the proposed approach achieved the lowest running time.

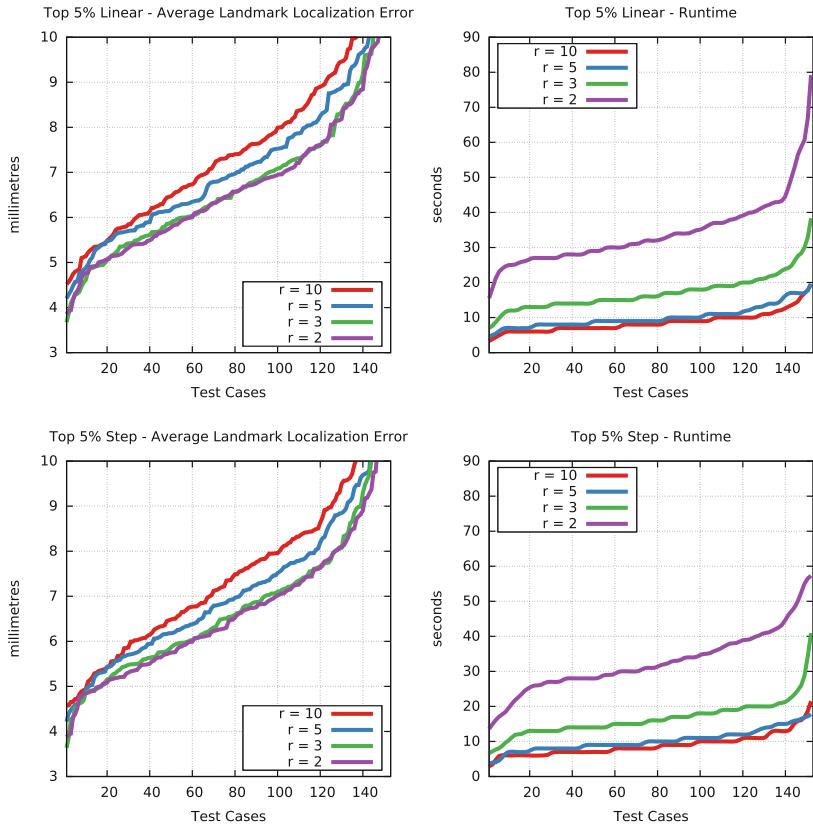


Fig. 3. Average landmark localization errors and running times for each resolutions of $r \in \{2, 3, 5, 10\}$ using the top 5 % thresholds. The localization error determined as the Euclidean distance of the transformed and the ground truth landmark locations. In the first line corresponds to the step, the second line to the linear interpolation method.

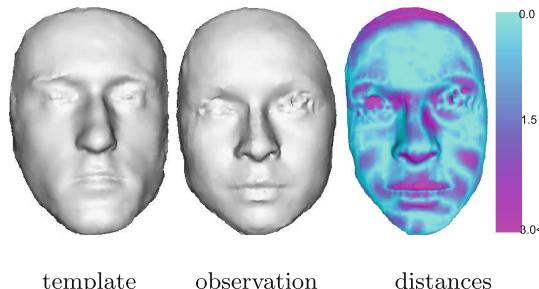


Fig. 4. Example alignment on the Bosphorus dataset. The $D_{RMS} = 1.17\text{mm}$ and the $D_{GT} = 5.4\text{mm}$.

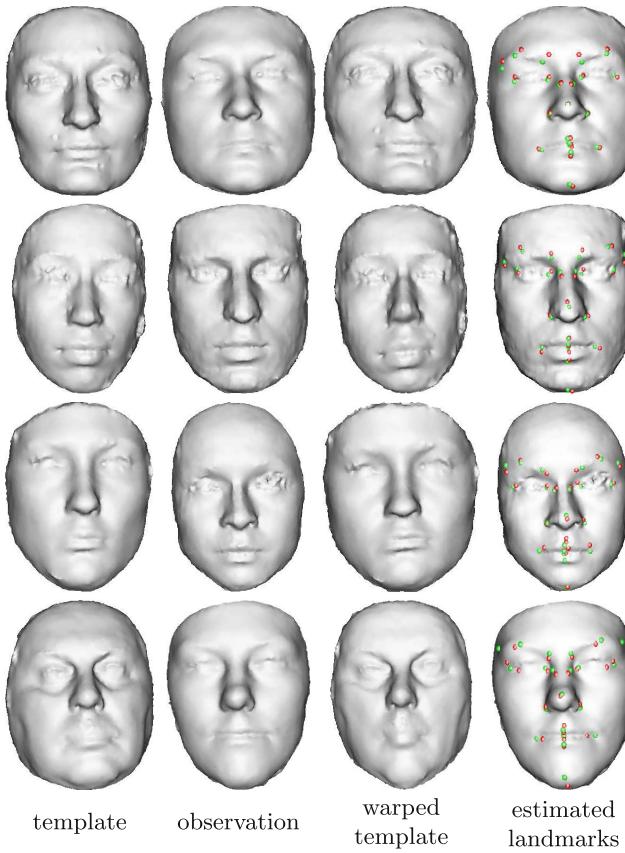


Fig. 5. Landmark detection results on the Bosphorus dataset. In each row we show the *template*, the *observation*, the warped *template* and the localization results, respectively. In the last column green denotes the ground truth position and red is the estimated location. While the proposed approach achieved good results near nose and mouth areas, the highest errors are near the eyebrows. (Color figure online)

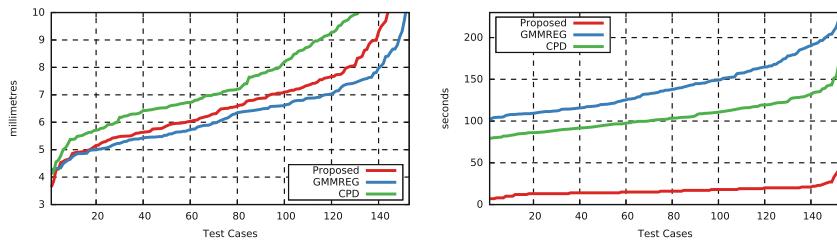


Fig. 6. Comparison between the proposed, the GMMREG and the CPD algorithms. On the left diagram the D_{GT} metric is presented, while on the right we show the running times of each algorithm. The GMMREG slightly outperformed the proposed approach and the CPD algorithm in the landmark localization error, but the proposed approach has the best running time

4 Conclusion

We proposed a novel deformable registration algorithm for aligning human faces. The algorithm is motivated by a recent approach proposed in [7, 24]. The physical deformation is approximated by a parametric TPS model and the parameters are directly obtained as a solution of a system of nonlinear equations. Experimental results on the Bosphorus dataset [25] confirmed the state of the art capabilities of the proposed approach. Furthermore, in terms of computational complexity, our method compares favorably to two recent state-of-the-art methods published in [11] and [18]. While the achieved results are promising, further research aims to enhance the accuracy of the method by introducing face specific priors into the system. Moreover, since the proposed algorithm has many independent components, the running time could be drastically reduced through parallelization on a GPU.

Acknowledgment. The face scans have been obtain from Bosphorus Dataset [25]. We would like to thank the authors of [11] and [18] for providing their implementations for our experiments.

References

- Allen, B., Curless, B., Popović, Z.: The space of human body shapes: Reconstruction and parameterization from range scans. *ACM Trans. Graph.* **22**(3), 587–594 (2003)
- Alliez, P., Saboret, L., Guennebaud, G.: Surface reconstruction from point sets. In: CGAL User and Reference Manual. CGAL Editorial Board, 4.4 edn. (2000). http://doc.cgal.org/4.4/Surface_reconstruction_points_3/index.html#Chapter_Surface_Reconstruction_from_Point_Sets
- Bookstein, F.L.: Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Trans. Pattern Anal. Mach. Intell.* **11**(6), 567–585 (1989)
- Cao, X., Wei, Y., Wen, F., Sun, J.: Face alignment by explicit shape regression. *Int. J. Comput. Vis.* **107**(2), 177–190 (2014)
- Corneanu, C.A., Simn, M.O., Cohn, J.F., Guerrero, S.E.: Survey on rgb, 3d, thermal, and multimodal approaches for facial expression recognition: History, trends, and affect-related applications. *IEEE Trans. Pattern Anal. Mach. Intell.* **38**(8), 1548–1568 (2016)
- Creusot, C., Pears, N., Austin, J.: A machine-learning approach to keypoint detection and landmarking on 3D meshes. *Int. J. Comput. Vis.* **102**(1–3), 146–179 (2013)
- Domokos, C., Nemeth, J., Kato, Z.: Nonlinear shape registration without correspondences. *IEEE Trans. Pattern Anal. Mach. Intell.* **34**(5), 943–958 (2012)
- Heike, C.L., Cunningham, M.L., Hing, A.V., Stuhaug, E., Starr, J.R.: Picture perfect?: Reliability of craniofacial anthropometry using three-dimensional digital stereophotogrammetry. *Plast. Reconstr. Surg.* **124**(4), 1261–1272 (2009)
- Holden, M.: A review of geometric transformations for nonrigid body registration. *IEEE Trans. Med. Imaging* **27**(1), 111–128 (2008)
- Jeni, L.A., Cohn, J.F., Kanade, T.: Dense 3d face alignment from 2d videos in real-time. *Proc. IEEE Int. Conf. Workshops Autom. Face Gesture Recogn.* **1**, 1–8 (2015)

11. Jian, B., Vemuri, B.: Robust point set registration using Gaussian mixture models. *IEEE Trans. Pattern Anal. Mach. Intell.* **33**(8), 1633–1645 (2011)
12. Kazhdan, M., Bolitho, M., Hoppe, H.: Poisson surface reconstruction. In: Proceedings of the Fourth Eurographics Symposium on Geometry Processing, pp. 61–70. SGP 2006, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland (2006)
13. Koehl, P.: Fast recursive computation of 3D geometric moments from surface meshes. *IEEE Trans. Pattern Anal. Mach. Intell.* **34**(11), 2158–2163 (2012)
14. Liang, S., Wu, J., Weinberg, S.M., Shapiro, L.G.: Improved detection of landmarks on 3d human face data. In: Proceedings of 2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 6482–6485, Jul 2013
15. Lourakis, M.: levmar: Levenberg-marquardt nonlinear least squares algorithms in C/C++. <http://www.ics.forth.gr/~lourakis/levmar/>
16. Mian, A., Bennamoun, M., Owens, R.: An efficient multimodal 2d–3d hybrid approach to automatic face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(11), 1927–1943 (2007)
17. Mitra, J., Kato, Z., Martí, R., Oliver, A., Llado, X., Sidibe, D., Ghose, S., Vilanova, J.C., Comet, J., Meriaudeau, F.: A spline-based non-linear diffeomorphism for multimodal prostate registration. *Med. Image Anal.* **16**(6), 1259–1279 (2012)
18. Myronenko, A., Song, X.B., Carreira-Perpiñán, M.Á.: Non-rigid point set registration: Coherent point drift. In: Schölkopf, B., Platt, J.C., Hoffman, T. (eds.) *Proceedings of the Advances in Neural Information Processing Systems*, pp. 1009–1016. MIT Press, Vancouver, British Columbia, Canada, Dec 2006
19. Nair, P., Cavallaro, A.: 3D face detection, landmark localization, and registration using a point distribution model. *IEEE Trans. Multimedia* **11**(4), 611–623 (2009)
20. Perakis, P., Passalis, G., Theoharis, T., Kakadiaris, I.A.: 3D facial landmark detection & face registration. Technical report TP-2010-01, CGL Technical Report (2010)
21. Peyré, G., Cohen, L.: Geodesic methods for shape and surface processing. In: Tavares, J.M.R.S., Jorge, R.M.N. (eds.) *Advances in Computational Vision and Medical Image Processing. Computational Methods in Applied Sciences*, vol. 13, pp. 29–56. Springer, Netherlands (2009)
22. Pozo, J.M., Villa-Uriol, M.C., Frangi, A.F.: Efficient 3D geometric and Zernike moments computation from unstructured surface meshes. *IEEE Trans. Pattern Anal. Mach. Intell.* **33**(3), 471–484 (2011)
23. Rineau, L., Yvinec, M.: A generic software design for Delaunay refinement meshing. *Comput. Geom.* **38**(12), 100–110 (2007)
24. Sánta, Z., Kato, Z.: Correspondence-less non-rigid registration of triangular surface meshes. In: *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2275–2282. IEEE, Portland, Oregon, June 2013
25. Savran, A., Alyüz, N., Dibeklioğlu, H., Çeliktutan, O., Gökberk, B., Sankur, B., Akarun, L.: Bosphorus database for 3D face analysis. In: Schouten, B., Juul, N.C., Drygajlo, A., Tistarelli, M. (eds.) *BIOID 2008. LNCS*, vol. 5372, pp. 47–56. Springer, Heidelberg (2008)
26. Sharifi, A., Jones, R., Ayoub, A., Moos, K., Walker, F., Khambay, B., McHugh, S.: How accurate is model planning for orthognathic surgery? *Int. J. Oral Maxillofac. Surg.* **37**(12), 1089–1093 (2008)
27. Shewchuk, J.R.: Delaunay refinement algorithms for triangular mesh generation. *Comput. Geom.* **22**(13), 21–74 (2002). 16th ACM Symposium on Computational Geometry

28. Sotiras, A., Davatzikos, C., Paragios, N.: Deformable medical image registration: A survey. *IEEE Trans. Med. Imaging* **32**(7), 1153–1190 (2013)
29. Sukno, F.M., Waddington, J.L., Whelan, P.F.: 3-d facial landmark localization with asymmetry patterns and shape regression from incomplete local features. *IEEE Trans. Cybern.* **45**(9), 1717–1730 (2015)
30. Unzueta, L., Pimenta, W., Goenetcx, J., Santos, L.P., Dornaika, F.: Efficient generic face model fitting to images and videos. *Image Vis. Comput.* **32**(5), 321–334 (2014)
31. Wahba, G.: Spline models for observational data. *Soc. Ind. Appl. Math.* (1990)
32. Zagorchev, L., Goshtasby, A.: A comparative study of transformation functions for nonrigid image registration. *IEEE Trans. Image Process.* **15**(3), 529–538 (2006)
33. Zimmermann, H.J.: *Fuzzy Set Theory - and Its Applications*. Springer, Heidelberg (2001)