Eurographics/ ACM SIGGRAPH Symposium on Computer Animation (2012)
P. Kry and J. Lee (Editors)

Mathematical Analysis on Affine Maps for 2D Shape Interpolation

S. Kaji[1], S. Hirose[2], S. Sakata[3], Y. Mizoguchi [4,6] and K. Anjyo[5,6]

[1]Yamaguchi University, [2]Kyoto University, [3]Tokyo Metropolitan University,
[4]Kyushu University, [5]OLM Digital, [6]JST CREST

Abstract

This paper gives a simple mathematical framework for 2D shape interpolation methods that preserve rigidity. An interpolation technique in this framework works for given the source and target 2D shapes, which are compatibly triangulated. Focusing on the local affine maps between the corresponding triangles, we describe a global transformation as a piecewise affine map. Several existing rigid shape interpolation techniques are discussed and mathematically analyzed through this framework. This gives us not only a useful comprehensive understanding of existing approaches, but also new algorithms and a few improvements of previous approaches.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Display algorithms

1. Introduction

Two-dimensional shape interpolation/deformation techniques have been widely used for many practical applications. For example, *Toon Boom* and *Adobe After Effects* are commercially available software tools, which provide functions based on them. A big trend of the techniques lies nowadays in a variety of the algorithms that preserve rigidity [ACOL00, XZWB05, SZGP05, BBA08]. Another stream provides the algorithms for computing high-quality shape-preserving deformation for 2D or 3D interactive applications [IMH05, BS08, JBPS11].

We consider 2D shape interpolation between the two input shapes: source and target. In general, when the two shapes are given without boundary matching nor compatible triangulation, we would need a preprocess to establish them. As for this issue, [BBA09] is a good reference describing the most relevant techniques along with their own approach. In this paper we assume that each shape is triangulated, and that one-to-one correspondence is established between the triangles of the source and target shapes.

There are many approaches, including those mentioned earlier, for 2D shape interpolation under the above assumptions. A typical scenario of these approaches came from the seminal work of [ACOL00]: We first define a homotopy of affine maps for each pair of the corresponding triangles of the source and target objects, such that it connects the identity map and the local affine map that gives a bijection between the corresponding triangles. Let us call this homotopy *local* in this paper. Next we construct the homotopy that gives global interpolation between the source and target. This homotopy is defined as a family of the piecewise affine maps, each of which is derived from the affine maps of the local homotopy through a certain energy minimization process. This scenario works well and has inspired many research works. However, from a practicality viewpoint, there remain many things to be improved and polished. For example, the following practical aspects of the methods should be addressed: (a) controllability - how to add constraints to get a better result?; (b) rotation consistency -how to treat large rotations (> 180 degrees)?; and (c) symmetry - Can we make it possible that the vertex paths for interpolation from shape A to shape B are the same as from B to A? Recently [BBA08] gave a formulation of rigid shape interpolation using normal equations, presenting the algorithms that meet these requirements.

This paper presents a new mathematical framework for the above homotopic approaches using affine maps. Unlike [BBA08], we start with analyzing the local affine map directly, and introduce a new local homotopy between the affine maps. We also present the algorithms to achieve global interpolation, each of which minimizes an energy func-

DOI: 10.2312/SCA/SCA12/071-076

tion with user-specified constraints. It is also discussed how the algorithms meet the above practical requirements. We demonstrate that our mathematical framework gives a comprehensive understanding of rigid interpolation/deformation approaches. In particular we illustrate the power of this framework with the animation examples obtained by several different constraint functions.

2. Formulation overview

As mentioned above, we consider the source and target 2D shapes that are to be interpolated, assuming that they are compatibly triangulated. We denote the source shape made of triangles by $P = (p_1, \dots, p_n), (p_i \in \mathbb{R}^2)$, where each p_i is a triangle vertex. Similarly we denote the target shape by $Q = (q_1, \dots, q_n), (q_i \in \mathbb{R}^2)$, which are the triangle vertices. The triangles are denoted by $\tau_1, \dots, \tau_m$, where $\tau_i = \{i_1, i_2, i_3\}$ is the set of the indices of the three vertices. Hence, the i-th source (respectively, target) triangle consists of p_{i_1}, p_{i_2}, and p_{i_3} (respectively, q_{i_1}, q_{i_2}, and q_{i_3}) for $i_1, i_2, i_3 \in \tau_i$.

Through sections 3 and 4, our local and global interpolation techniques are developed as follows:

- For each pair of the source and the target triangles corresponding to τ_i, we initially get the affine map, denoted by $\hat{A}_i$, that maps the initial triangle to the target triangle, where $\hat{A}_i$ is a 3×3-matrix. We then construct a homotopy between the 2×2 identity matrix and the linear part A_i of $\hat{A}_i$ (i.e., $A_i \in \mathrm{GL}^+(2)$ and see its precise definition in the next section). The homotopy is parameterized by t, with $t \in \mathbb{R}$.
- The collection of affine maps $\hat{A}_i$'s can be considered as a *piecewise affine transformation* from P to Q (see its precise definition in §4). We next construct a global homotopy between the inclusion map $P \hookrightarrow \mathbb{R}^2$ and the piecewise affine transformation from P to Q, which will be denoted by $\{\hat{B}_i(t) \in \mathrm{Aff}(2) \mid i \in \{1, 2, \dots, m\}\}$ with $t \in \mathbb{R}$ in section 4. It is obtained by minimizing a global error function regarding the linear part B_i (of $\hat{B}_i$) and A_i along with the user-specified constraint function.

3. Local theory

As described above, we begin with interpolating the local affine transformation. We note that interpolating affine transformation itself may have other interesting applications; see, for example, [Ale02] and [SD92].

First of all, recall that there is a unique affine transformation that maps a given triangle to another one. Specifically, suppose that we are given three points ${}^T(x_1, y_1), {}^T(x_2, y_2)$, and ${}^T(x_3, y_3) \in \mathbb{R}^2$ and want to map them onto ${}^T(x'_1, y'_1), {}^T(x'_2, y'_2)$, and ${}^T(x'_3, y'_3) \in \mathbb{R}^2$ in this order. Then the following 3×3-matrix

$$\hat{A} = \begin{pmatrix} x'_1 & x'_2 & x'_3 \\ y'_1 & y'_2 & y'_3 \\ 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ 1 & 1 & 1 \end{pmatrix}^{-1} \tag{1}$$

is of the form $\begin{pmatrix} a_{1,1} & a_{2,1} & d_x \\ a_{1,2} & a_{2,2} & d_y \\ 0 & 0 & 1 \end{pmatrix}$, and it represents the requested affine transformation. We denote the group of the two-dimensional affine transformations by $\mathrm{Aff}(2)$, which are represented by 3×3-matrices of the above form. We call $A = \begin{pmatrix} a_{1,1} & a_{2,1} \\ a_{1,2} & a_{2,2} \end{pmatrix}$ as the *linear part* and $d_{\hat{A}} = {}^T(d_x, d_y)$ as the *translation part* of $\hat{A}$ and consider them separately for interpolation. Interpolating the translation part can be neglected (see the discussion in §4). We focus on interpolation of linear transformation here. In general we may assume that transformation is orientation preserving, that is, it does not "flip" 2D shapes. We denote the group of the orientation preserving linear transformations by $\mathrm{GL}^+(2)$, which are represented by matrices with positive determinants.

A *homotopy of a linear transformation* $A \in \mathrm{GL}^+(2)$ is a series of matrices $A(t)$ parametrized by time $t \in \mathbb{R}$ such that $A(0) = I$ and $A(1) = A$, where I is the identity matrix. For example, the *linear homotopy* is given by

$$A^L(t) := (1-t)I + tA,$$

where ":=" means the left hand side is defined by the right hand side. This homotopy is not good; when, for example, $A = -I$ then $A(0.5) = 0$, which results in a collapse. We therefore need to avoid those kind of undesirable behaviors and to get better results.

In [ACOL00], the following method is proposed: By the polar decomposition [SD92], we can write $A = R_\theta S$, where R_θ is the rotation matrix of angular $\theta \in (-\pi, \pi]$ and S a positive definite symmetric matrix. (Note that θ has choice up to modulo 2π, which may cause a problem. We will discuss this issue in §4.) The interpolated transformation at t is defined to be

$$A^P(t) := R_{t\theta}((1-t)I + tS). \tag{2}$$

It is claimed in [ACOL00] that this method produces visually the best result by comparing several methods.

Here we propose a new homotopy and compare it with others based on mathematical analysis.

Non-degeneracy is a desirable property for "good" interpolation. In this sense the interpolated transformation should always give an affine homeomorphism at any t. More precisely, the determinant should always be positive. Unfortunately, none of previous methods satisfy it. In fact, we observe that the determinant of (2) can vanish;

$$\begin{aligned} \det A^P(t) &= \det(R_{t\theta}((1-t)I + tS)) \\ &= \det R_{t\theta} \det((1-t)I + tS)) \\ &= ((\lambda_1 - 1)t + 1)((\lambda_2 - 1)t + 1), \end{aligned}$$

where λ_1 and λ_2 are the singular values of A. Hence, $\det A^P(t)$ becomes negative for some t when $\lambda_1 \neq 1$ or $\lambda_2 \neq 1$.

We propose an alternative method which *does* satisfy the desirable property: As mentioned above, let $A = R_\theta S \in \mathrm{GL}^+(2)$. The interpolated transformation $A^E(t) \in \mathrm{GL}^+(2)$ for $t \in \mathbb{R}$ is then defined as

$$A^E(t) := R_\theta^t S^t. \tag{3}$$

Note that R_θ^t is nothing but $R_{t\theta}$ and $S^t = \exp(t \log S)$ is well-defined since S is a positive definite symmetric matrix (see Appendix). This method, which uses both the polar decomposition and the exponential map, can be seen as a combination of the ideas in [Ale02] and [SD92].

We observe that $\det A^E(t)$ is actually positive. Recall that for any matrix B, $\det(\exp(B)) = \exp(\mathrm{tr}(B))$ and, in particular, $\det(\exp(B)) > 0$. Then for any $t \in \mathbb{R}$,

$$\det A^E(t) = \det(R_\theta^t S^t) = \det R_\theta^t \det S^t = \exp(\mathrm{tr}(t \log S)) > 0.$$

Furthermore, one can see that $\det A^E(t)$ is monotonic with respect to t; we compute

$$\frac{d}{dt} \det A^E(t) \;=\; \frac{d}{dt}(\lambda_1^t \lambda_2^t) = \log(\lambda_1\lambda_2) \exp(t \log(\lambda_1\lambda_2))$$

where λ_1 and λ_2 are the singular values of A. Since $\exp(t \log(\lambda_1\lambda_2)) > 0$, we deduce that $\frac{d}{dt} \det A^E(t)$ either is constantly 0 when $\lambda_1\lambda_2 = 1$ or otherwise never vanishes. The monotonicity of $\det A^E(t)$ implies that of the area change during interpolation. This is also a nice property of our method. Figure 1 and the animation example in the supplemental video illustrate these advantages.

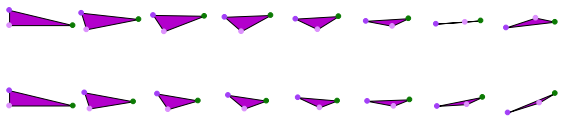

Figure 1: *A triangle is extrapolated from $t = 0$ through $t = 1.35$. The top row is obtained by (2), while the bottom is by (3). In the top, the triangle collapses around $t = 1.1$ and then turns over. In the bottom, the triangle never collapses. Please enlarge to see details.*

Several other methods are discussed in [ACOL00] such as those based on SVD. One can show that none of them satisfy the non-degeneracy by a similar argument. We therefore adopt $A^E(t)$ in (3) for the local homotopy in our method.

Remark

In [ACOL00], the following properties for local homotopy were claimed to be desirable:

(a) The homotopy should be symmetric with respect to t.
(b) The rotational angle and scale should change linearly.
(c) The triangle should keep its orientation, i.e. should not be reflected.
(d) The resulting vertices' paths should be simple.

As we mention in §4, (a) can be realized at global interpolation level for whatever local homotopy we use. Through the above discussion, we know that the method in [ACOL00] does not satisfy (c). As for (b), the scale change is not even monotonic. On the other hand, our method satisfies (c) and can easily achieve (b), by adjusting the speed of time parameter t. As for (d), we would need to estimate how "simple" a path is. It is possible to define a mathematical evaluation indicators with an analogy to topological invariants of curves such as winding numbers. We also find that our method is satisfactory in terms of those indicators. We will address these issues in a separate paper.

4. Global interpolation

To achieve global interpolation between the two shapes, we have to assemble local translations considered in the previous section. In our context, this means that we represent a global transformation as a piecewise affine transformation. More precisely, we construct a collection of affine maps

$$\mathbf{B}(t) := \{\hat{B}_i(t) \in \mathrm{Aff}(2) \mid i \in \{1, 2, \ldots, m\}\}, \ (t \in \mathbb{R})$$

such that

- $\hat{B}_i(t)$'s are consistent on the edges. More precisely, $\hat{B}_i(t)p_k = \hat{B}_j(t)p_k$ $(\forall t \in \mathbb{R})$ whenever $k \in \tau_i \cap \tau_j$.
- $\mathbf{B}(t)$ interpolates P and Q, i.e., $\mathbf{B}(0)p_k = p_k$ and $\mathbf{B}(1)p_k = q_k$, where $\mathbf{B}(t)p_k = \hat{B}_i(t)p_k$ for $k \in \tau_i$.
- $B_i(t)$ is "close" to $A_i(t)$, where $B_i(t)$ is the linear part of $\hat{B}_i(t)$ and $A_i(t)$ is the local homotopy obtained in the previous section.
- Each $B_i(t)$ varies continuously with respect to t.

The following observation is vital in this section. Let $v_i(t) := \mathbf{B}(t)p_i, (1 \le i \le n)$ be the image of the initial vertices. The piecewise affine transformation $\mathbf{B}(t)$ which maps p_i's to $v_i(t)$'s is uniquely determined by (1) and its entries are linear with respect to $v_i(t)$'s. Therefore, giving $\mathbf{B}(t)$ and giving $v_i(t)$'s are equivalent and we identify them and interchange freely in the following argument.

We will give a mathematical framework to obtain global interpolation from given local homotopies. For a moment we consider a fixed t. We then need two more ingredients other than local homotopy data; The first one is a set of *local error functions*

$$E_i : \mathrm{GL}^+(2) \times \mathrm{GL}^+(2) \to \mathbb{R}_{\ge 0}, \ (1 \le i \le m)$$

which is positive definite and quadratic with respect to the entries of the second $\mathrm{GL}^+(2)$. Intuitively, it measures how different the given two local transformations are. The second one is a *constraint function*

$$C : (\mathbb{R}^2)^n \to \mathbb{R}_{\ge 0}$$

which is also positive definite and quadratic. It controls the global translation. Furthermore, with this function, we can

incorporate various constraints on the vertex path as we will describe later.

If we are given local error functions for each triangle $\tau_i, (1 \le i \le m)$ and a constraint function, we combine them into a single *global error function*

$$\mathbf{E}_t(\mathbf{B}) := \sum_{i=1}^{m} E_i(A_i(t), B_i(t)) + C(v_1(t), \ldots, v_n(t)),$$

where we regard $\mathbf{B}(t)$ (or more precisely, the entries of $B_i(t)$ which are linear combinations of $v_i(t)$'s) as indeterminants to be solved. For each t, the *minimizer* of $\mathbf{E}_t$ is the set

$$\{\mathbf{B}(t) \mid \mathbf{E}_t(\mathbf{B}) \text{ attains the minimum at } \mathbf{B}(t)\},$$

which is equal to

$$\left\{ \mathbf{B}(t) \,\middle|\, \frac{\partial}{\partial v_i(t)} \mathbf{E}_t(\mathbf{B}) = 0,\ (1 \le \forall i \le n) \right\}$$

since $\mathbf{E}_t$ is positive definite quadratic form with respect to $v_i(t)$'s. The minimizer may have positive dimension in general, however, one can modify the constraint function C such that it becomes a single point, as we see by concrete examples later. The single minimizer $\mathbf{B}(t)$ is the piecewise affine map that we take as a global interpolation method.

For example, the error function used in [ACOL00] is obtained by setting

$$E_i^F(A_i(t), B_i(t)) := ||A_i(t) - B_i(t)||_F^2,$$

where the Frobenius norm of a matrix $M = (m_{ij})$ is $\|M\|_F^2 = \sum_{i,j} m_{ij}^2$. It measures how the local transformation and the final global transformation differ as linear maps. The resulting global error function is invariant under translation and hence requires two dimensional constraints to get a unique minimizer. For example, [ACOL00] proposes the following constraint function:

$$C(v_1(t), \ldots, v_n(t)) = ||(1-t)p_1 + tq_1 - v_1(t)||^2.$$

It produces a fairly satisfactory global transformation when the constraint function is very simple and rotation is "homogeneous." However, this method fails if 1) we want to put some constraints (see Figure 2), or 2) the expected rotation angles vary beyond 2π from triangles to triangles (see Figure 3):

To achieve more flexibility of shape deformation and easier manipulation by a user, [II09] and [IMH05] considered error functions which are invariant under similarity transformation, i.e., rotation and scale. An error function which is slightly different from them is proposed in [WW95] as

$$E_i^S(A_i(t), B_i(t)) := \min_{s,\delta \in \mathbb{R}} \sum_{k \in \tau_i} \|sR_\delta A_i(t)p_k - B_i(t)p_k\|^2,$$

which measures how different the two sets of points $\{A_i(t)p_k\}$ and $\{B_i(t)p_k\}$ are up to similarity transformation. In [II09] and [IMH05] they used a constraint function which

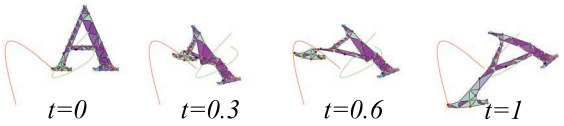

Figure 2: *An example of global interpolation obtained by E_i^F with the constraints on the vertices loci indicated by the curves. In the intermediate frames around $t = 0.3$ and $t = 0.6$, extreme shrink and flip of triangles are observed.*

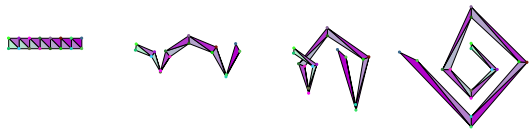

Figure 3: *An example of global interpolation obtained by E_i^F. To obtain smooth interpolation between the leftmost and rightmost figures, local transformations should deal with rotation angles larger than π, but E_i^F fails to make it.*

forces the vertex loci to be on the specified curves. We will see the detailed construction later.

For the purpose of finding a best matching global transformation with given local transformations, it is better to use a metric in the space of transformations, rather than in the space of points. Thus, we propose the following local error function, which is a slight modification of E_i^S:

$$E_i^R(A_i(t), B_i(t)) := \min_{s,\delta,\in \mathbb{R}} \|sR_\delta A_i(t) - B_i(t)\|_F^2, \qquad (4)$$

which measures how different $A_i(t)$ and $B_i(t)$ are as linear maps up to rotation and scale. We prove in Appendix that the above function is a positive definite quadratic form. Since it is invariant under similarity transformation, it avoids the flaws of E_i^F in the cases of 1) and 2); Compare Figure 4 with Figure 2, and Figure 5 with Figure 3, respectively. Also see the supplemental video.

Now we give a concise list of the constraints we can incorporate into a constraint function $C(v_1(t), \ldots, v_n(t))$:

- Some points must trace specified loci (for example, given by B-spline curves). This is realized as follows: let $u_i(t)$ be a user specified locus of p_i with $u_i(0) = p_i$ and $u_i(1) = q_i$. Then add the term $c_i||u_i(t) - v_i(t)||^2$, where $c_i > 0$ is a weight.
- The directions of some edges must be fixed. This is realized by adding the term $c_{ij}||v_i(t) - v_j(t) - e_{ij}(t)||^2$, where $e_{ij}(t) \in \mathbb{R}^2$ is a user specified vector and $c_{ij} > 0$ a weight. This gives a simple way to control the global rotation.
- The barycenter must trace a specified locus $u_o(t)$. This is realized by adding the term $c_o||\sum_{i=1}^n v_i(t)/n - u_o(t)||^2$,

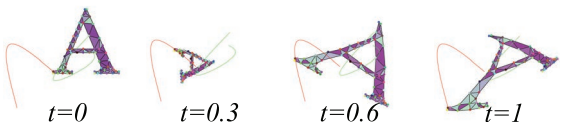

Figure 4: *An example of global interpolation obtained by E_i^R with the same input data as Figure 2. By allowing rotational and scale variance without any penalty in the error function, we can get more flexible control of the output animation.*

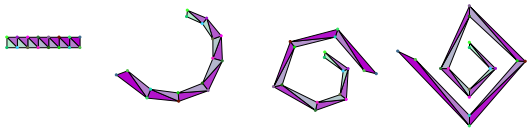

Figure 5: *An example of global interpolation obtained by E_i^R with the same input data as Figure 3. The proper rotation angles for the local triangles are automatically chosen by minimizing the global error function.*

where $c_o > 0$ is a weight. This gives a simple way to control the global translation.

Likewise we can add as many constraints as we want.

Improvements

Here we give three modifications to get better outcome and to allow further user control.

In assembling local error functions, we can take weighted sum instead of ordinary sum. We can put large weights to more important parts (triangles). For example, the more the area of triangle is, more important its rigidity becomes. Hence, it is reasonable to weight by the areas of the initial triangles:

$$E_i \leftarrow \mathrm{Area}(\Delta(p_{i_1}, p_{i_2}, p_{i_3}))E_i \quad (i_1, i_2, i_3 \in \tau_i).$$

This was already discussed in [XZWB05] and [BBA08] as well. Figures 6, 7 and the animation examples associated illustrate this effect. Of course it's also possible to manually specify which parts are important, if necessary.

We proposed to use the local error function E_i^R in (4) for a general use. However, we may not want some parts of the 2D shape to rotate or to scale (such as a face of a character). In such cases, we can use a balanced local error function

$$w_i E_i^F(t) + (1 - w_i)E_i^R(t),$$

where $w_i \in [0,1]$. If we put a large w_i, the rotation and scale of the triangle τ_i would be suppressed. We thus believe that our framework provides more user controllability over previous approaches.

In addition, as is shown in [BBA08], we can symmetrize

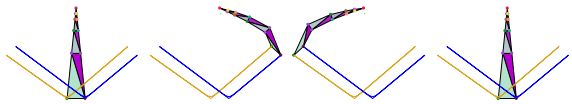

Figure 6: *An example of global interpolation obtained by E_i^R without weight. Since all parts are treated equally, the larger triangles yield to strain as much as the smaller ones*

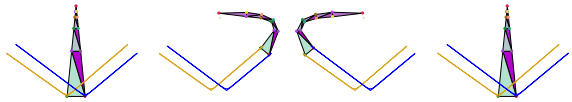

Figure 7: *An example of global interpolation obtained by E_i^R weighted with the areas of the initial triangles. Since the rigidity of larger parts is more counted, it produces natural interpolation.*

the global interpolation by symmetrizing the global error function. Let $\mathbf{E}(t)$ be any global error function for any local homotopies $A_i^{-1}(t)$. Then define a new error function by

$$\mathbf{E}'(t) := \mathbf{E}(t) + \mathbf{E}^{-1}(1-t).$$

This is symmetric in the sense that it is invariant under the substitution $A_i \leftarrow A_i^{-1}$ and $t \leftarrow 1-t$. That means, it gives the same minimizing solution if we swap the initial and the terminal polygons and reversing time.

5. Discussions and future work

This paper describes a simple framework for rigid shape interpolation between 2D shapes, where global interpolation is defined with the homotopy as a family of the piecewise affine maps. We have shown that most of the rigid interpolation methods are characterized in this framework, while providing several new algorithms to improve quality and usability of the existing methods.

In our framework, users can interact and control an outcome by, for example, specifying loci of vertices. This is achieved by putting the corresponding constraints to the global error function. The constraints are, however, handled only in the global interpolation stage without considering local rigidity. It is preferable that these constraints can also be specified in the local interpolation stage.

Our interpolation method for the local transformations assures that it does not flip images. Our global interpolation method, however, may not carry this property. We thus need to clarify when such an undesirable situation occurs, and then should develop a new technique to avoid it.

There are a number of future avenues for this work. For example, our method can be easily and directly extended into

3D cases. We can also develop finer mathematical evaluators than determinant which capture various aspects of an interpolation method and enable us to compare different methods. We will discuss these in a sequel to this paper.

6. Appendix

Well-definedness of the exponential: We prove that the exponential in (3) is uniquely determined and can be computed using SVD. Recall the *singular value decomposition* [GvVL96] gives a factorization $A = R_\alpha D R_\beta$, where R_α and R_β are rotation matrices and D is a diagonal matrix. The entries of D is called the *singular values* of A. Then, the polar decomposition is obtained as

$$R_\theta = R_{\alpha+\beta}, \quad S = R_{-\beta} D R_\beta.$$

Since $\exp(U^{-1}BU) = U^{-1}\exp(B)U$ for any $U, B \in \mathrm{GL}(2)$, we have $S^t = R_{-\beta} D^t R_\beta$.

Closed form for the similarity invariant error function: We give a closed form for the similarity invariant metric (4).

$$\min_{s,\delta\in\mathbb{R}} \|sR_\delta A - B\|_F^2 = \|B\|_F^2 - \frac{\left\|B\cdot {}^T A\right\|_F^2 + 2\det\left(B\cdot {}^T A\right)}{\|A\|_F^2}.$$

This is positive definite quadratic with respect to the entries of B. We prove the above equality: Put $J = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$ and we have

$$\min_{s,\delta\in\mathbb{R}} \|sR_\delta A - B\|_F^2 = \min_{x,y\in\mathbb{R}} \|(xI + yJ)A - B\|_F^2.$$

Define $\langle A, B\rangle := \mathrm{tr}(A\cdot {}^T B)$, then we have $\langle JA, A\rangle = 0$ and

$$\begin{aligned}
&\|(xI+yJ)A - B\|_F^2 = \langle xA + yJA - B, xA + yJA - B\rangle \\
=&x^2\|A\|_F^2 - 2x\langle A,B\rangle + y^2\|A\|_F^2 - 2y\langle JA,B\rangle + \|B\|_F^2 \\
=&\|A\|_F^2\left(x - \frac{\langle A,B\rangle}{\|A\|_F^2}\right)^2 + \|A\|_F^2\left(y - \frac{\langle JA,B\rangle}{\|A\|_F^2}\right)^2 \\
&+ \|B\|_F^2 - \frac{\langle A,B\rangle^2 + \langle JA,B\rangle^2}{\|A\|_F^2} \\
\geq&\|B\|_F^2 - \frac{\left\|B\cdot {}^T A\right\|_F^2 + 2\det\left(B\cdot {}^T A\right)}{\|A\|_F^2}.
\end{aligned}$$

This proof is due to [Och12].

Efficiency of finding the minimizer: We show that finding the minimizer of a global error function is efficient enough. Since the global error function is positive definite quadratic form, it can be written as a function of $\mathbf{v}(t) = {}^T(v_1(t)_x, v_1(t)_y, \ldots, v_n(t)_x, v_n(t)_y) \in \mathbb{R}^{2n}$ as

$$E(\mathbf{v}(t)) = {}^T\mathbf{v}(t) G \mathbf{v}(t) + {}^T\mathbf{v}(t)\mathbf{u}(t) + c,$$

for some $(2n\times 2n)$-symmetric matrix G, $\mathbf{u}(t) \in \mathbb{R}^{2n}$, and $c \in \mathbb{R}$. Since $\frac{\partial}{\partial \mathbf{v}(t)}({}^T\mathbf{v}(t)G\mathbf{v}(t) + {}^T\mathbf{v}(t)\mathbf{u}(t) + c) = 2G\mathbf{v}(t) + \mathbf{u}(t)$, we have that $\mathbf{v}(t) = -\frac{1}{2}G^{-1}\mathbf{u}(t)$ is the minimizer. Note that G is time-independent and we need to compute G^{-1} just once for all frames as in [ACOL00].

7. Acknowledgments

This work stems from the Institute of Math-for-Industry joint research project which was held at the Institute of Mathematics for Industry, Kyushu University, during 5-9 March in 2012. S. Hirose and S. Sakata were supported by GCOE "Fostering top leaders in mathematics" in Kyoto University and Suudenki GP of Tokyo Metropolitan University, respectively.

References

[ACOL00] ALEXA M., COHEN-OR D., LEVIN D.: As-rigid-as-possible shape interpolation. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques* (2000), SIGGRAPH '00, ACM, pp. 157–164. 1, 2, 3, 4, 6

[Ale02] ALEXA M.: Linear combination of transformations. *ACM Trans. Graph. 21*, 3 (July 2002), 380–387. 2, 3

[BBA08] BAXTER W., BARLA P., ANJYO K.: Rigid shape interpolation using normal equations. In *Proceedings of the 6th international symposium on Non-photorealistic animation and rendering* (2008), NPAR '08, ACM, pp. 59–64. 1, 5

[BBA09] BAXTER W., BARLA P., ANJYO K.: Compatible embedding for 2d shape animation. *IEEE Transactions on Computer Graphics and Visualization 15*, 5 (2009), 867–879. 1

[BS08] BOTSCH M., SORKINE O.: On linear variational surface deformation methods. *IEEE Transactions on Computer Graphics and Visualization 14*, 1 (2008), 213–230. 1

[GvVL96] GOLUB G. H., VAN VAN LOAN C. F.: *Matrix Computations (Johns Hopkins Studies in Mathematical Sciences)*, 3rd ed. The Johns Hopkins University Press, Oct. 1996. 6

[II09] IGARASHI T., IGARASHI Y.: Implementing as-rigid-as-possible shape manipulation and surface flattening. *J. Graphics, GPU, & Game Tools 14*, 1 (2009), 17–30. 4

[IMH05] IGARASHI T., MOSCOVICH T., HUGHES J. F.: As-rigid-as-possible shape manipulation. *ACM Trans. Graph. 24*, 3 (2005), 1134–1141. 1, 4

[JBPS11] JACOBSON A., BARAN I., POPOVIĆ J., SORKINE O.: Bounded biharmonic weights for real-time deformation. *ACM Trans. Graph 30*, 4 (2011), Article 78. 1

[Och12] OCHIAI H.: personal communication (2012). 6

[SD92] SHOEMAKE K., DUFF T.: Matrix animation and polar decomposition. In *Proceedings of the conference on Graphics interface '92* (1992), Morgan Kaufmann Publishers Inc, pp. 258–264. 2, 3

[SZGP05] SUMNER R. W., ZWICKER M., GOTSMAN C., POPOVIĆ J.: Mesh-based inverse kinematics. *ACM Trans. Graph 24*, 3 (2005), 488–495. 1

[WW95] WERMAN M., WEINSHALL D.: Similarity and affine invariant distances between 2d point sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence 17* (1995), 810–814. 4

[XZWB05] XU D., ZHANG H., WANG Q., BAO H.: Poisson shape interpolation. In *Proceedings of the 2005 ACM symposium on Solid and physical modeling* (2005), SPM '05, ACM, pp. 267–274. 1, 5