Non-rigid Shape Registration:
A Single Linear Least Squares Framework⋆

Mohammad Rouhani and Angel D. Sappa

Computer Vision Center,
Edifici O, Campus UAB,
08193 Bellaterra, Barcelona, Spain
{rouhani,asappa}@cvc.uab.es

Abstract. This paper proposes a non-rigid registration formulation capturing both global and local deformations in a single framework. This formulation is based on a quadratic estimation of the registration distance together with a quadratic regularization term. Hence, the optimal transformation parameters are easily obtained by solving a liner system of equations, which guarantee a fast convergence. Experimental results with challenging 2D and 3D shapes are presented to show the validity of the proposed framework. Furthermore, comparisons with the most relevant approaches are provided.

1 Introduction

The shape registration problem has been largely studied in the literature and represents a fundamental problem in different computer vision and image processing applications. It aims at recovering a set of transformation parameters that brings a given data set as close as possible to the corresponding model set. In the rigid case, also known as shape alignment, it involves rotations and translations. While in the non-rigid shape registration case, in addition to the rotation and translation parameters it includes a deformation stage. The development of formulations able to tackle the non-rigid registration case are attracting the interest of the research community. They are capable of handling situations with shape distortions due to deformation, noise or missing parts (e.g., [1], [2], [3]).

In general, most of the approaches proposed for non-rigid shape registration follow a two step scheme, where first a global rigid alignment is performed and then a local process deforms the shape of data set towards the given model set. The differences between the methods in the literature mainly lie in the way they formulate these two steps (most relevant approaches are summarized in the next section). Variations to this global-rigid/local-non-rigid strategy have been also proposed in the literature. For instance, an alternative has been recently introduced in [4], where shape rigidity is firstly considered locally and then a global shape deformation process is performed.

On the contrary to previous approaches, the current work proposes to use a single formulation to tackle both, global and local alignment and deformation. The main features of the current work are as follow: (i) a robust distance approximation based on local curvature information is used for non-rigid registration; (ii) the proposed objective function is in the linear least squares form, hence it can be solved by a linear system of equations; (iii) the proposed method captures all deformation from rigid to non-rigid by the same framework; there is no need to use different steps to capture global and local deformations separately; (iv) unlike the sign distance field, the proposed function is not discretely approximated.

The rest of the paper is organized as follows. Section 2 presents the most relevant related works. The proposed technique is presented in Section 3. Section 4 gives experimental results using 2D and 3D shapes; additionally comparisons with state of the art are presented. Finally, conclusions and future work are detailed in Section 5.

2   Related Work

During the last decade several approaches have been proposed in the literature to tackle the non-rigid shape registration problem (e.g., [5], [6], [7], [8], [9]). These approaches can be classified into different categories according to: i) the way in which the distance between the data and model sets is approximated (e.g., point wise [10], distance field [1]); ii) the transformation used for mapping data set towards model set (e.g., Thin Plate Splines (TPS) [11]; Free Form Deformations (FFD) [1]; Laplacian deformation [12]); iii) the approach used to compute the best set of transformation parameters (e.g., iterative algorithm [13], global optimization [8]).

Chui and Rangarjan in [11] present a Robust Point Matching (RPM) for non-rigid registration using TPS. Their proposed method is based on a soft assignment technique to avoid the correspondence search. This technique addresses the common problem in feature based matching when there is no counterparts for some points. They propose an algorithm to devote appropriate weights to describe the correspondence of each data point as a linear combination of the points in the model set. Moreover, it includes an annealing parameter that allows to control the influence domain of points to each other.

Jian and Vemuri in [14] present a probabilistic model for point set registration. They describe both data set and model set by mixture of Gaussians and the problem is treated as an alignment problem between two density functions. This modeling provides a more robust method for non-rigid registration through TPS. Although the formulation they provide avoids the correspondence search like the ICP kind algorithm ([13], [10]) and the gradient information can be explicitly derived and exploited in an optimization stage, it considers all the combinations between the data points and the model points which is quite expensive.

Huang et al. [1] propose a region based algorithm to measure the similarity between distance fields of each data set and model set. They used FFD to capture
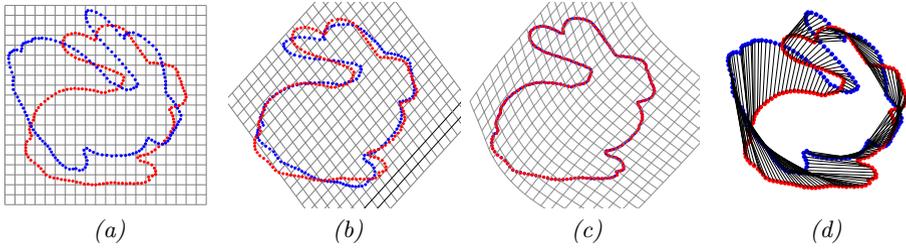
Fig. 1. The proposed method (SD-FFD): (*a*) initial configuration of the data set (blue) and the model set (red); (*b*) result with $\lambda = 10^5$ (at iteration 15); (*c*) result with $\lambda = 1$ (at iteration 22); (*d*) the optimal FFD control lattice results in a very dense correspondence, not only on the boundary but over the whole space.

the non-rigid deformation. The final similarity function is described in a form of a double integral over the region. The smoothness term in the formulation can be analytically computed, but the data term should be done numerically over the region. The data term compares the model distance field and the data distance field after applying the deformation; however, theoretically, it is not a distance field anymore. Finally, the whole similarity distance is optimized through the gradient descent algorithm. This method is later extended by Taron et al. [5] to consider the uncertainty of the point sets encoded in the covariance matrix.

Signed distance fields (SDFs) are used in [4] to capture the local transformation in small sampling grids. These sampling grids are defined by the FFD control lattice. Their proposed algorithm uses the local rigid transformation in these grids to define the global deformation over the whole region. These local rigid motions are firstly found by optimizing a non-linear function that measures the differences between the two SDFs. Then, the translation and rotation parameters in the sampling grid are used to guide the FFD control lattice. This method is computationally expensive since it performs many non-linear optimizations in each iteration to find the local motions. Moreover, no global regularization term can be applied on the FFD control lattice.

As mentioned above there have been different approximations of the distance between model and data sets. Choosing a proper error term leads to a precise and fast registration algorithm. Most of the methods previously reviewed result in a non-linear error term, which must be iteratively optimized. In the current work we exploit a quadratic distance approximation in the non-rigid registration problem. The presented objective function is in the linear least squares form and can handle both rigid and non-rigid deformations with the same framework. Figure 1 illustrates how the proposed method handles both the rigid alignment (Fig. 1(*b*)) and the non-rigid deformation (Fig. 1(*c*)) just by relaxing the regularization term. This term controls the rigidity of deformation during the evolution. Thanks to the proposed objective function each iteration is linearly solved. Hence, the whole framework has a fast convergence. Furthermore, the optimal deformation provides a dense correspondence between the given data and model set.

3 Proposed Approach

This section presents the main elements of the proposed framework. Firstly, in Section 3.1, the approximation error used to estimate the distance between the current data set and model set is presented. This distance is a quadratic term based on the curvature information of the points in the model set. Our contribution is to use this distance to capture both rigid and non-rigid deformations by means of the Free Form Deformation. In Section 3.2 this deformation space is defined. Finally, Section 3.3 details how both rigid and non-rigid registration problems can be solved in the same framework. In this section we propose a novel objective function to find the deformation parameters. This function is in the least squares form and it can be easily optimized by solving a linear system of equations.

3.1 Registration Error (SD)

All the registration methods seek for the best transformation parameters to move the given *data set* (source shape) $\mathcal{S} = \{\mathbf{s}_i\}_1^{N_d}$ close to the *model set* (target shape) $\mathcal{T} = \{\mathbf{t}_i\}_1^{N_m}$. As mentioned in the previous Section, all registration methods can be classified based on the distance used to measure the closeness, the transformation to move the data set, and the optimization method that finds the best transformation parameters. The first and most important matter is how to choose a proper and precise distance to define the registration error term.

Approaches using the precise geometric distance between model and data sets have been proposed in the literature. For instance, a well known example is the Iterative Closest Point (ICP) algorithm. It moves the data set in each iteration based on a simple criterion: for the given data point it searches for the closest corresponding model point (or *foot-point*). Therefore, the distance used by ICP is a point-to-point distance and ICP performs a Point Distance Minimization (PDM) in each iteration to find the best transformation parameters. Figure 2(a) illustrates a simple case where ICP is stuck in a local minima. Some of the data points in the figure lie on the curve passing through the model set; hence their distance to the model set must be quite low, but ICP devotes a quite high distance to these points since the model set is quite sparse. If there could be a better approximation for the distance, the ICP would devote more weights to the data point which are still far from the model point. More elaborated approaches have been also proposed using ICP philosophy [15]; in [10] ICP is used in a Tangent Distance Minimization (TDM) framework.

Implicit descriptions like distance fields provide another metric to measure the distance between the data and model sets. In these techniques both model set and data set [1], or only the model set [17], are described by signed distance fields at first. Then, the registration error is measured through these distance fields instead of the point sets. Unfortunately, these methods require expensive computation to build the distance fields over the whole region. In addition, since the distance fields are made discretely, the precision is up to a specific point. Finally, using distance fields for non-rigid registration results in a non-linear optimization function, which is usually solved by a gradient descent algorithm.

Illustration with legend:
● Data set
✳ Model set
--- Model Shape

(a)   (b)
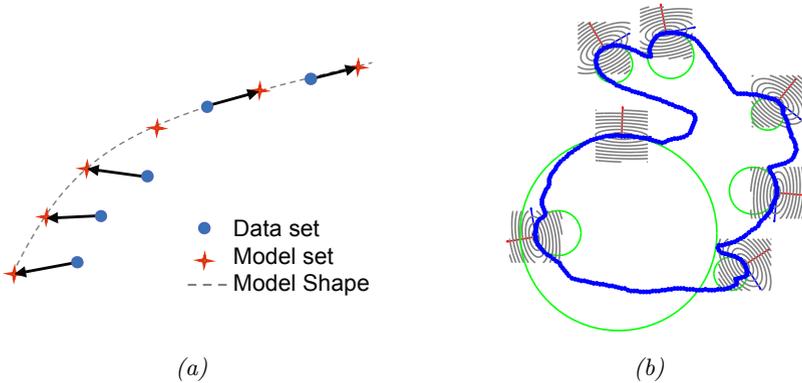
Fig. 2. (*a*) Illustration of a point-to-point distance based approach (e.g., ICP [10]). (*b*) Local quadratic approximation [16] used in current work.

In this paper we use a quadratic approximation of the geometric distance in order to define the registration error term in the least squares form. This distance is based on the curvature information in the model sets. Consider the data point $\mathbf{s}_i$ with its closest corresponding model point $\mathbf{t}_j$. Then the Squared Distance (SD) of $\mathbf{s}_i$ to the whole model set $\mathcal{T}$ can be approximated as follows:

$$SD(\mathbf{s}_i, \mathcal{T}) = \frac{d}{d - \rho}[(\mathbf{s}_i - \mathbf{t}_j).\mathbf{T}_j]^2 + [(\mathbf{s}_i - \mathbf{t}_j).\mathbf{N}_j]^2, \tag{1}$$

where $\mathbf{T}_j$ and $\mathbf{N}_j$ are the unit tangent and unit outer normal, respectively, defined in the Frenet frame at $\mathbf{t}_j$. The value ρ is the curvature radius at the model point $\mathbf{t}_j$ and d is the signed distance between the data point $\mathbf{s}_i$ and the model point $\mathbf{t}_j$. The sign of d is positive if $\mathbf{s}_i$ and $\mathbf{N}_j$ lie on the same side and is negative otherwise [16].

The distance approximation in (1), referred to as SD, works with the Frenet frame at the foot-point $\mathbf{t}_j$. It projects the data point on the normal and tangent vectors firstly and the final approximation will be quadratic with respect to these projections. This final property of SD is very important and fits our need, since it results in a least squares form. In the special case, where the data point is along the normal at the foot-point, the first quadratic term vanishes and the distance will be equal to $|\mathbf{s}_i - \mathbf{t}_j|^2$, which is the squared point to point distance. In another special case, where the curvature of model set at $\mathbf{t}_j$ is zero, the first quadratic term vanishes again, and the projection of the data point on the normal will be the SD approximation of quadratic distance. Figure 2(*b*) shows an illustration depicting quadratic approximations for a few points of a given 2D shape.

3.2   Deformation Space (FFD)

Rigid transformation is able to align the global appearance of the objects. Hence, in order to capture the local deformation we should use a more flexible family of

transformations. In the current work we propose to use a Free Form Deformation (FFD) to describe any transformation from global (rigid) to local (non-rigid). FFD has been already used in the computer vision and graphics communities in the form of iFFD (incremental FFD); in other words, they were considered only for capturing local deformation. In this work, thanks to the metric selected in Section 3.1, a single framework is used to apply FFD to describe the deformation space; without loss of generality let us consider the 2D case, where the FFD describes a deformation field by means of the *control lattice* $\{\mathbf{P}_{ij}\}_{M \times N}$ in 2D:

$$\mathbf{L}(x, y) = \sum_{i=1}^{M} \sum_{j=1}^{N} \mathbf{P}_{i,j} B_i(x) B_j(y), \tag{2}$$

where $\{B_i(x)B_j(y)\}$ are cubic spline basis functions to guarantee C^2 continuity. In our implementation we use a square control lattice ($M = N$) covering the unit square $[0, 1]^2$. The B-Spline knot sequence is uniform with a step of $\Delta = 1/(N - 3)$.

Since we consider a square control lattice, both sets of basis functions behave similarly. Having considered a row-by-row order, we can represent the control lattice and the basis functions in a vector form:

$$\mathbf{L}(x, y) = \begin{bmatrix} \mathbf{p}_x^T \mathbf{m}(x, y) \\ \mathbf{p}_y^T \mathbf{m}(x, y) \end{bmatrix} = \begin{bmatrix} \mathbf{p}_x^T \\ \mathbf{p}_y^T \end{bmatrix} \mathbf{m}(x, y), \tag{3}$$

where $\mathbf{m}(\mathbf{x})$ is the vector form of the monomials $\{B_i(x)B_j(y)\}$, and $\mathbf{p}_x$, $\mathbf{p}_y$ are the vector form of the x and y components of control lattice.

The FFD definition in (2) can be simplified through the *blending functions* which are cubic patches on $[0, 1]$, which builds up the B-Spline basis function by assembling together:

$$\begin{array}{ll} b_0(u) = (1 - u)^3/6, & b_1(u) = (3u^3 - 6u^2 + 4)/6, \\ b_2(u) = (-3u^3 + 3u^2 + 3u + 1)/6, & b_3(u) = u^3/6. \end{array} \tag{4}$$

Then an equivalent definition of FFD will be achieved that is computationally useful:

$$\mathbf{L}(x, y) = \sum_{r=0}^{3} \sum_{s=0}^{3} \mathbf{P}_{i+r,j+s} b_r(u) b_s(v), \tag{5}$$

where the indices start from:

$$i = \lfloor x/\Delta \rfloor + 1, \quad j = \lfloor y/\Delta \rfloor + 1, \tag{6}$$

and the given coordinates in XY will be mapped in UV as:

$$\begin{aligned} u &= x/\Delta - \lfloor x/\Delta \rfloor, \\ v &= y/\Delta - \lfloor y/\Delta \rfloor. \end{aligned} \tag{7}$$

This definition provides us with the computational efficiency useful for calculating the monomial matrix. Note that the value $b_r(u).b_s(v)$ will be accumulated in the proper cell of the monomial $\mathbf{m}(x, y)$ corresponding to $B_{i+r}(x).B_{j+s}(y)$.

The Free Form Deformation in (2) has $2N^2$ degrees of freedom due to the free movement of the control lattice. It should be indicated that this movement should be controlled in order to have a meaningful deformation. Otherwise we will have only a 2D to 2D mapping. Moreover, using this FFD formulation together with the SD may lead us to a singularity problem in the Least Squares solution presented in the next section. In order to avoid these problems a regularization term must be considered as well.

In the current work a global tension term is considered to regularize the control lattice. On the contrary to the conventional regularization term used in iFFD, which measures the first order changes of iFFD, we use a second order term. This term, similarly to [18], is computed by measuring the curvature of $\mathbf{L}$ over the whole domain:

$$T(\mathbf{P}) = \iint_{XY} \|\mathbf{L}_{xx}\|^2 + 2\|\mathbf{L}_{xy}\|^2 + \|\mathbf{L}_{yy}\|^2 dxdy. \tag{8}$$

Since the vector field $\mathbf{L}(x,y)$ is a linear function of $\mathbf{P}$, the whole regularization term will be a quadratic function of $\mathbf{P}$. Using the vector form of $\mathbf{L}$ in (3) the regularization term can be simplified as follows:

$$T(\mathbf{P}) = \mathbf{p}_x^T \mathbf{H} \mathbf{p}_x + \mathbf{p}_y^T \mathbf{H} \mathbf{p}_y, \tag{9}$$

where matrix $\mathbf{H}$ is a $N^2 \times N^2$ symmetric matrix including the integral of basis functions' derivatives:

$$\mathbf{H} = \iint_{XY} \mathbf{m}_{xx}\mathbf{m}_{xx}^T + 2\mathbf{m}_{xy}\mathbf{m}_{xy}^T + \mathbf{m}_{yy}\mathbf{m}_{yy}^T dxdy. \tag{10}$$

This matrix can be analytically constructed once the size of control lattice is known. Hence it can be computed off-line and be used during the algorithm.

3.3   SD-FFD: A Novel Non-Rigid Registration

So far the registration error (SD) as well as the transformation model (FFD) are defined, where the first one defines the fitting term to measure the external energy and the second one defines the solution space to describe the deformation. Assembling these two terms will result in a novel non-rigid registration method:

$$\varphi(\mathbf{P}) = \sum_{i=1}^{N_d} SD(\mathbf{L}(\mathbf{s}_i), \mathcal{T}) + \lambda T(\mathbf{P}). \tag{11}$$

Our proposed registration function is a function of the control lattice $\mathbf{P}$ consisting of the data fitting term and the regularization term. As defined in the pervious section, the regularization term $T(\mathbf{P})$ is quadratic with respect to $\mathbf{P}$. In addition, since SD is quadratic with respect to the given coordinates and $\mathbf{L}(\mathbf{s}_i)$ is linear with respect to $\mathbf{P}$, the whole registration function in (11) is linear in terms of the control lattice coordinates.

Thanks to vector form representation our proposed registration distance can be reformulated as follows:

$$\varphi(\mathbf{p}_x, \mathbf{p}_y) = \sum_{i=1}^{N_d} \omega_i [(\begin{bmatrix} \mathbf{p}_x^T \\ \mathbf{p}_y^T \end{bmatrix} \mathbf{m}(\mathbf{s}_i) - \mathbf{t}_j).\mathbf{T}_j]^2$$

$$+ \sum_{i=1}^{N_d} [(\begin{bmatrix} \mathbf{p}_x^T \\ \mathbf{p}_y^T \end{bmatrix} \mathbf{m}(\mathbf{s}_i) - \mathbf{t}_j).\mathbf{N}_j]^2 + \lambda(\mathbf{p}_x^T \mathbf{H} \mathbf{p}_x + \mathbf{p}_y^T \mathbf{H} \mathbf{p}_y). \quad (12)$$

It is now clear that the function φ is quadratic with respect to $\mathbf{p}_x$ and $\mathbf{p}_y$; hence vanishing the partial derivatives $\frac{\partial \varphi}{\partial \mathbf{p}_x}$ and $\frac{\partial \varphi}{\partial \mathbf{p}_y}$ result in two linear system of equations $\mathbf{A}_x \mathbf{p}_x = \mathbf{b}_x$ and $\mathbf{A}_y \mathbf{p}_y = \mathbf{b}_y$ where:

$$\mathbf{A}_x = \lambda \mathbf{H} + \sum_{i=1}^{N_d} (\omega_i \mathbf{T}_j^{x^2} + \mathbf{N}_j^{x^2}) \mathbf{m}(\mathbf{s}_i) \mathbf{m}(\mathbf{s}_i)^T \quad (13)$$

$$\mathbf{b}_x = \sum_{i=1}^{N_d} (\omega_i \mathbf{T}_j^{x^2} + \mathbf{N}_j^{x^2})(\mathbf{t}_j^x) \mathbf{m}(\mathbf{s}_i).$$

Similarly, the coefficient matrix $\mathbf{A}_y$ and the right hand vector $\mathbf{b}_y$ corresponding to the y coordinate of the control lattice can be obtained.

Therefore, our proposed method, SD-FFD, finds the optimal control lattice through solving two linear system of equations in each iteration. In order to converge to the global minimum, we can start with a high regularization parameter λ and decrease it gradually. It must be mentioned that SD-FFD, unlike other methods, neither uses implicit distance field, which is computationally expensive, nor relies on the single corresponding foot-point point. SD-FFD uses the local curvature information around the foot-point and use this information to build up a quadratic function.

4   Experimental Results

The performance of the proposed approach has been evaluated and compared with state of the art algorithms. Several 2D and 3D shapes, obtained from public databases ([19], [20] and [21]), have been registered. In all the cases the data set corresponds to a deformed shape of the model set; as an exception, Fig. 3(*bottom*) shows the result when shapes from different objects are registered together—data set corresponds to a *Donkey* 2D shape, while model set to a *Cat* 2D shape. Figure 3 shows seven illustrations of 2D shapes registered with the proposed approach. Figure 3(*a*) presents the initial configurations where not only deformation but also rotations and translations between model and data sets can be appreciated. In the current implementation the regularization parameter (λ), which somehow represents the registration rigidity, was automatically tuned. It starts with a high regularization value ($\lambda = 10^5$, see illustrations in Fig. 3(*b*)), which is mainly devoted to tackle the alignment problem. Once the ratio of registration error

between consecutive iterations is below a given threshold λ is divided by 10; this relaxation is performed till $\lambda = 1$. Figure 3(c) depicts intermediate results, while Fig. 3(d) presents results after convergence is reached.

All the 2D examples presented above (Fig. 1 and Fig. 3) have been used to compare the results from the proposed approach with two state of the art algorithms (i.e., [1] and [4]). Additionally, the performance of the proposed framework is evaluated by using a point wise based approach. In other words, instead of using the quadratic approximation of the geometric distance (Section 3.1) a precise point-to-point distance is considered. This second approach is only implemented for comparisons and will be referred to as ICP-FFD. This ICP-FFD has been chosen since it is simple and can be derived from SD-FFD as a special case. During the comparisons, the techniques iterate till the maximum number of iterations (#Iter=50) is reached or the relative registration error is smaller than a given threshold (in the current implementation $\epsilon < 0.001$); relative registration error is defined as: $\epsilon = |E_t - E_{t-1}|/E_t$, where E_t refers to the registration error between the model and data set at iteration t. The registration error is used as a quantitative value for the comparisons and it is computed by accumulating the residual error, in a point wise manner, from data set to a *reference model set*. The reference model set corresponds to a highly detailed description of the model set (it contains on average ten times the number of points in the model set). Residual errors are computed by finding the nearest point in between the registered data set and the reference model set. Table 1 depicts the number of points in the data set (N_d), the number of points in the model set (N_m), the registration error (*Error*) and the number of iterations (#*Itr*) for all the algorithms tested during the comparisons. It should be highlighted that the proposed approach reaches the best registration in the lowest number of iterations.

Figure 4 illustrates a qualitative comparison between the four different techniques mentioned in Table 1. In the case of *Camel* all other techniques get stuck in a local minima. The first two methods [1] and [4] are based on distance fields and the ICP-FFD is based on the point to point distance. Since our method

Table 1. Comparisons of non-rigid shape registration algorithms

Figure	N_d	N_m	Huang et al. [1]		Fujiwara et al. [4]		Comp.: ICP-FFD		Prop. App.: SD-FFD	
			Error	#*Itr*	*Error*	#*Itr*	*Error*	#*Itr*	*Error*	#*Itr*
Fig. 1	341	341	7.39	37	3.05	32	0.72	48	0.24	37
Fig. 3(1^{st}row)	415	361	2.02	29	1.92	26	1.52	31	1.21	27
Fig. 3(2^{nd}row)	297	293	2.25	25	1.78	29	1.29	38	1.17	27
Fig. 3(3^{rd}row)	272	253	1.74	21	1.72	27	1.85	39	1.76	12
Fig. 3(4^{th}row)	376	299	2.69	24	1.87	27	1.14	37	1.11	31
Fig. 3(5^{th}row)	417	358	1.89	32	1.22	28	1.13	30	1.10	22
Fig. 3(6^{th}row)	598	535	5.97	32	4.52	28	13.37	51	2.38	43
Fig. 3(7^{th}row)	485	361	3.71	25	2.89	32	2.51	55	2.09	38

(a)   *(b)*   *(c)*   *(d)*   *(e)*
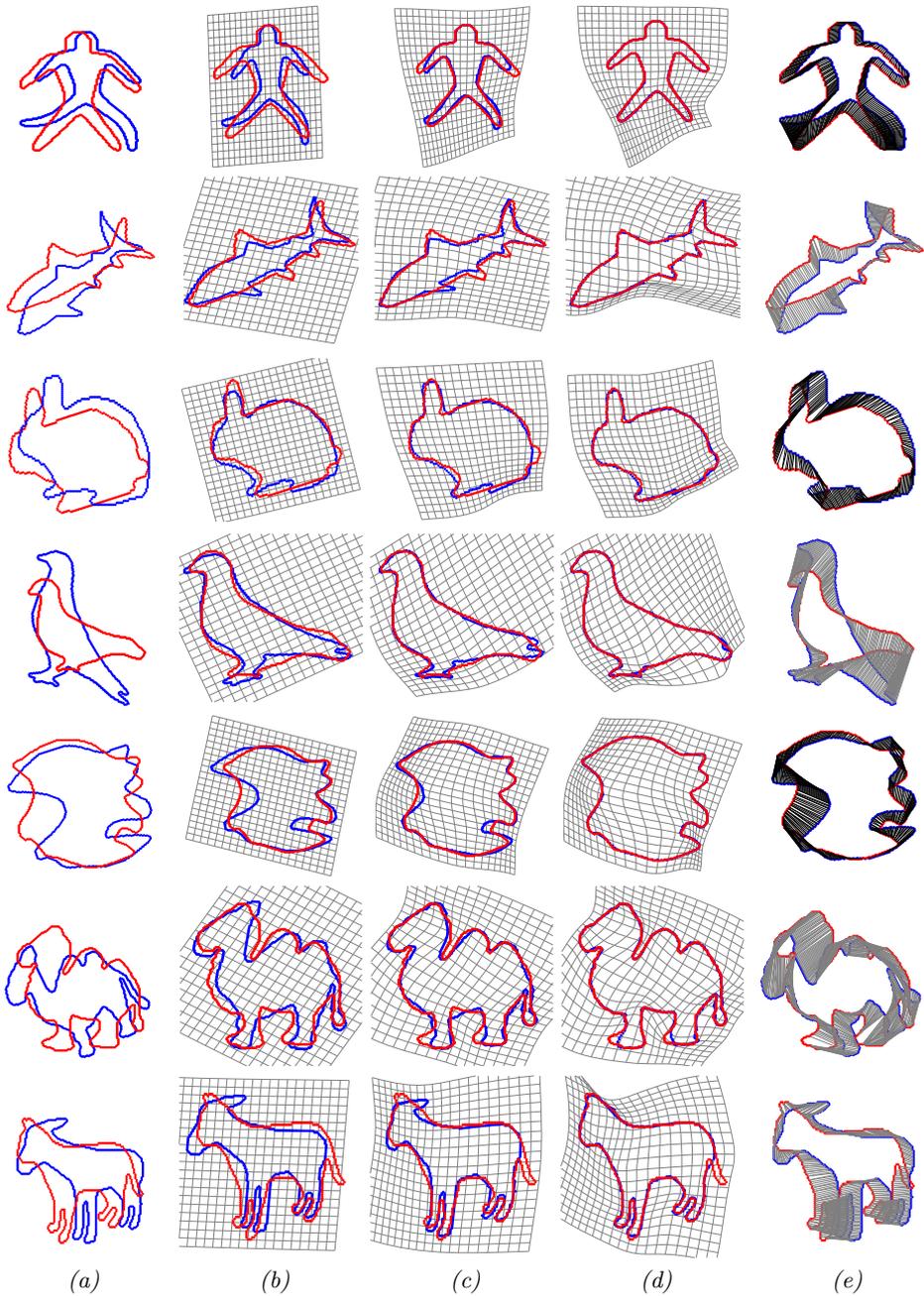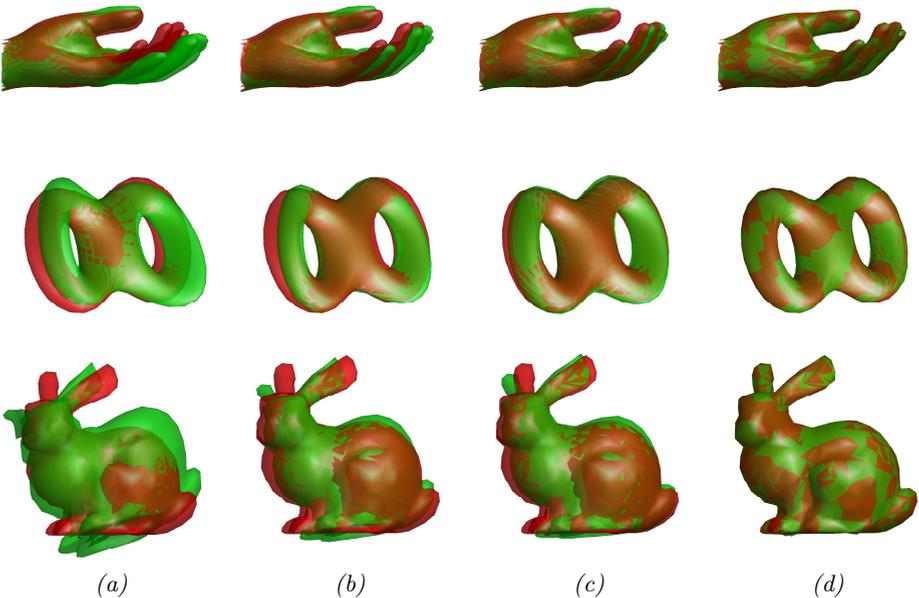
Fig. 3. Registration results of 2D shapes ([19], [20]) using the proposed approach (SD-FFD): (*a*) initial configurations of data sets (blue) and model sets (red); (*b*) results with $\lambda = 10^5$; (*c*) results with $\lambda = 10^2$; (*d*) results with $\lambda = 1$; (*e*) the obtained FFD control lattice results in a very dense correspondence over the whole region.
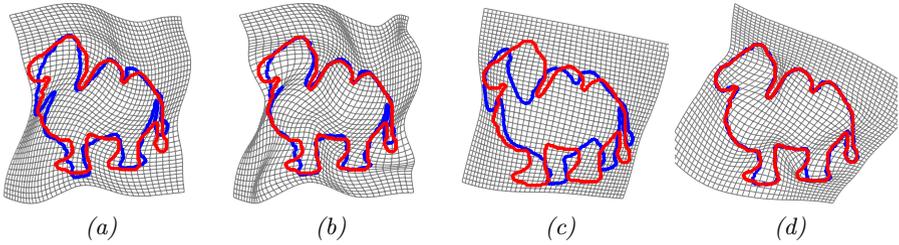
Fig. 4. Visual comparison: registration results of a 2D shape [20] using: (*a*) distance field [1]; (*b*) locally rigid globally non-rigid [4]; (*c*) ICP-FFD; (*d*) proposed approach (SD-FFD)
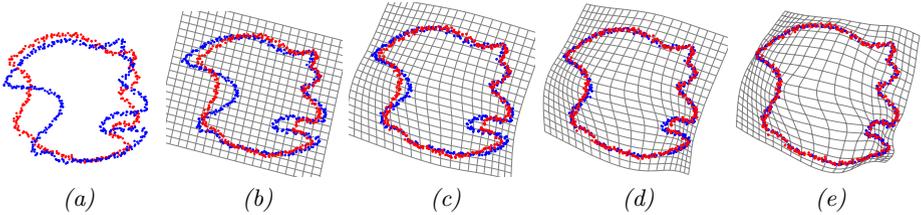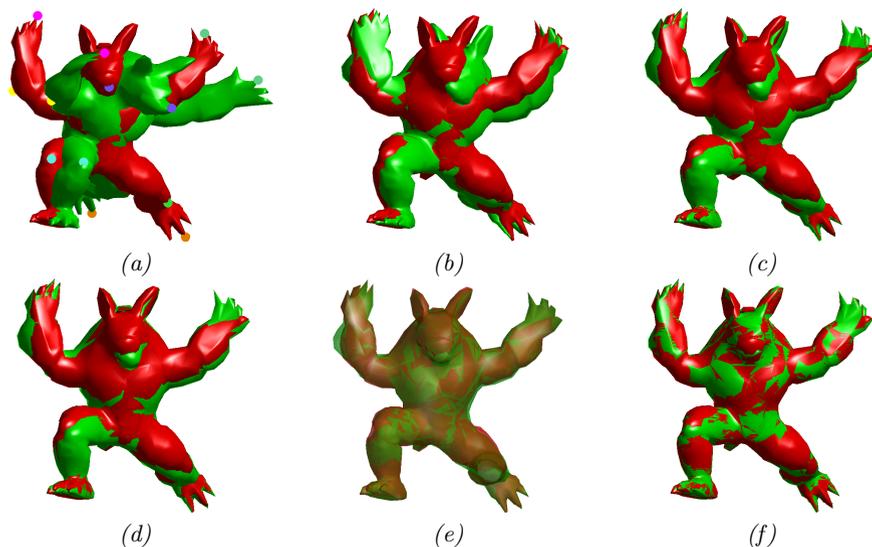

Fig. 5. Noisy case: registration results of a noisy 2D shape [20] using the proposed approach (SD-FFD); $\lambda \in \{10^5, 10^3, 10^2, 1\}$ from (*b*) to (*e*) respectively

Fig. 6. Registration results of 3D shapes using the proposed approach (SD-FFD): (*a*) initial configurations of data sets (green) and model sets (red); (*b*) results with $\lambda = 10^4$; (*c*) results with $\lambda = 10^2$; (*d*) Final registration results showing the blending of the two shapes (data & model)

exploits the curvature information it has a better estimation on the distance and converges to a better results. Moreover, this measurement can tolerate noise disruption on the data. Figure 5 presents a 2D data (*Misk*) disrupted by 5% Gaussian noise. As in the previous examples, we, we start with a high regularization ($\lambda = 10^5$) that easily tackles noise and then it decreases gradually. It should be mentioned that in our implementation we use a local quadratic fitting to approximate the curvature value.

The proposed approach has been also evaluated using public 3D shapes from [19]. Figure 6 presents three examples of model sets, together with their corresponding deformed data sets, which were registered with the proposed approach (SD-FFD). Data sets were obtained by deforming the given model sets. In the case of the hand Fig. 6(*top*), the data set was obtained by opening the model set using a Laplacian deformation [12]; the data set corresponding to the eight-like shape Fig. 6(*middle*) has been obtained by twisting and deforming the model set's shape; finally, in the case of *Bunny* [21], the data set corresponds to a Laplacian deformation that moves down both ears and several distortions of body's parts from the model set (mainly on the back side). Figure 6(*a*) shows the initial configurations where data sets are rotated and translated from the model set, in addition to the deformations mentioned above. Intermediate results, obtained with $\lambda = 10^4$ and $\lambda = 10^2$ are presented in Fig. 6(*b*) and Fig. 6(*c*) respectively. Final registration results are depicted in Fig. 6(*d*). The accuracy of the registration from the proposed approach (SD-FFD) can be appreciated from the blending of the two surfaces.

(a)   *(b)*   *(c)*

(d)   *(e)*   *(f)*

Fig. 7. 3D case: registration results of a deformed 3D shape [19] using the proposed approach (SD-FFD); (*a*) initial configuration with the corresponding feature points; (*b*) result in iteration 2 with ($\lambda = \mu = 10^3$); (*c*) iteration 5 with ($\lambda = 10^2$); (*d*) iteration 10 with ($\lambda = 10$); (*e*) iteration 15 ($\mu = 10^2$); (*f*) iteration 25 with ($\lambda = 1, \mu = 10$)

In all the implementations above no feature constraints are used, but despite that our proposed method converges to the optimal configuration. Our proposed formulation in (12) can follow previous works to incorporate feature constraints. A feature constraint forces the FFD to move some special source points $\{\hat{\mathbf{s_k}}\}$ to their corresponding target points $\{\hat{\mathbf{t_k}}\}$. Figure 7(a) shows 8 feature points and their correspondences with the same color. The feature term can be easily formulated as $\mu \sum_k \|\mathbf{L}(\hat{\mathbf{s_k}}) - \hat{\mathbf{t_k}}\|^2$ where μ is the feature parameter. This term is quadratic with respect to the FFD parameters, hence adding it to (12) results in a quadratic term that can be minimized through a linear least squares form. Figure 7 depicts the evolution of SD-FFD starting from a high regularization ($\lambda = 10^3$) and a high feature parameter ($\mu = 10^3$). Moreover, it shows how these parameters gradually decrease to give more importance to the distance term.

5   Conclusions

This paper presents a novel formulation to tackle the non-rigid shape registration problem. It is based on both a quadratic estimation term, which measures the registration distance and a quadratic regularization term, which controls the deformation of the data set towards the model set. The whole formulation can be solved in a single least squares framework. In summary, in this work: (i) a robust distance approximation based on local curvature information is used for non-rigid registration; (ii) the proposed objective function can be solved by a linear system of equations; (iii) all deformation from rigid to non-rigid are captured by the same framework; there is no need to use different steps to capture global and local deformations separately. Experimental results and comparisons with challenging 2D and 3D shapes are provided showing the validity of the proposed approach as well as the speed of convergence.

References

1. Huang, X., Paragios, N., Metaxas, D.: Shape registration in implicit spaces using information theory and free form deformations. IEEE Trans. on Pattern Analysis and Machine Intelligence 28, 1303–1318 (2006)
2. Brown, B., Rusinkiewicz, S.: Global non-rigid alignment of 3-D scans. ACM Trans. Graph. 26, 21 (2007)
3. El Munim, H., Farag, A.: Shape representation and registration using vector distance functions. In: Proc. IEEE International Conference on Computer Vision and Pattern Recognition, Minneapolis, Minnesota, USA (2007)
4. Fujiwara, K., Nishino, K., Takamatsu, J., Zheng, B., Ikeuchi, K.: Locally rigid globally non-rigid surface registration. In: Proc. IEEE International Conference on Computer Vision, Barcelona, Spain, pp. 1527–1534 (2011)
5. Taron, M., Paragios, N., Jolly, M.: Registration with uncertainties and statistical modeling of shapes with variable metric kernels. IEEE Trans. on Pattern Analysis and Machine Intelligence 31, 99–113 (2009)
6. Wang, F., Vemuri, B.C., Rangarajan, A., Eisenschek, S.: Simultaneous nonrigid registration of multiple point sets and atlas construction. IEEE Trans. on Pattern Analysis and Machine Intelligence 30, 2011–2022 (2008)

7. Wang, J., Chan, K.: Shape evolution for rigid and nonrigid shape registration and recovery. In: Proc. IEEE International Conference on Computer Vision and Pattern Recognition, Miami, USA, pp. 164–171 (2009)
8. Li, H., Shen, T., Huang, X.: Global optimization for alignment of generalized shapes. In: Proc. IEEE International Conference on Computer Vision and Pattern Recognition, Miami, USA, pp. 856–863 (2009)
9. Schmidt, F., Farin, D., Cremers, D.: Fast matching of planar shapes in sub-cubic runtime. In: Proc. IEEE International Conference on Computer Vision, Rio de Janeiro, Brazil, pp. 1–6 (2007)
10. Chen, Y., Medioni, G.: Object modelling by registration of multiple range images. Image Vision Computing 10, 145–155 (1992)
11. Chui, H., Rangarajan, A.: A new point matching algorithm for non-rigid registration. Computer Vision and Image Understanding 89, 114–141 (2003)
12. Sorkine, O.: Differential representations for mesh processing. Comput. Graph. Forum 25, 789–807 (2006)
13. Besl, P., McKay, N.: A method for registration of 3-d shapes. IEEE Trans. on Pattern Analysis and Machine Intelligence 14, 239–256 (1992)
14. Jian, B., Vemuri, B.: A robust algorithm for point set registration using mixture of Gaussians. In: Proc. IEEE International Conference on Computer Vision, Beijing, China, pp. 1246–1251 (2005)
15. Rusinkiewicz, S., Levoy, M.: Efficient variants of the ICP algorithm. In: Proc. IEEE International Conference on on 3-D Digital Imaging and Modeling, Quebec, Canada (2001)
16. Pottmann, H., Leopoldseder, S., Hofer, M.: Registration without ICP. Computer Vision and Image Understanding 95, 54–71 (2004)
17. Fitzgibbon, A.: Robust registration of 2d and 3d point sets. Image and Vision Computing 21, 1145–1153 (2001)
18. Aigner, M., Juttler, B.: Gauss-Newton-type technique for robustly fitting implicit defined curves and surfaces to unorganized data points. In: Proc. IEEE International Conference on Shape Modeling and Applications, New York, USA, pp. 121–130 (2008)
19. http://shapes.aimatshape.net/ (AIM@SHAPE, Digital Shape WorkBench)
20. Sharvit, D., Chan, J., Tek, H., Kimia, B.: Symmetry-based indexing of image databases. Journal of Visual Communication and Image Representation 9, 366–380 (1998)
21. http://graphics.stanford.edu/data/3Dscanrep/ (The Stanford 3D Scanning Repository)