Landmark-Guided Elastic Shape Analysis of Human Character Motions

Martin Bauer[*], Markus Eslitzbichler[†], Markus Grasmair[‡]

November 22, 2016

Abstract. Motions of virtual characters in movies or video games are typically generated by recording actors using motion capturing methods. Animations generated this way often need postprocessing, such as improving the periodicity of cyclic animations or generating entirely new motions by interpolation of existing ones. Furthermore, search and classification of recorded motions becomes more and more important as the amount of recorded motion data grows.

In this paper, we will apply methods from shape analysis to the processing of animations. More precisely, we will use the by now classical elastic metric model used in shape matching, and extend it by incorporating additional inexact feature point information, which leads to an improved temporal alignment of different animations.

Keywords. Curve matching, feature point information, skeletal animation, shape analysis, square root velocity transform.

MSC Subject Classification. 65D18; 58D10; 49Q10.

1 Introduction

Animations. Virtual characters in movie and TV special effects or video games are most commonly animated using *skeletal animation*, where a character's motion is described in terms of joint-angles in an approximation of a human skeleton. *Motion capturing* is a typical way to generate such animations, whereby an actor or stuntman

[*] Faculty of Mathematics, Florida State University
E-mail: bauer@math.fsu.edu
MB was supported by FWF project P24625.

[†] Department of Mathematical Sciences, NTNU Trondheim.
E-mail: m.eslitzbichler@gmail.com

[‡] Department of Mathematical Sciences, NTNU Trondheim.
E-mail: markus.grasmair@math.ntnu.no

performs the requested motion while being recorded from multiple angles. From this, the underlying skeletal pose can then be reconstructed. While motion capturing can produce very life-like animations, it has a number of drawbacks, some of which can be addressed algorithmically [24, 10].

In [14], methods from shape analysis were applied to a number of problems in computer animation: periodicity of animations, interpolation between animations and motion recognition. A central component in this is the modelling of curve reparametrizations as elements of the diffeomorphism group on the circle or the unit interval, $\mathrm{Diff}(S^1)$ and $\mathrm{Diff}([0,1])$ respectively. For animations, such reparametrizations can be used to align them in time, similar to the concept of *timewarp curves* in [16, 17]. Note that the skeletal animation approach is very different in concept and applications from silhouette-based representations of motions, as for example in [1].

In this paper we will demonstrate how feature point information can be incorporated into shape matching techniques, and how this can be applied in the context of computer animations in order to improve the temporal alignment of related actions.

Shape matching. The field of shape analysis concerns itself with the study and classification of similarities and dissimilarities within certain classes of shapes. In order to achieve this objective, a variety of different methods has been developed, each tailored to the actual class of shapes under consideration. A particularly important example for a shape space, which is also the main focus of this article, is the space of unparametrized curves,[1] and, in recent years, Riemannian methods for shape spaces of curves have been deeply explored both theoretically [33, 22, 28, 21, 15, 26, 20, 27, 11, 9] as well as from an application oriented point of view [23, 25, 19, 30]. See [7] for an overview of these topics.

Although the main contribution of this article is not limited to a specific Riemannian metric on the shape space of unparametrized curves, we will focus for simplicity on a particular one that is related to the so called *Square Root Velocity* (SRV) framework [29, 6]. This metric is given as the pullback of the L^2–metric via a transformation called the SRVT. This allows for the development of extremely efficient numerical methods and, as a consequence, it has been successfully used in a series of applications [32, 14]. See also [34, 8] for other reparametrization invariant metrics that can be represented in a similar way.

Our main addition to this standard setting will be the incorporation of point correspondences between two curves one wants to match. In the particular application of animation processing, these correspondences will describe similar poses at different points in time in the two animations. We assume here that the point correspondences are manually entered pieces of information that are possibly error-prone. Thus we are interested in an exact matching of the unparametrized curves, but only an inexact matching of the point correspondences. Our approach differs from the one presented in [31], where feature information is included in the form of auxiliary functions that are combined with the geometric curve to form higher dimensional curves that can then be matched by the usual methods. Instead, we augment the matching energy functional

[1] Closed curves can be used, for example, to represent outlines of objects in recognition applications.

with an extra feature matching term, leaving the elastic matching terms unchanged. In comparison to [18], which focused on matching three dimensional surfaces, we allow inexact matching of landmarks in our model. We will discuss in this article both the theoretical framework of shape matching with feature points and computational aspects.

Overview In Sect. 2 we give an overview of the general framework for shape matching with additional feature point information and will then apply this framework to one particular choice of metric: the metric that is induced by the SRV transformation. We then present the basics of two algorithms that can be used to determine the optimal parametrization to match two curves: In Sect. 3.1 we derive all the formulas necessary to find the parametrization using a gradient descent approach. In addition, in Sect. 3.2, we discuss the application of a dynamic programming based algorithm. In Sect. 4 we apply the previously presented framework to process human motion data. Moreover, we show some examples of an application to two-dimensional curves, where the workings of the method can be better visualized.

2 The general framework

Problem formulation. We will start this article by formulating our main problem:

Problem 1. *Given two unparametrized curves $[c_0]$ and $[c_1]$ and a number of n point correspondences between the curves – i.e., points C_0^i and C_1^i that should be matched onto each other – we want to find an optimal deformation between the two curves, that also respects the alignment of the feature points.*

In order to achieve this goal, we will first choose representatives of the shapes $[c_0]$ and $[c_1]$, i.e., parametrized curves c_0 and c_1. Each representative c_j determines parameter values θ_j^i that correspond to the feature points C_j^i in the sense that

$$c_j(\theta_j^i) = C_j^i.$$

Thus we can represent any shape $[c_0]$ with additional feature point information as a tuple $(c_0, \boldsymbol{\theta}_0)$, with $c_0 \in \operatorname{Imm}(S^1, \mathbb{R}^d)$ and $\boldsymbol{\theta}_0 = (\theta_0^i)_i \in (S^1)^n$. One way to tackle Problem 1 is then to construct a similarity measure on the product space $\operatorname{Imm}(S^1, \mathbb{R}^d) \times (S^1)^n$ that has certain invariance properties with respect to the action of the diffeomorphism group.

To do this mathematically rigorously we will need some results and definitions from infinite dimensional Riemannian geometry first.

2.1 The manifold of parametrized curves

In this article we will consider the space of regular curves from a parameter space M into some – possibly high dimensional – $\mathbb{R}^d$:

$$\operatorname{Imm}(M, \mathbb{R}^d) := \left\{ c \in C^\infty(M, \mathbb{R}^d) : |c'| > 0 \right\}. \tag{1}$$

Here M stands for the circle S^1 in the case of closed curves and for the interval $[0,2\pi]$ for open curves.

The space $\operatorname{Imm}(M,\mathbb{R}^d)$ is a smooth infinite dimensional Fréchet manifold with tangent space $T_c\operatorname{Imm}(M,\mathbb{R}^d)$, the set of all vector fields along the curve c. Using the trivialization of $\mathbb{R}^d$ we can identify the tangent space with

$$T_c\operatorname{Imm}(M,\mathbb{R}^d) := \left\{h \in C^\infty(M,\mathbb{R}^d)\right\}. \tag{2}$$

On the manifold $\operatorname{Imm}(M,\mathbb{R}^d)$ we consider reparametrization invariant metrics G_c, i.e., Riemannian metrics that satisfy

$$G_c(h,k) = G_{c\circ\varphi}(h\circ\varphi, k\circ\varphi) \quad \text{for all} \quad \varphi \in \operatorname{Diff}(M). \tag{3}$$

Here, $\operatorname{Diff}(M)$ denotes the group of orientation preserving diffeomorphisms of M, which consists of all reparametrizations of the curves under consideration.

The most prominent example of such a metric is the elastic metric $G^{a,b}$ that is defined by

$$G_c^{a,b}(h,k) = \int_M a^2|(D_sh)^\perp||(D_sk)^\perp| + b^2|(D_sh)^\top||(D_sk)^\top|ds. \tag{4}$$

Here a^2, b^2 are positive constants, $D_s = \frac{1}{|c'|}\partial_\theta$ denotes the arc length derivative, $ds = |c'|d\theta$ is arc length integration, $v = D_sc$ is the unit length tangent vector, $|(D_sh)^\top| = \langle D_sh, v\rangle$ denotes the tangential component of D_sh and $|(D_sh)^\perp| = D_sh - |(D_sh)^\top|v$ the normal component. Note that constant vector fields h are in the kernel of $G^{a,b}$, thus (4) defines only a metric on the manifold of immersions modulo translations.

Other metrics that have been introduced include higher order Sobolev metrics, i.e., metrics of the form

$$G_c^l(h,k) = \int_M \left(\sum_j a_j\langle D_s^jh, D_s^jk\rangle\right) ds, \tag{5}$$

with coefficients a_j possibly depending on the foot point c. Depending on the order of the metric G, local and global well-posedness of the geodesic equation have been shown and the metric completions of the corresponding spaces have been studied [11]. Recently a numerical framework for these metrics has been developed [3, 4, 5] and in future work we plan to combine this algorithm with the framework introduced in this article.

In the experimental part of this article we will focus on one particular member of this family, namely the elastic metric that corresponds to the parameters $a = 1$, $b = \frac{1}{2}$. This metric has the advantage that it has a very helpful representation as a pullback of the flat L^2–metric. To see this we introduce the so-called *square root velocity transform* (or short *SRVT*):

$$R : \begin{cases} \operatorname{Imm}(M,\mathbb{R}^d) & \mapsto C^\infty(M,\mathbb{R}^d\setminus\{0\}), \\ c & \to \dfrac{c'}{\sqrt{|c'|}}. \end{cases} \tag{6}$$

The SRVT, when regarded for curves modulo translations, has an inverse, which is given by:

$$R^{-1}: \begin{cases} C^\infty(M, \mathbb{R}^d \setminus \{0\}) & \mapsto \operatorname{Imm}(M, \mathbb{R}^d), \\ q & \to \int_0^\tau |q| q \, d\theta. \end{cases} \tag{7}$$

On $C^\infty(M, \mathbb{R}^d)$ we can consider the flat L^2–metric. In [29] it has been shown, that the pullback via R of the L^2–metric is exactly the elastic metric (4). The situation for open curves is particularly easy, as one has then explicit formulas for geodesics and geodesic distance:

Theorem 1. *The image of the R-map of the manifold of open curves is an open subset of $C^\infty([0, 2\pi], \mathbb{R}^d)$:*

$$\operatorname{Im}(R) = \left\{ q \in C^\infty([0, 2\pi], \mathbb{R}^d) : |q| \neq 0 \right\}. \tag{8}$$

Two open curves c_0, c_1 can be connected by a geodesic if and only if there exist no $\theta \in [0, 2\pi]$ and no $\lambda > 0$ such that $c_0'(\theta) = -\lambda c_1'(\theta)$. In this case, the unique geodesic connecting them is given by

$$c(t, \theta) = R^{-1}\left((1-t)R(c_0) + tR(c_1)\right). \tag{9}$$

Moreover, the induced geodesic distance on $\operatorname{Imm}([0, 2\pi], \mathbb{R}^d)$ is given by

$$d(c_0, c_1) = \sqrt{\int_0^{2\pi} \|R(c_0) - R(c_1)\|^2_{\mathbb{R}^d} d\theta}. \tag{10}$$

For a proof of this theorem see [6]. The situation for closed curves is less explicit. We have the following characterization of the image of the SRVT, which will build the fundament of our algorithms:

Theorem 2. *The image $R(\operatorname{Imm}(S^1, \mathbb{R}^d))$ of the manifold of closed curves under the SRVT-transform is a codimension d submanifold of the flat space $C^\infty(S^1, \mathbb{R}^d)$. A basis of the orthogonal complement $\left(T_q R(\operatorname{Imm}(S^1, \mathbb{R}^d))\right)^\perp$ is given by the d vectors*

$$U_i(q) = \frac{1}{|q|}\left(q_i q_1, \ldots, q_i^2 + |q|^2, \ldots, q_i q_d\right). \tag{11}$$

Using this basis, efficient numerical methods for calculating geodesics between closed curves have been developed, see [29, 6].

2.2 The matching functional on the space of parametrized curves

To define our similarity measure on the product space $\operatorname{Imm}(M, \mathbb{R}^d) \times M^n$, we will first introduce an energy functional that is defined for arbitrary paths in $\operatorname{Imm}(M, \mathbb{R}^d)$. We will then define the similarity measure as the minimal energy over all paths with given boundary shapes $[c_0]$ and $[c_1]$. The important features of the similarity measure on the space of unparametrized curves will be:

(a) The similarity measure does not depend on the choice of representatives $(c_i, \boldsymbol{\theta}_i)$ of the observed shapes $([c_i], \mathbf{C}^i)$.

(b) The optimal deformation is guided by both the shape of the boundary curves and by the feature point information.

(c) The similarity measure forces an exact matching of the unparametrized curves, but only an inexact matching of the feature point information.

Given a curve $\hat{c} \in \operatorname{Imm}(M, \mathbb{R}^d)$, we denote in the following by $\hat{c}(\boldsymbol{\theta})$ the whole vector of points $\hat{c}(\theta^i) \in \mathbb{R}^d$, $1 \leq i \leq n$.

For parameter values $\boldsymbol{\theta}_0 = (\theta_0^i)_i \in M^n$ and feature points $\mathbf{C}_1 = (C_1^i)_i \in (\mathbb{R}^d)^n$ we define the energy functional for a given path $c\colon [0,1] \to \operatorname{Imm}(M, \mathbb{R}^d)$ as:

$$\mathcal{E}(\boldsymbol{\theta}_0, \mathbf{C}_1)(c) = \int_0^1 G_c(c_t, c_t)dt + \lambda \operatorname{FM}\left(c(1, \boldsymbol{\theta}_0), \mathbf{C}_1\right) . \tag{12}$$

Here $G_c(\cdot,\cdot)$ is any reparametrization invariant metric on $\operatorname{Imm}(M, \mathbb{R}^d)$ and FM denotes some similarity measure on $\mathbb{R}^{d \times n}$. The only conditions on FM we impose at the moment are that FM is lower semi-continuous and $\operatorname{FM}(\mathbf{C}, \mathbf{C}) = 0$. The first condition is necessary for the subsequent energy minimization, while the second condition implies that constant paths actually have zero energy. When we discuss later the actual computation of energy minimizing paths, we will introduce further conditions that make their numerical approximation possible.

Lemma 3. *The energy functional* (12) *satisfies the invariance property*

$$\mathcal{E}(\varphi^{-1}(\boldsymbol{\theta}_0), \mathbf{C}_1)(c \circ \varphi) = \mathcal{E}(\boldsymbol{\theta}_0, \mathbf{C}_1)(c). \tag{13}$$

Remark 4. The meaning of this invariance property will become clear in Sect. 2.3, where we will consider the action of the diffeomorphism group on the quotient space $\operatorname{Imm}(M, \mathbb{R}^d) \times M^n$. ■

Proof. Using the reparametrization invariance of the metric G_c we calculate

$$\begin{aligned}\mathcal{E}(\varphi^{-1}(\boldsymbol{\theta}_0), \mathbf{C}_1)(c \circ \varphi) &= \int_0^1 G_{c\circ\varphi}((c\circ\varphi)_t, (c\circ\varphi)_t)dt + \lambda \operatorname{FM}\left((c\circ\varphi)(1, \varphi^{-1}(\boldsymbol{\theta}_0)), \mathbf{C}_1\right) \\ &= \int_0^1 G_c(c_t, c_t)dt + \lambda \operatorname{FM}\left(c(1, \boldsymbol{\theta}_0), \mathbf{C}_1\right) = \mathcal{E}(\boldsymbol{\theta}_0, \mathbf{C}_1)(c).\end{aligned}$$ □

Using this energy functional we define our similarity measure on the product space $\operatorname{Imm}(M, \mathbb{R}^d) \times M^n$ of parametrized curves with feature points as

$$\boxed{d_P\left((c_0, \boldsymbol{\theta}_0), (c_1, \boldsymbol{\theta}_1)\right) := \inf_{c:[0,1]\to\operatorname{Imm}} \mathcal{E}(\boldsymbol{\theta}_0, c_1(\boldsymbol{\theta}_1))(c)} \tag{14}$$

where the infimum is taken over all paths c in $\operatorname{Imm}(M, \mathbb{R}^d)$ that satisfy

$$c(0, \cdot) = c_0 \text{ and } c(1, \cdot) = c_1 \, . \tag{15}$$

Remark 5. We do not call the similarity measure a distance, since it is not symmetric in general. However, it would be straightforward to construct a symmetric version of this. This will be described in Sect. 2.4. ■

Because we fix the endpoint $c(1,\cdot) = c_1$, we can write the similarity measure d_P as

$$d_P((c_0,\boldsymbol{\theta}_0),(c_1,\boldsymbol{\theta}_1)) = \inf_{c:[0,1]\to \mathrm{Imm}} \Big[\int_0^1 G_c(c_t,c_t)\,dt\Big] + \lambda\,\mathrm{FM}(c_1(\boldsymbol{\theta}_0), c_1(\boldsymbol{\theta}_1)).$$

That is, we only minimize the first term of the energy functional, and we do not allow to change the value of the second term at all. The meaning of the second term, will become clear when we consider it on the shape space of unparametrized curves.

2.3 The similarity measure on the shape space of unparametrized, feature curves.

In this section we want to use the previously defined similarity measure on parametrized curves to induce a similarity measure on the shape space of unparametrized curves with feature point information. Therefore we have to determine the induced action of the diffeomorphism group on the product space $\mathrm{Imm}(M,\mathbb{R}^d)\times M^n$. On the first factor $\mathrm{Imm}(M,\mathbb{R}^d)$ it is simply given by composition from the right. To compute the action on the second factor M^n we need to compute the effect of a reparametrization on the feature points. We have:

$$C_0^i = c_0(\theta_0^i) = c(\varphi(\varphi^{-1}(\theta_0^i))) = (c\circ\varphi)(\varphi^{-1}(\theta_0^i)). \tag{16}$$

Thus the induced action on the product space is given by

$$(c_0,\boldsymbol{\theta}_0)\circ\varphi = (c_0\circ\varphi, \varphi^{-1}(\boldsymbol{\theta}_0)). \tag{17}$$

Using the invariance of our similarity measure – cf. Lemma 3 – we obtain the following result:

Theorem 6. *The similarity measure* (14) *on* $\mathrm{Imm}(M,\mathbb{R}^d)\times M^n$ *induces a similarity measure on the shape space of unparametrized curves with additional feature point information. The induced functional is given by:*

$$d\left(([c_0],\mathbf{C}_0),([c_1],\mathbf{C}_1)\right) := \inf_{\varphi\in\mathrm{Diff}(M)} d_P\left((c_0,\boldsymbol{\theta}_0),(c_1\circ\varphi,\varphi^{-1}(\boldsymbol{\theta}_1)\right). \tag{18}$$

Here $(c_j,\boldsymbol{\theta}_j)$ *are arbitrary representatives of the shapes* $([c_j],\mathbf{C}_j)$*.*

Remark 7. Note, that the energy functional will force an exact matching of the unparametrized curves, but only an inexact matching of the feature points. The reason for this is that we assume the feature points to be an additional manually entered information that is possibly error-prone. ■

Proof. We need to show that d does not depend on the actual choice of representatives c_0 and c_1. Any other representatives of $[c_i]$ can be written as $c_i \circ \varphi$ for some diffeomorphism φ. Since we are minimizing over all possible reparametrizations of c_1, the functional clearly does not depend on the choice of the representative c_1. It remains to verify the independence of reparametrizations of c_0. Therefore we calculate

$$\begin{aligned}\inf_{\varphi \in \operatorname{Diff}(M)} & d_P \left((c_0 \circ \psi, \psi^{-1}(\boldsymbol{\theta}_0)), (c_1 \circ \varphi, \varphi^{-1}(\boldsymbol{\theta}_1)\right) \\ &= \inf_{\varphi \in \operatorname{Diff}(M)} \left(\inf_{c:[0,1]\to \operatorname{Imm}} \mathcal{E}\left(\psi^{-1}(\boldsymbol{\theta}_0), c_1(\boldsymbol{\theta}_1)\right)(c) \right),\end{aligned}$$

where the infimum is taken over all paths c that satisfy the boundary conditions

$$c(0,\cdot) = c_0 \circ \psi, \quad c(1,\cdot) = c_1 \circ \varphi\,.$$

Using the invariance property of the functional – cf. Lemma 3 – we can rewrite this as

$$\begin{aligned}\inf_{\varphi \in \operatorname{Diff}(M)} & d_P \left((c_0 \circ \psi, \psi^{-1}(\boldsymbol{\theta}_0)), (c_1 \circ \varphi, \varphi^{-1}(\boldsymbol{\theta}_1)\right) \\ &= \inf_{\varphi \in \operatorname{Diff}(M)} \left(\inf_{c:[0,1]\to \operatorname{Imm}} \mathcal{E}\left(\boldsymbol{\theta}_0, c_1(\boldsymbol{\theta}_1)\right)(c \circ \psi^{-1}) \right) \\ &= \inf_{\varphi \in \operatorname{Diff}(M)} \left(\inf_{\tilde{c}:[0,1]\to \operatorname{Imm}} \mathcal{E}\left(\boldsymbol{\theta}_0, c_1(\boldsymbol{\theta}_1)\right)(\tilde{c}) \right)\end{aligned}$$

such that

$$\tilde{c}(0,\cdot) = c_0 \circ \psi \circ \psi^{-1} = c_0, \quad c(1,\cdot) = c_1 \circ \varphi,$$

which concludes the proof. □

We note that the similarity measure d can also be written as

$$\begin{aligned}& d(([c_0], \mathbf{C}_0), ([c_1], \mathbf{C}_1)) \\ &\qquad = \inf_{\varphi \in \operatorname{Diff}(M)} \left(\inf_{c:[0,1]\to \operatorname{Imm}} \left[\int_0^1 G_c(c_t, c_t)\, dt \right] + \lambda \operatorname{FM}(c_1 \circ \varphi(\boldsymbol{\theta}_0), \mathbf{C}_1) \right),\end{aligned}$$

where the inner infimum is taken over all paths c that satisfy the conditions $c(0,\cdot) = c_0$ and $c(1,\cdot) = c_1 \circ \varphi$.

Remark 8. Due to the invariance with respect to the reparametrization group, all the metrics G_c descend to the shape space of unparametrized curves, i.e., they induce a metric on the quotient space $\mathcal{S} := \operatorname{Imm}(M, \mathbb{R}^d)/\operatorname{Diff}(M)$ such that the projection

$$\pi : \operatorname{Imm}(M, \mathbb{R}^d) \to \operatorname{Imm}(M, \mathbb{R}^d)/\operatorname{Diff}(M)$$

is a Riemannian submersion. For a detailed discussion of this topic we refer to the article [21]. For $\lambda = 0$ – i.e., no feature point matching – the similarity measure (18) is then given by the induced geodesic distance on the quotient space. If we assume existence of a minimizer, it would be given by a horizontal geodesic on the top space, the manifold of parametrized curves. For $\lambda > 0$ minimizers of (18) will still be geodesics on $\operatorname{Imm}(M, \mathbb{R}^d)$, however their initial velocity will in general not be horizontal anymore. The induced curve on the quotient space $\operatorname{Imm}(M, \mathbb{R}^d)/\operatorname{Diff}(M)$ will thus not be a geodesic. In [2] such curves have been called ballistic curves. ■

Remark 9. In certain applications, there exist natural *reference parametrizations* $c_{i,\mathrm{ref}} \in [c_i]$ of the curves one is interested in. For instance, if one deals with skeletal animations (for details see Section 4 below), the curves are mappings from a time interval into the so-called joint space. In this case, the reference parametrization of a given animation uses a uniform frame rate, and reparametrizations correspond to local speed-ups or slow-downs. In such a setting, it makes sense to define the feature matching term based on the similarity of parameter values rather than the points on the curve. Given some distance measure $\widehat{\mathrm{FM}}$ on M^n, this can be achieved in our setting by defining

$$\mathrm{FM}(\mathbf{C}_1, \mathbf{C}_2) := \begin{cases} \widehat{\mathrm{FM}}(c_{0,\mathrm{ref}}^{-1}(\mathbf{C}_0), c_{1,\mathrm{ref}}^{-1}(\mathbf{C}_1)) & \text{if } \mathbf{C}_i \in c_{i,\mathrm{ref}}(M)^n, \\ +\infty & \text{else.} \end{cases}$$

Then the similarity measure d can be written as

$$\begin{aligned} &d(([c_0], \mathbf{C}_0), ([c_1], \mathbf{C}_1)) \\ &\qquad = \inf_{\varphi \in \operatorname{Diff}(M)} \left(\inf_{c:[0,1]\to \operatorname{Imm}} \left[\int_0^1 G_c(c_t, c_t)\, dt \right] + \lambda\, \widehat{\mathrm{FM}}(\varphi(\boldsymbol{\theta}_{0,\mathrm{ref}}), \boldsymbol{\theta}_{1,\mathrm{ref}}) \right), \end{aligned}$$

where we consider in the inner infimum only paths satisfying $c(0, \cdot) = c_{0,\mathrm{ref}}$ and $c(1, \cdot) = c_{1,\mathrm{ref}} \circ \varphi$. ■

2.4 Symmetrization of the feature matching term

With the definition in (12), the energy is not symmetric with respect to the two shapes, because the feature points are treated differently. It is, however, straightforward to symmetrize the energy functional by defining

$$\mathcal{E}_{\mathrm{sym}}(\boldsymbol{\theta}_0, \mathbf{C}_0, \boldsymbol{\theta}_1, \mathbf{C}_1)(c) = \int_0^1 G_c(c_t, c_t) dt + \lambda\big(\mathrm{FM}(c(1, \boldsymbol{\theta}_0), \mathbf{C}_1) + \mathrm{FM}(c(0, \boldsymbol{\theta}_1), \mathbf{C}_0)\big).$$

This energy functional satisfies the invariance property

$$\mathcal{E}(\varphi^{-1}(\boldsymbol{\theta}_0), \mathbf{C}_0, \varphi^{-1}(\boldsymbol{\theta}_1), \mathbf{C}_1)(c \circ \varphi) = \mathcal{E}(\boldsymbol{\theta}_0, \mathbf{C}_0, \boldsymbol{\theta}_1, \mathbf{C}_1)(c)$$

for any diffeomorphism φ of M. From this energy functional we obtain a distance on the space of parametrized curves with feature points,

$$d_{P,\mathrm{sym}}\big((c_0, \boldsymbol{\theta}_0), (c_1, \boldsymbol{\theta}_1)\big) := \inf_{c:[0,1]\to \operatorname{Imm}} \mathcal{E}_{\mathrm{sym}}\big(\boldsymbol{\theta}_0, c_0(\boldsymbol{\theta}_0), \boldsymbol{\theta}_1, c_1(\boldsymbol{\theta}_1)\big)(c),$$

where the infimum is taken over all paths c such that $c(0,\cdot) = c_0$ and $c(1,\cdot) = c_1$. Now, the invariance property of $\mathcal{E}$ implies that

$$d_{P,\mathrm{sym}}\big((c_0 \circ \psi, \psi^{-1}(\boldsymbol{\theta}_0)), (c_1 \circ \psi, \psi^{-1}(\boldsymbol{\theta}_1))\big) = d_{P,\mathrm{sym}}\big((c_0, \boldsymbol{\theta}_0), (c_1, \boldsymbol{\theta}_1)\big),$$

whenever ψ is a diffeomorphism of M. This allows, similar to Theorem 6, to define a symmetric similarity measure on the shape space of unparametrized curves with feature points by

$$d(([c_0], \mathbf{C}_0)), ([c_1], \mathbf{C}_1)) = \inf_{\varphi \in \mathrm{Diff}(M)} d\big((c_0, c_0^{-1}(\mathbf{C}_0)), (c_1 \circ \varphi, \varphi^{-1} \circ c_1^{-1}(\mathbf{C}_1))\big). \tag{19}$$

The problem of this similarity measure is that the computation of the infimum in (19) requires the evaluation of terms of the form $\mathrm{FM}(\varphi^{-1} \circ c_1^{-1}(\mathbf{C}_1), \mathbf{C}_0)$, which involve the inverse of the diffeomorphism φ. In particular for derivative based optimization methods like gradient descent, this poses problems, as they would require in addition to φ^{-1} also its derivative. For this reason we have used only the non-symmetric similarity measure in all the computational examples below. We note, however, that discretizations of the symmetric term can, in certain cases, be minimized efficiently with an approach based on dynamic programming (see Sect. 3.2).

3 Matching feature curves with the elastic metric.

In the following, we will study one particular choice for both the Riemannian metric and the feature matching term. Our choice of the Riemannian metric G on $\mathrm{Imm}(M, \mathbb{R}^d)$ is the elastic metric with coefficients $a = 1$, $b = \frac{1}{2}$, see equation (4). This is particularly beneficial if we work on open curves. In this case, we have an explicit formula for the induced geodesic distance of the metric G, cf. Theorem 1. For the feature point matching we use the squared ℓ^2–norm on the parameter space M with respect to some reference curves $c_{i,\mathrm{ref}} =: c_i$. We will discuss different choices for the feature matching term below in Remarks 16 and 17.

As a direct consequence, we obtain the following formula for the matching functional for feature curves:

Corollary 10. *Using the elastic metric G with coefficients $a = 1, b = \frac{1}{2}$ and the ℓ^2–norm error term the similarity measure* (18) *on the set of open feature curves reads as:*

$$\begin{aligned} &d\left(([c_0], \mathbf{C}_0), ([c_1], \mathbf{C}_1)\right) \\ &\quad = \inf_{\varphi \in \mathrm{Diff}(S^1)} \left(\int_0^{2\pi} \| \frac{c_0'}{\sqrt{|c_0'|}} - \sqrt{\varphi'} \frac{c_1' \circ \varphi}{\sqrt{|c_1'| \circ \varphi}} \|_{\mathbb{R}^d}^2 d\theta + \lambda \sum_{i=1}^{n} |\varphi(\theta_0^i) - \theta_1^i|^2 \right). \end{aligned} \tag{20}$$

Here, $(c_j, \boldsymbol{\theta}_j)$ are arbitrary representatives of the shapes $([c_j], \mathbf{C}_j)$.

This observation yields the following strategy for solving the feature curves matching problem:

- Minimize

$$\mathcal{E}_1^{\mathrm{op}}(\varphi) := \int_0^{2\pi} \| \frac{c_0'}{\sqrt{|c_0'|}} - \sqrt{\varphi'}\frac{c_1' \circ \varphi}{\sqrt{|c_1'| \circ \varphi}} \|_{\mathbb{R}^d}^2 d\theta + \lambda \sum_{i=1}^{n} |\varphi(\theta_0^i) - \theta_1^i|^2 \,, \tag{21}$$

  over $\varphi \in \mathrm{Diff}([0, 2\pi])$.

- Calculate the geodesic connecting c_0 to $c_1 \circ \varphi$ using the explicit formula from Theorem 1.

Remark 11. Note, that the minimizer of (21) will in general not be a diffeomorphism, but will only have a non-negative derivative, cf. [34] To guarantee existence of the minimizer in the diffeomorphism group one would need to use a stronger metric on $\mathrm{Imm}(M, \mathbb{R}^d)$ see [11]. ■

For closed curves the situation is more complicated, since there is no explicit formula for the geodesic distance. Thus the matching functional does not simplify and the algorithm for closed curves reads as:

- Minimize

$$\mathcal{E}_1^{\mathrm{cl}}(\varphi) := \mathrm{dist}(c_0, c_1 \circ \varphi)^2 + \lambda \sum_{i=1}^{n} |\varphi(\theta_0^i) - \theta_1^i|^2. \tag{22}$$

  over $\varphi \in \mathrm{Diff}([0, 2\pi])$.

- Numerically calculate the geodesic connecting c_0 to $c_1 \circ \varphi$.

Remark 12. Note, that the first minimization involves the calculation of the geodesic distance on the space of closed curves for each evaluation of the energy functional. In contrast to the case of open curves, there exists no explicit formula for it. However, due to the characterization of the image of the SRVT, there are fast and efficient ways to numerically calculate the geodesic distance, which are described in detail in [29, 6]. The same algorithms can then be used to calculate the minimizing geodesic of step 2. ■

In the following we will present two methods to minimize these functionals: a dynamic programming approach and a gradient descent algorithm.

3.1 A gradient descent approach

The integral component of the gradient descent algorithm – the variation of the energy functional (21) – will be derived in the following lemma. To simplify the exposition we introduce the notation

$$q \star \varphi := \sqrt{\varphi'}(q \circ \varphi), \tag{23}$$

for the action of the diffeomorphism group on the space of SRV-transformed functions. Then we have:

Lemma 13. *The variation of $\mathcal{E}^{\mathrm{op}}$ in direction $\delta\varphi$ is given by:*

$$\begin{aligned}\delta\mathcal{E}_1(\varphi)(\delta\varphi) = \int_0^{2\pi} &\langle q_0 - q_1 \star \varphi, \delta\varphi'.(q_1 \star \varphi) - 2(q_1 \star \varphi)'.\delta\varphi\rangle_{\mathbb{R}^d} d\theta \\ &+ 2\lambda \sum_{i=1}^{n} (\varphi(\theta_0^i) - \theta_1^i)\delta\varphi(\theta_0^i).\end{aligned} \tag{24}$$

The L^2–gradient of the energy functional (21) *is then given by:*

$$\operatorname{grad}(\mathcal{E}_1(\varphi)) = -\left\langle q_0, \frac{(q_1 \star \varphi)'}{\varphi'}\right\rangle_{\mathbb{R}^d} + \left\langle q_0', \frac{q_1 \star \varphi}{\varphi'}\right\rangle_{\mathbb{R}^d} + 2\lambda \sum_{i=1}^{n} (\varphi(\theta) - \theta_1^i)\delta_{\theta_0^i}(\theta),$$

where $\delta_{\theta_0^i}$ denotes the delta distribution and $q_j = R(c_j)$.

Proof. Using the notation $q_j = R(c_j)$ the Energy functional can be written as

$$\mathcal{E}_1(\varphi) = \|q_0 - q_1 \star \varphi\|_{L^2}^2 + \lambda \sum_{i=1}^{n} |\varphi(\theta_0^i) - \theta_1^i|^2 .$$

We will calculate the variation of the two parts separately. For the first part we have:

$$\begin{aligned}\delta\left(\|q_0 - q_1 \star \varphi\|_{L^2}^2\right)(\delta\varphi) &= \delta\left(\|q_0 - \sqrt{\varphi'}(q_1 \circ \varphi)\|_{L^2}^2\right)(\delta\varphi) \\ &= -2\int_0^{2\pi} \langle q_0 - \sqrt{\varphi'}(q_1 \circ \varphi), \frac{\delta\varphi'}{2\sqrt{\varphi'}} q_1 \circ \varphi + \sqrt{\varphi'} q_1' \circ \varphi \, \delta\varphi\rangle_{\mathbb{R}^d} d\theta \\ &= -2\int_0^{2\pi} \langle q_0 - q_1 \star \varphi, \frac{\delta\varphi'}{2\varphi'} q_1 \star \varphi + (q_1' \star \varphi)\, \delta\varphi\rangle_{\mathbb{R}^d} d\theta \,.\end{aligned}$$

To read off the L^2-gradient we have to integrate by parts the $\delta\varphi'$ term. Since $\delta\varphi$ vanishes at the boundary we have

$$\begin{aligned}-2\int_0^{2\pi} \langle q_0 - q_1 \star \varphi, \frac{\delta\varphi'}{2\varphi'} q_1 \star \varphi\rangle_{\mathbb{R}^d} d\theta &= 2\int_0^{2\pi} \delta\varphi \left(\langle q_0 - q_1 \star \varphi, \frac{q_1 \star \varphi}{2\varphi'}\rangle_{\mathbb{R}^d}\right)' d\theta \\ = \int_0^{2\pi} &\delta\varphi \langle q_0' - (q_1 \star \varphi)', \frac{q_1 \star \varphi}{\varphi'}\rangle_{\mathbb{R}^d} d\theta \\ &+ \int_0^{2\pi} \delta\varphi \langle q_0 - (q_1 \star \varphi), \frac{(q_1 \star \varphi)'\varphi' - (q_1 \star \varphi)\varphi''}{\varphi'^2}\rangle_{\mathbb{R}^d} d\theta.\end{aligned}$$

Using that

$$q_1' \star \varphi = \frac{(q_1 \star \varphi)'}{\varphi'} - \frac{1}{2}(q_1 \star \varphi)\frac{\varphi''}{(\varphi')^2} \,, \tag{25}$$

we obtain the gradient of the first term:

$$\begin{aligned}\operatorname{grad}\left(\|q_0-q_1\star\varphi\|_{L^2}^2\right)&=\left\langle q_0-q_1\star\varphi,-2q_1'\star\varphi+\frac{(q_1\star\varphi)'}{\varphi'}-\frac{(q_1\star\varphi)\varphi''}{\varphi'^2}\right\rangle_{\mathbb{R}^d}\\&\qquad+\left\langle q_0'-(q_1\star\varphi)',\frac{q_1\star\varphi}{\varphi'}\right\rangle_{\mathbb{R}^d}\\&=\left\langle q_0-q_1\star\varphi,-\frac{(q_1\star\varphi)'}{\varphi'}\right\rangle_{\mathbb{R}^d}+\left\langle q_0'-(q_1\star\varphi)',\frac{q_1\star\varphi}{\varphi'}\right\rangle_{\mathbb{R}^d}\\&=-\left\langle q_0,\frac{(q_1\star\varphi)'}{\varphi'}\right\rangle_{\mathbb{R}^d}+\left\langle q_0',\frac{q_1\star\varphi}{\varphi'}\right\rangle_{\mathbb{R}^d}.\end{aligned}$$

For the second summand we calculate

$$\begin{aligned}\delta\left(\sum_{i=1}^n|\varphi(\theta_0^i)-\theta_1^i|^2\right)(\delta\varphi)&=2\sum_{i=1}^n(\varphi(\theta_0^i)-\theta_1^i)\delta\varphi(\theta_0^i)\\&=2\sum_{i=1}^n\int_0^{2\pi}\delta_{\theta_0^i}(\theta)(\varphi(\theta)-\theta_1^i)\delta\varphi(\theta)d\theta\,.\end{aligned}$$

Putting everything together, the formula for the gradient follows. □

Remark 14. Since $\mathcal{E}_1^{\mathrm{cl}}$ can be seen as the restriction of $\mathcal{E}_1^{\mathrm{op}}$ to a co-dimension d submanifold, the gradient of $\mathcal{E}_1^{\mathrm{cl}}$ is simply given by the projection onto the tangent space of this submanifold, cf. Theorem 2. ■

Using the above formulas, the implementation of the gradient descend algorithm is straightforward. In [29], however, it has been shown that it can be beneficial to represent diffeomorphisms φ as the tuple $\varphi=(x_0,\sqrt{\varphi'})$. If one works with open curves the initial value x_0 is always zero. Then there is a one to one correspondence between φ and $\sqrt{\varphi'}$. Denoting $\psi=\sqrt{\varphi'}$, the energy functional on open curves reads as:

$$\mathcal{E}_2(\psi)=\|q_0-\psi.(q_1\circ(\int_0^\theta\psi^2d\tau))\|_{L^2}^2+\lambda\sum_{i=1}^n|\int_0^{\theta_0^i}\psi^2d\tau-\theta_1^i|^2\,.$$

We can now also derive the variation of $\mathcal{E}$ in the ψ–representation:

Lemma 15. *The variation of $\mathcal{E}_2(\psi)$ in direction $\delta\psi$ is given by:*

$$\begin{aligned}&\delta\mathcal{E}_2(\psi)(\delta\psi)\\&=-2\int_0^{2\pi}\left(\delta\psi\left\langle q_0-q_1\star\varphi,\frac{q_1\star\varphi}{\psi}\right\rangle_{\mathbb{R}^d}+\left(\int_0^\theta\psi\delta\psi d\tau\right)\langle q_0-q_1\star\varphi,2q_1'\star\varphi\rangle_{\mathbb{R}^d}\right)d\theta\\&\qquad+4\lambda\sum_{i=1}^n(\varphi(\theta_0^i)-\theta_1^i)\int_0^{\theta_0^i}\psi\delta\psi\,d\tau\end{aligned}\tag{26}$$

Proof. For a variation $\delta\psi$ we calculate

$$\begin{aligned}
&\delta\Big(\Big\|q_0 - \psi.\Big(q_1 \circ \Big(\int_0^\theta \psi^2 d\tau\Big)\Big)\Big\|^2_{L^2}\Big)(\delta\psi) \\
&= 2\int_0^{2\pi} \Big\langle q_0 - \psi.q_1 \circ \Big(\int_0^\theta \psi^2 d\tau\Big), -\delta\psi.q_1 \circ \Big(\int_0^\theta \psi^2 d\tau\Big) \\
&\qquad\qquad - 2\psi.\Big(q_1' \circ \Big(\int_0^\theta \psi^2 d\tau\Big)\Big)\int_0^\theta \psi\delta\psi\, d\tau\Big\rangle_{\mathbb{R}^d} d\theta \\
&= -2\int_0^{2\pi} \delta\psi \Big\langle q_0 - \psi.q_1 \circ \Big(\int_0^\theta \psi^2 d\tau\Big), q_1 \circ \Big(\int_0^\theta \psi^2 d\tau\Big)\Big\rangle_{\mathbb{R}^d} d\theta \\
&\quad - 4\int_0^{2\pi}\int_0^\theta \psi\delta\psi\, d\tau \Big\langle q_0 - \psi.q_1 \circ \Big(\int_0^\theta \psi^2 d\tau\Big), \psi.\Big(q_1' \circ \Big(\int_0^\theta \psi^2 d\tau\Big)\Big)\Big\rangle_{\mathbb{R}^d} d\theta\,.
\end{aligned}$$

Using that $\int_0^\theta \psi^2 d\tau = \varphi$ we obtain the formula for the first part. For the second summand we calculate

$$\delta\left(\sum_{i=1}^n \Big|\int_0^{\theta_0^i} \psi^2 d\tau - \theta_1^i\Big|^2\right)(\delta\psi) = 2\sum_{i=1}^n \Big(\int_0^{\theta_0^i} \psi^2 d\tau - \theta_1^i\Big)\int_0^{\theta_0^i} 2\psi\delta\psi\, d\tau\,. \qquad \square$$

Remark 16. We do note that a gradient descent based algorithm can also be applied if the feature matching term FM is not the squared ℓ^2-norm but rather a general differentiable function, the only difference being a corresponding modification of the last terms in the variations (24) and (26), respectively. For instance, in the case of a feature matching term $\widehat{\mathrm{FM}}$ defined on the parameter space M^n, the last term in (26) becomes

$$2\lambda\sum_{i=1}^n \partial_i\,\widehat{\mathrm{FM}}(\varphi(\boldsymbol{\theta}_0) - \boldsymbol{\theta}_1)\int_0^{\theta_0^i} \psi\delta\psi\, d\tau.$$

■

3.2 Dynamic Programming

As an alternative to the gradient descent method discussed above, Dynamic Programming (DP) is often used to determine a piecewise linear approximation of the optimal parametrization. We begin by introducing a local version of the energy functional (21).

Let $\mathcal{I} = \{\tau_0, \ldots, \tau_M\}$ be a discretization of the interval $[0, 2\pi]$. Given $k < i \in \mathcal{I}$ and $l < j \in \mathcal{I}$ and a strictly increasing function φ satisfying $\varphi([k,i]) = [l,j]$, we define

$$\bar{\mathcal{E}}_1(\varphi; k, l; i, j) := \int_k^i \Big|\frac{c_0'}{\sqrt{|c_0'|}} - \sqrt{\varphi'}\frac{c_1' \circ \varphi}{\sqrt{|c_1'|} \circ \varphi}\Big|^2_{\mathbb{R}^d} d\theta + \lambda \sum_{m: l < \theta_1^m \le j} |\varphi(\theta_0^m) - \theta_1^m|^2. \quad (27)$$

In the special case where φ is the linear function

$$\varphi_{k,l;i,j}(\tau) := l + (\tau - k)\frac{j-l}{i-k},$$

and

$$q_{k,l;i,j}(\tau) := \frac{c_1' \circ \varphi_{k,l;i,j}}{\sqrt{|c_1' \circ \varphi_{k,l;i,j}|}} \sqrt{\frac{j-l}{i-k}},$$

is the corresponding SRV transform of the reparametrized curve, the energy functional reduces to

$$E(k,l;i,j) := \bar{\mathcal{E}}_1(\varphi_{k,l;i,j};k,l;i,j) = \int_k^i |q_0 - q_{k,l;i,j}|^2_{\mathbb{R}^d}\,d\theta + \lambda \sum_{m : l < \theta_1^m \leq j} |\varphi_{k,l;i,j}(\theta_0^m) - \theta_1^m|^2. \tag{28}$$

Denote now by Φ the set of all piecewise linear and increasing homeomorphisms $\varphi\colon [0,2\pi] \to [0,2\pi]$ with vertices on the grid $\mathcal{I} \times \mathcal{I}$. Denote moreover by $\Phi_{k,l}$ the set of all $\varphi \in \Phi$ satisfying $\varphi(k) = l$. Now let for i, $j \in \mathcal{I}$

$$H(i,j) := \min_{\varphi \in \Phi_{i,j}} \bar{\mathcal{E}}_1(\varphi;0,0;i,j)$$

and denote by $\varphi_{i,j}$ the (any) corresponding minimizer. That is, $H(i,j)$ is the minimal energy required for matching the curve segments $c_0|_{[0,i]}$ and $c_1|_{[0,j]}$ using a piecewise linear reparametrization defined on the given grid. In order to find a global reparametrization, we need to find $H(2\pi,2\pi)$ and a corresponding optimal reparametrization $\bar{\varphi} := \varphi_{2\pi,2\pi}$.

Now note that H satisfies the recursion

$$H(i,j) = \min_{k,l \in \mathcal{I},\ k<i,\ l<j} E(k,l;i,j) + H(k,l), \tag{29}$$

because of the additivity of $\bar{\mathcal{E}}_1$. Thus $\varphi_{i,j}$ is given by

$$\varphi_{i,j}(\tau) = \begin{cases} \varphi_{k,l;i,j}(\tau) & \tau \in [k,i], \\ \varphi_{k,l}(\tau) & \tau \in [0,k], \end{cases} \quad \text{with } (k,l) \in \operatorname*{arg\,min}_{k,l \in \mathcal{I},\ k<i,\ l<j} E(k,l;i,j) + H(k,l). \tag{30}$$

In practice, this computation consists of two steps. In a first step, we create the $M \times M$ matrix H inductively while keeping track of the minimizing indices k and l (see (29)). In the second step, we determine the function $\bar{\varphi}$ by backtracking the minimizing indices and using formula (30).

In order to speed up the computation, we can restrict the set of admissible indices in (29) and consider only indices k, l close to i, j. In practice, this corresponds to a restriction of the possible slopes of the piecewise linear reparametrization $\bar{\varphi}$. See Fig. 1 for an example.

Remark 17. The minimization approach based on dynamic programming can also be applied for non-quadratic feature matching terms, as long as they decompose as

$$\mathrm{FM}(\mathbf{C}_0, \mathbf{C}_1) = \sum_m \mathrm{FM}_m(C_0^m, C_1^m),$$

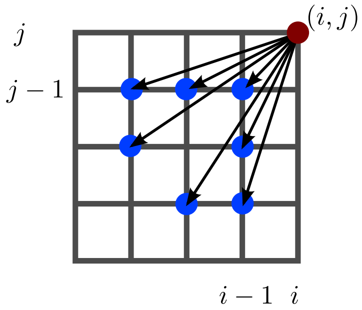

Figure 1: The dynamic programming algorithm can be sped up massively by only considering predecessor indices (k, l) close to the current index (i, j) in (29).

with $\mathrm{FM}_i : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}_{\geq 0}$. For this, one only has to replace in (28) the last sum by

$$\sum_{m: l < \theta_1^m \leq j} \mathrm{FM}_m\big(c_0(\varphi_{k,l;i,j}(\theta_0^m)), c_1(\theta_1^m)\big).$$

Note that this can be also used to implement hard constraints on the deviation of the feature points by setting

$$\mathrm{FM}_m(C_0^m, C_1^m) = \begin{cases} +\infty & \text{if } \|C_0^m - C_1^m\| > d_m, \\ 0 & \text{if } \|C_0^m - C_1^m\| \leq d_m, \end{cases}$$

for some hard bounds $d_m \geq 0$.

In addition, the dynamic programming approach readily extends to the symmetrization discussed in Section 2.4, again as long as the feature matching terms decompose. Here, the last sum in (28) has to be replaced by

$$\sum_{m: l < \theta_1^m \leq j} \mathrm{FM}_m\big(c_0(\varphi_{k,l;i,j}(\theta_0^m)), c_1(\theta_1^m)\big) + \sum_{p: k < \theta_0^p \leq i} \mathrm{FM}_p\big(c_1(\varphi_{k,l;i,j}^{-1}(\theta_1^p)), c_0(\theta_0^p)\big).$$

We stress here that the function $\varphi_{k,l;i,j}$ is linear on the interval $[k, i]$, and thus its inverse, which appears in the formula above, can be trivially computed. ■

4 Applications

As demonstrated in [14], shape matching techniques can be applied to certain computer animations such as, for instance, human walking motions. This has uses in the entertainment industry (movie and TV production, and especially video games) as well as potential biomedical applications.

We will be working with skeletal animations, where motions are described in terms of bones and joints in an approximation of a human skeleton. A typical approach to

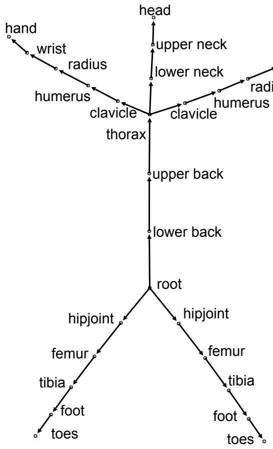

Figure 2: This skeleton, which is based on data from the CMU Graphics Lab Motion Capture Database [12], was used for the animation experiments. Figure taken from [14].

generate such animation data is to use *motion capturing* methods, where a stuntman's motions are recorded by multiple cameras in a studio. By tracking a multitude of points on the stuntman's body as he moves, the corresponding skeletal animation can be recovered. These animations face a number of limitations, however, and often require additional postprocessing. We refer to [24, 10, 16, 17, 14] for more details and examples.

A *skeleton* is a directed acyclic graph where vertices and edges represent bones and joints, respectively. A joint represents a transformation relationship between two bones. In the case of human motions, transformations between bones are restricted to rotations. Joints can have one to three degrees of freedom. For example, the knee has one degree of freedom while the foot has two, and the shoulder has three. Fig. 2 shows the skeleton used for our numerical experiment. A character?s pose is then specified by assigning values to all degrees of freedom in the skeleton (i.e., all the joints). Formally each such configuration is an element of joint space $J = \mathrm{SE}(3)^d$. An animation is then a curve from a time interval $I = [a, b]$ into joint space J, specifying a character pose for every point in time. In this article decided to ignore the manifold structure of the joint space J, but instead representing rotations using Euler angles, and thus we can collect all degrees of freedom of all joints in the skeleton as a high-dimensional torus, i.e.,

$$\mathcal{J} := \mathbb{T}^d,$$

where d denotes the total number of degrees of freedom in the skeleton.

We can now unroll the joint-space torus in $\mathbb{R}^d$ and thus an animation is then a path from the time interval $[a, b]$ to $\mathbb{R}^d$, so that for every point in time we get a pose of the skeleton. Thus we have represented animations as parametrized curves with values in $\mathbb{R}^d$. Then the shape matching techniques developed previously can then be

No. of points	Elastic reparam.	Correct features	Incorrect features
100	17 s (6)	11 s (3)	22 s (6)
200	32 s (6)	22 s (3)	50 s (6)
400	67 s (6)	48 s (3)	95 s (8)

Table 1: Computation times for Fig. 4.

applied to these curves. In [13] the authors tackled the problem of matching curves with values in a Lie-group and applied it to animations. The main difference from the work presented here to the work in [14, 13] is the incorporation of feature points into the numerical framework. In Sect. 4.2, we will show how feature matching can be successfully used to complement such existing shape matching methods. It would be interesting to combine the manifold valued framework with our framework allowing for feature point information.

We will begin, however, with a few examples of feature matching for planar curves to demonstrate the effects the additional feature term has on the curve matching.

4.1 2d-curves

As would be expected, adding feature points to shape matching can have a big impact on the resulting paths.

As a first example, we consider the matching of two open curves, the first of which has three maxima and minima, while the second only has two (see Fig. 3). Using only the elastic matching term without any specification of feature points, the resulting minimum energy path between the two curves approximately maps the first and the last extremum of the first curve to the first and the last extremum of the second curve, while the extremum in the middle vanishes slowly.

Adding feature points, one can change the behavior of the optimal path significantly. If, for instance, feature points are set on the last extremum of the first curve but in the vicinity of only the last extremum of the second curve, then the optimal path tries, during its evolution, to merge the last extrema of the first curve, while its lower portion is matched quite closely to the lower half of the second curve (see Fig. 3, upper left). Different behaviors follow from different choices of the feature points.

Fig. 4 shows similar behavior for closed curves. When using a purely elastic matching term, moving from the first hand-pose to second one seen in Fig. 4, we end up with visually unappealing interpolations. By picking to corresponding fingertips on both hands as feature points, the algorithm achieves a much more natural looking transition from one shape to the other. However, feature points need to be selected carefully, as the last row in Fig. 4 shows. By attempting to match for example the thumb in one hand shape to the midway point between index and middle finger on the second hand shape, we induce a physically implausible interpolation that involves growing a new thumb.

To give an indication of the computational performance of the method, Table 1 shows the time needed to calculate the geodesics in Fig. 4. The numbers in parenthe-

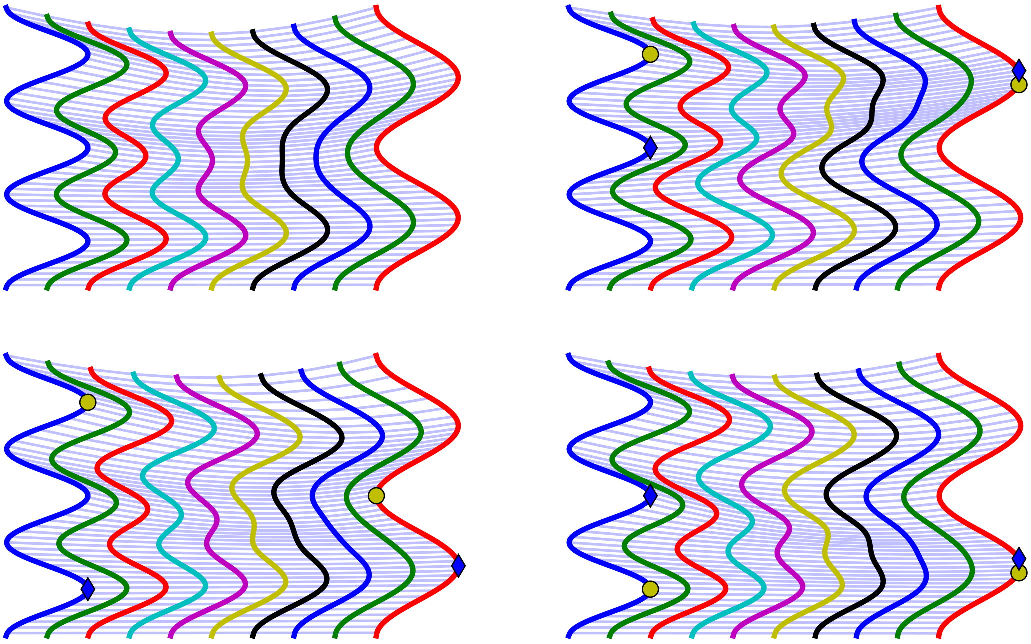

Figure 3: Effect of picking different feature points when matching two shapes. The top left figure shows results for shape matching using only an elastic energy functional without feature points. The remaining figures show matching results for different combinations of feature points. Corresponding markers on the left and right are matched, resulting in different paths between the given curves.

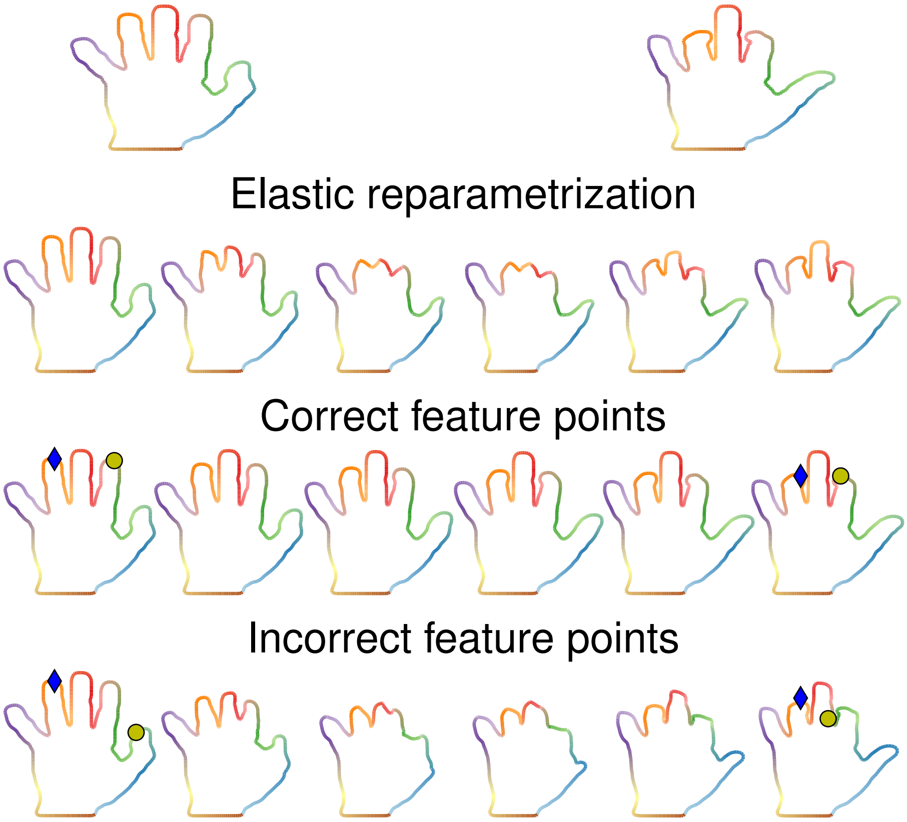

Figure 4: Effect of picking different feature points when matching two different hand shapes. In the top row, no feature points were set. The purely elastic matching produces distorted shapes along the geodesic path between the two hand shapes. In the middle row, feature points were set to match the tips of ring and index fingers correspondingly. This results in more natural interpolated shapes. In the bottom row, we see how incorrect feature matches cause some fingers to merge and new fingers to grow along the interpolation between the two shapes. Corresponding markers on the left and right are matched, resulting in different paths between the given curves. Colors along the curves indicate parametrization.

ses indicate the number of iterations that were necessary to achieve a specified level of accuracy. Note that the numerical experiments were implemented in Python, with no focus on optimization. We expect a re-implementation in, e.g., C++, to provide massive performance improvements. Furthermore, for the illustrations in Fig. 4, we have chosen very small tolerances; if necessary, these could be relaxed to further improve performance. The results in the table also indicate that a good choice of feature points can yield to a speed up in optimization (compared to a pure elastic method), whereas a bad choice of feature point might result in a slow down.

We refer to the supplementary material[2] for videos demonstrating the differences.

4.2 Applications to animations

We now turn to the use of feature point matching for animations. Human animations come in an immensely large variety. Walking motions alone can vary in speed, rhythm, step length, motions of the arms and so on. Traditional elastic matching methods (i.e., without feature points) can be applied to a large number of animations, but can sometimes struggle with animations that have large differences in rhythm, for example when matching a walking animation to a limping animation. Feature points can be used to help with determining an optimal reparametrization to align two animations in time.

Fig. 5 shows an example of using feature point information to aid in animation interpolation. The goal is to calculate interpolations between two different walking animations. These can be seen in the top row of Fig. 5. The animations differ in the number of steps, rhythm and arm motions. In addition, the character in the second animation starts walking forward only after a short delay compared to the first.

The results of four different interpolation schemes (linear interpolation of the Euler angles, elastic matching with and without reparametrization and elastic and feature matching with reparametrization) can be seen on the bottom of Fig. 5. The two superimposed lines show the trajectories of the left and right feet for each calculated interpolation. We can see how especially the varying numbers of steps in the two initial animation causes problems for the matching algorithms. Especially in the linear case, it can be seen how the right leg sometimes takes two steps during one step with the left leg. In the elastic case without reparametrization, we see that the character takes very uneven steps compared to the input data.

As feature points we picked the first three times when the left knee moves forward and past the right knee. This is already enough information for the shape matching algorithm to determine a “good” (i.e., visually convincing) interpolation.

Similar results can be seen in the second example, which shows two walking animations stepping over an obstacle of different height as seen in Fig. 6. Again, the superimposed lines represent the trajectories of the feet, with noticeable bumps as the character steps over the obstacle. As before, we have picked the first three times when the left knee moves forward and past the right knee as feature points. With

[2] Supplementary material available at `https://wiki.math.ntnu.no/optimization/skeletal_animations`.

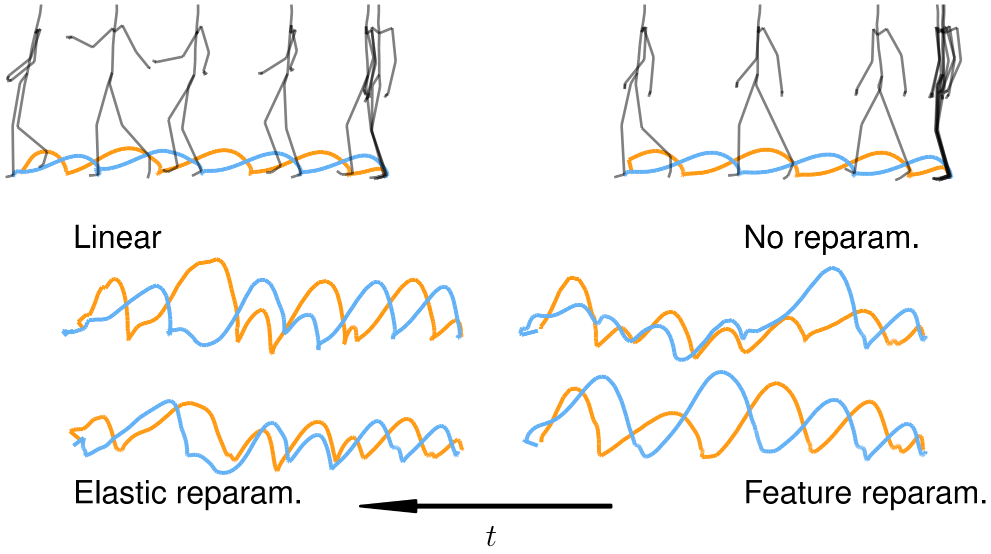

Figure 5: Example of using various methods to interpolate between two walking animations. The blue and orange lines are the trajectories of the left and right feet respectively. Note in particular how the two walking animations have different numbers of steps and how the various interpolated animations struggle with that. We have from left to right and top to bottom the following methods: linear interpolation of the Euler angles, elastic matching without reparametrization, elastic matching with reparametrization and finally elastic and feature point matching with reparametrization.

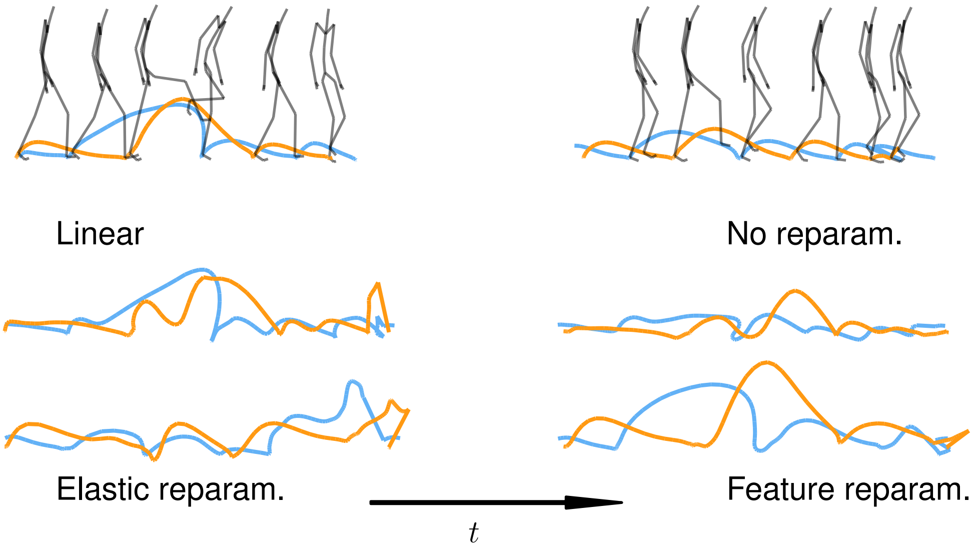

Figure 6: Example of using various methods to interpolate between two walking animations stepping over an obstacle. The blue and orange lines are the trajectories of the left and right feet respectively. We have from left to right and top to bottom the following methods: linear interpolation of the Euler angles, elastic matching without reparametrization, elastic matching with reparametrization and finally elastic and feature point matching with reparametrization.

this additional information, the feature point matching algorithm manages to produce a visually convincing interpolation between the two animations, whereas the other matching algorithms fail and produce only garbled results.

We refer to the supplementary material[3] for a video, which demonstrates the differences and advantages of our method more clear.

References

[1] M. F. Abdelkader, W. Abd-Almageed, A. Srivastava, and R. Chellappa. Silhouette-based gesture and action recognition via modeling trajectories on Riemannian shape manifolds. *Computer Vision and Image Understanding*, 115(3):439 – 455, 2011.

[2] D. Alekseevsky, A. Kriegl, M. Losik, and P. W. Michor. The Riemannian geometry of orbit spaces—the metric, geodesics, and integrable systems. *Publ. Math. Debrecen*, 62(3-4):247–276, 2003. Dedicated to Professor Lajos Tamássy on the occasion of his 80th birthday.

[3] Martin Bauer, Martins Bruveris, Philipp Harms, and Jakob Møller Andersen. Curve Matching with Applications in Medical Imaging. In *5th MICCAI Workshop on Mathematical Foundations of Computational Anatomy*, 2015.

[4] Martin Bauer, Martins Bruveris, Philipp Harms, and Jakob Møller Andersen. Second order elastic metrics on the shape space of curves. In H. Drira, S. Kurtek, and P. Turaga, editors, *Proceedings of the 1st International Workshop on DIFFerential Geometry in Computer Vision for Analysis of Shapes, Images and Trajectories (DIFF-CV 2015)*, pages 9.1–9.11. BMVA Press, September 2015.

[5] Martin Bauer, Martins Bruveris, Philipp Harms, and Jakob Møller Andersen. A numerical framework for sobolev metrics on the space of curves. To appear in *SIAM J. Imaging Sci.*, 2016.

[6] M. Bauer, M. Bruveris, S. Marsland, and P. W. Michor. Constructing reparameterization invariant metrics on spaces of plane curves. *Differential Geom. Appl.*, 34:139–165, 2014.

[7] M. Bauer, M. Bruveris, and P. W. Michor. Overview of the Geometries of Shape Spaces and Diffeomorphism Groups. *J. Math. Imaging Vision*, 50(1-2):60–97, 2014.

[8] M. Bauer, M. Bruveris, and P. W. Michor. R-transforms for Sobolev H^2-metrics on spaces of plane curves. *Geometry, Imaging and Computing*, 1(1):1–56, 2014.

[9] M. Bauer and P. Harms. Metrics on Spaces of Surfaces where Horizontality equals Normality. *http://arxiv.org/abs/1403.1436*, 2014.

[3] Supplementary material available at `https://wiki.math.ntnu.no/optimization/skeletal_animations`.

[10] A. Bruderlin and L. Williams. Motion signal processing. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 97–104. ACM, 1995.

[11] M. Bruveris, P. W. Michor, and D. Mumford. Geodesic Completeness for Sobolev Metrics on the Space of Immersed Plane Curves. *Forum of Mathematics, Sigma*, 2, 8 2014.

[12] Carnegie-Mellon. Carnegie-mellon mocap database., 2003.

[13] E. Celledoni, M. Eslitzbichler and A. Schmeding, Shape analysis on Lie groups with application in computer animation. *Journal of Geometric Mechanics.* 8, (3) 2016.

[14] M. Eslitzbichler. Modelling character motions on infinite-dimensional manifolds. *The Visual Computer*, page 112, 2014.

[15] M. Fuchs, B. Jüttler, O. Scherzer, and H. Yang. Shape metrics based on elastic deformations. *J. Math. Imaging Vision*, 35(1):86–102, 2009.

[16] L. Kovar and M. Gleicher. Flexible automatic motion blending with registration curves. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '03, pages 214–224, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association.

[17] L. Kovar and M. Gleicher. Automated extraction and parameterization of motions in large data sets. In *ACM Transactions on Graphics (TOG)*, volume 23, pages 559–568. ACM, 2004.

[18] S. Kurtek, A. Srivastava, E. Klassen, and H. Laga. Landmark-guided elastic shape analysis of spherically-parameterized surfaces. *Computer Graphics Forum*, 32(2pt4):429–438, 2013.

[19] H. Laga, S. Kurtek, A. Srivastava, and S. J. Miklavcic. Landmark-free statistical analysis of the shape of plant leaves. *J. Theoret. Biol.*, 363:41–52, 2014.

[20] A. Mennucci, A. Yezzi, and G. Sundaramoorthi. Properties of Sobolev-type metrics in the space of curves. *Interfaces Free Bound.*, 10(4):423–445, 2008.

[21] P. W. Michor and D. Mumford. An overview of the Riemannian metrics on spaces of curves using the Hamiltonian approach. *Appl. Comput. Harmon. Anal.*, 23(1):74–113, 2007.

[22] W. Mio and A. Srivastava. Elastic-string models for representation and analysis of planar shapes. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–10–II–15 Vol.2, 2004.

[23] W. Mio, A. Srivastava, and S. Joshi. On Shape of Plane Elastic Curves. *Int. J. Comput. Vision*, 73(3):307–324, July 2007.

[24] T. Pejsa and I. Pandzic. State of the art in example-based motion synthesis for virtual characters in interactive applications. *Computer Graphics Forum*, 29(1):202–226, 2010.

[25] C. Samir, P.-A. Absil, A. Srivastava, and E. Klassen. A gradient-descent method for curve fitting on Riemannian manifolds. *Found. Comput. Math.*, 12(1):49–73, 2012.

[26] J. Shah. H^0-type Riemannian metrics on the space of planar curves. *Quart. Appl. Math.*, 66(1):123–137, 2008.

[27] J. Shah. An H^2 Riemannian metric on the space of planar curves modulo similitudes. *Adv. in Appl. Math.*, 51(4):483–506, 2013.

[28] E. Sharon and D. Mumford. 2D-Shape Analysis Using Conformal Mapping. *International Journal of Computer Vision*, 70:55–75, 2006.

[29] A. Srivastava, E. Klassen, S. Joshi, and I. Jermyn. Shape Analysis of Elastic Curves in Euclidean Spaces. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(7):1415–1428, 2011.

[30] J. Su, S. Kurtek, E. Klassen, and A. Srivastava. Statistical analysis of trajectories on Riemannian manifolds: bird migration, hurricane tracking and video surveillance. *Ann. Appl. Stat.*, 8(1):530–552, 2014.

[31] Wei Liu. *A Riemannian Framework For Annotated Curves Analysis*. PhD thesis, The Florida State University, 2011.

[32] Q. Xie, S. Kurtek, and A. Srivastava. Analysis of AneuRisk65 data: Elastic shape registration of curves. *Electron. J. Stat.*, 8:1920–1929, 2014.

[33] L. Younes. Computable elastic distances between shapes. *SIAM J. Appl. Math.*, 58(2):565–586 (electronic), 1998.

[34] L. Younes, P. W. Michor, J. Shah, and D. Mumford. A metric on shape space with explicit geodesics. *Atti Accad. Naz. Lincei Cl. Sci. Fis. Mat. Natur. Rend. Lincei (9) Mat. Appl.*, 19(1):25–57, 2008.