Affine-Invariant, Elastic Shape Analysis of Planar Contours

Darshan Bryner and Anuj Srivastava
Department of Statistics
Florida State University, Tallahassee, FL
{dbryner, anuj}@stat.fsu.edu

Eric Klassen
Department of Mathematics
Florida State University, Tallahassee, FL
klassen@math.fsu.edu

Abstract

We present a Riemannian framework for analyzing shapes of planar contours in which metrics and other analyses are invariant to affine transformations and re-parameterizations of contours. Current methods that are affine invariant are restricted to point sets and do not handle full curves, while methods that analyze parameterized curves are restricted to equivalence under similarity transformation (rigid motion and scale). We construct a pre-shape manifold of standardized curves—curves whose centroid is at the origin, are of unit length, and their x and y coordinates are uncorrelated—and develop a path-straightening technique for computing geodesics on this nonlinear manifold under the elastic Riemannian metric. The removal of the rotation and the re-parameterization groups results in a quotient space, termed affine elastic shape space, and the resulting geodesic paths exhibit an improved matching of features across curves. These geodesics are used for shape comparison, retrieval, and statistical modeling of given curves. Experimental results using both simulated and real data, and an application involving pose-invariant activity recognition, demonstrate the success of this framework.

1. Introduction

Shape analysis plays an important role in computer vision and image analysis with far-reaching applications in medical diagnostics, target recognition, activity analysis, and other branches of science. A large majority of the past work in shape analysis was based on representation of planar contours using a finite set of points (or landmarks) under the equivalence relation based on similarity transformations (rigid translation, rotation, and global scale) [6]. In recent years, however, there have been several efforts on shape analysis of parameterized curves using Riemannian metrics that are invariant to the actions of similarity transformation and re-parameterization groups [13]. This framework leads to the notion of elastic shape analysis in which curves are compared using a combination of bending and stretching of parts in order to better match features across curves. Specifically, Srivastava et al. use square-root velocity functions (SRVF’s) of given curves to compute geodesics between given shapes under the elastic metric. Elastic shape analysis has been found to be superior to non-elastic methods in terms of preserving features and compactness in capturing shape variability.

Although this body of work uses full curves rather than just selected landmarks, it has a major limitation that the shape equivalence is defined under similarity transformations only. In a variety of practical problems, especially those arising in imaging contexts, shapes can get transformed beyond a rigid motion and scale, and one often needs a large group, say the affine group, to define shape equivalences. These types of transformations occur, for example, when the image plane of a camera is not parallel to the plane containing the defining part of the shape. Also, in situations where the training and the test images of an object are from moderately different viewing angles, the transformation between the resulting silhouettes can be approximately characterized as affine. There have been a number of papers dedicated to handling this larger affine variability (see next paragraph), but only using the landmark representation of shapes. To the best of our knowledge, there does not exist any method that can compare shapes of parametrized curves such that the metrics, comparisons, and statistical models are invariant to both the re-parameterizations and affine transformations of contours. Our goal is to develop a framework that removes both affine and re-parameterizations from the representation space and performs a statistical analysis of shapes. For instance, we want the shape distance between all of the shapes in Fig. 1 to be zero, and their statistical mean should be any one of these shapes.

There have been many efforts dedicated to characterizing affine invariant shape descriptors. A prominent example of type, the Curvature Scale Space (CSS) representation [1] has been shown to be robust to some affine transformations. Affine invariant fourier descriptors (AIFD) defined
by Arbter in [2] are also popular. A major drawback to both CSS and AIFD methods is that they do not provide tools for statistical analysis of shapes in the sense that, given a set of shapes, one cannot compute their means and covariances under these methods. Sparr [11] develops affine shape theory considering shape as a finite point set. Here an affine shape is defined as a linear subspace of a larger Euclidean space, and shape comparison is made via computing distances between subspaces. Begelfor and Werman in [3] use this idea and compute distances between shapes as points on a Grassmannian manifold. Again, elasticity is not considered since their framework treats shapes as a series of discrete points, or landmarks, rather than as continuous curves. Berthilsson and Åström in [4] extend the theory in [11] to crete points, or landmarks, rather than as continuous curves.

Figure 1. Examples of affine transformations of the shape on left.

The advantage of an SRVF representation is that with this transformation, the elastic Riemannian metric becomes the $L^2$ metric [13] and greatly simplifies the analysis. Using the differential geometry of this pre-shape space we compute geodesic paths between affine standardized shapes via a numerical procedure called path straightening. The geodesic lengths provide a proper distance between affine shapes. The removal of the rotation group $SO(2)$ and the re-parameterization group $\Gamma$ from the pre-shape space leads to the shape space $S = C/(SO(2) \times \Gamma)$. Optimal re-parameterizations of curves results in not only better matching of features but also makes distances invariant to original parameterizations.

The rest of this paper is as follows. Section 2 describes a canonical (standardized) representation of an affine shape class and a mechanism for generating the canonical representation for an arbitrary curve. Section 3 presents the main theoretical result of this paper on forming the shape space of canonical representations and computing geodesic paths on this manifold. Section 4 summarizes the computations of means in affine shape space and Section 5 illustrates several applications of this framework to real data.

2. Affine Standardization of Curves

We start by introducing the standardization of planar curves under the group of affine transformations. Let $\beta : [0, 1] \to \mathbb{R}^2$ be a continuous, parameterized curve. An affine transformation of $\beta$ is defined as $\tilde{\beta} = A \beta + b$, where $A \in GL(2)$ and $b \in \mathbb{R}^2$. The affine orbit of a $\beta$ is the set $[\beta] = \{A \beta + b | A \in GL(2), b \in \mathbb{R}^2\}$. We will define the membership of the same orbit as an equivalence relation so that any two curves that are within an affine transformation are termed equivalent. Each equivalence class is going to be represented by a specific element that is defined as follows. Let $L[\beta] = \int_0^1 \| \dot{\beta}(t) \| dt$ be the length of the curve $\beta$, where $\| \cdot \|$ is the 2-norm. The centroid, or the center of mass, of $\beta$ is defined as $C[\beta] = \frac{1}{L[\beta]} \int_0^1 \beta(t) \| \dot{\beta}(t) \| dt$. The covariance of $\beta$ is defined as $\Sigma[\beta] = \frac{1}{L[\beta]} \int_0^1 (\beta(t) - C[\beta](t)) (\beta(t) - C[\beta](t))^T \| \dot{\beta}(t) \| dt$. For now we will only focus on curves of unit length. We have computational evidence that there exists, uniquely up to rotation, an element of the affine orbit of $\beta$ that has its centroid located at the origin and its covariance matrix proportional to $I$, the $2 \times 2$ identity matrix. We define the space of affine standardized curves $\mathcal{A}$ as

$$\mathcal{A} = \{ \beta : [0, 1] \to \mathbb{R}^2 \mid L[\beta] = 1, C[\beta] = 0, \Sigma[\beta] = c_\beta I \}$$

(1)

where $c_\beta$ is a real scalar. Note that by imposing the unit length condition, we may not achieve $\Sigma[\beta] = I$ exactly but only within a constant.

Our idea is to represent any arbitrary parameterized curve by the sole element (up to rotation) of its equivalence
class in $A$. So, the natural question arises that, given any
unit length curve $\beta$, how does one find a representative of
the orbit of $\beta$ that is also in $A$? Since it is trivial to rescale
and translate $\beta$ so that $L[\beta] = 1$ and $C[\beta] = 0$, we focus
on the more complicated task of setting $\Sigma[\beta] = c_3 I$. We
implement an iterative technique to search for the unique
matrix $M$ such that $A^* \beta - C[A^* \beta] = c_3 I$. Since any real, invertible
matrix $M$ can be written as $M = U P$, the product of a rotation
matrix $U$ and a symmetric positive definite matrix $P$, we need to search only for the optimal SPD matrix. We
find this optimal SPD matrix $A^*$ as a root of the function
$F(A) - I$ where

$$F(A) = \int_0^1 (A \dot{\beta}(t) - C[A\dot{\beta}(t)]) (A \dot{\beta}(t) - C[A\dot{\beta}(t)])^T ||A\dot{\beta}(t)|| dt.$$  

The details of the actual optimization are omitted due to
a lack of space. Shown in Fig. 2 are several examples of
this standardization. In the left half of the figure, each curve
represents a shape at an arbitrary affine transformation and the
corresponding spot on the right shows the standardized
version of that shape. Note that each row represents the
same shape class.

3. Geodesics in Affine Elastic Shape Space

As stated earlier, our goal is to be invariant to affine
transformations as well as re-parameterizations. To achieve
the latter invariance, we use a mathematical representation of
curves that was suggested in [13], called the square-root
velocity function (SRVF). Since each curve is represented
by the canonical (or standardized) element of its affine or-
bit, we need to further restrict the analysis to SRVF’s of
standardized curves. Also, as we are interested in closed
curves here, we further impose a closure condition on the
SRVF’s of standardized curves.

3.1. Affine Pre-Shape Space Using SRVF

For a parameterized curve $\beta : [0,1] \rightarrow \mathbb{R}^2$, its
SRVF is given by the function $q(t) = \frac{\dot{\beta}(t)}{||\dot{\beta}(t)||}$. Note
that since $q(t) ||q(t)|| = \dot{\beta}(t)$, the curve $\beta(t)$ can be
recovered from the $q$-function up to a translation. Let
$x(q,t) = \int_0^t q(u) ||q(u)|| du$ be our original curve $\beta(t)
but with seed located at the origin, i.e. $\beta(0) = 0$. Also
note that since $||q(t)||^2 = ||\dot{\beta}(t)||$, the set of all unit
length curves $B = \{ q \in L^2([0,1], \mathbb{R}^2) | \int_0^1 ||q(t)||^2 dt = 1 \}$
is the unit hypersphere in the Hilbert space $L^2$. The
centroid and covariance of a curve $x(q,t)$ can be stated in
terms of the $q$-function as follows: The centroid is given by
$C[q] = \int_0^1 x(q,t)||q(t)||^2 dt \in \mathbb{R}^2$ and the covariance is
$\Sigma[q] = \int_0^1 (a + x(q,t))(a + x(q,t))^T ||q(t)||^2 dt \in \mathbb{R}^{2 \times 2}$
where $a = -C[q]$. In order to impose the condition that the curve
$\beta$ be closed, we set $x(q;1) = 0$, i.e. the endpoint of the
curve $x(q,t)$ is equal to the initial point, the origin. Define
a mapping $\Psi : B \rightarrow \mathbb{R}^4$ as:

$$
\Psi_1(q) = \int_0^1 ((a_1 + x_1(q,t))^2 - (a_2 + x_2(q,t))^2) ||q(t)||^2 dt
\Psi_2(q) = \int_0^1 (a_1 + x_1(q,t)) (a_2 + x_2(q,t)) ||q(t)||^2 dt
\Psi_3(q) = \int_0^1 q_1(t) ||q(t)|| dt, \quad \Psi_4(q) = \int_0^1 q_2(t) ||q(t)|| dt,
$$

where a subscript indicates the $i$th coordinate in Euclidean
space, that is $\Psi_i(q) \in \mathbb{R}$ for each $i$. $\Psi_1(q) = 0$ implies that
the difference of diagonal entries in the covariance matrix is
0. $\Psi_2(q) = 0$ implies that the off-diagonal entry in the
covariance matrix is 0. Together, they imply that $\Sigma(\beta) = c_3 I$
the constraint $\Psi_3(q) = \Psi_4(q) = 0$ implies the closure
condition. Since SRVF’s are translation invariant, we don’t
need any explicit condition on the centroid. The space of
all affine-standardized, unit length, closed curves is therefore
the level set $C = \Psi^{-1}((0,0,0,0)) \subset B \subset \mathbb{L}^2$. The
space $C$ is called affine pre-shape space since it does not
consider shapes modulo rotation or re-parameterization. Let
$SO(2)$ be the set of $2 \times 2$ rotation matrices, and let $\Gamma$ be
the re-parameterization group, that is, the group of all orienta-
tion preserving homeomorphisms $\gamma : I \rightarrow I$, such that
$\gamma$ and $\gamma^{-1}$ are absolutely continuous. The quotient space
$S = C/(SO(2) \times \Gamma)$ removes these shape invariants and is
denoted affine shape space.

3.2. Path Straightening on Affine Pre-Shape Space

Now we need a tool to compute geodesic paths in $C$. For
this we will use the path straightening technique [8], where
the main idea is to initialize a path between two given points
in $C$ and iteratively straighten the path according to the gra-
dient of an appropriate energy function until it cannot be
straightened any further. The resulting path minimizes the
energy function and is consequently a geodesic. Since the
gradient of the energy function can be written analytically,
the computational cost of this approach is low.

We first summarize the path-straightening procedure for a
general Riemannian manifold $M$, with the assumption
that $M$ is a submanifold of a larger ambient Hilbert space $V$
from which $M$ inherits its Riemannian structure. Given any

Figure 2. Affine standardization: Each row shows shapes under
arbitrary affine transformation (on left side) and affine standard-
izations of each shape (on the right).
two points $p_1$ and $p_2$ in $M$, our goal is to find a geodesic between them in $M$. Let $\mathcal{H}$ be the set of all differentiable paths in $M$ that begin at $p_1$ and end at $p_2$, that is $\mathcal{H} = \{ \alpha : [0, 1] \rightarrow M \mid \alpha(0) = p_1, \alpha(1) = p_2 \}$. Define an energy function on $\mathcal{H}$ according to $E : \mathcal{H} \rightarrow \mathbb{R}_+$ by $E[\alpha] = \frac{1}{2} \int_0^1 (\dot{\alpha}(\tau), \dot{\alpha}(\tau)) d\tau$, where $\langle \cdot, \cdot \rangle$ is the chosen Riemannian metric on $M$ (in our case where $M = \mathbb{C}$, the metric is simply $L^2$ inner product on $\mathbb{C}$). It can be shown that the critical points of $E$ on $\mathcal{H}$ are precisely the geodesic paths on $M$ between $p_1$ and $p_2$. The path straightening procedure is to iteratively calculate the gradient of $E$ on $T_\alpha(\mathcal{H})$ with respect to an appropriate Riemannian metric, then update $\alpha$ towards a critical point of $E$ on $\mathcal{H}$. It was shown in [8] that under the Palais metric, the gradient of $E$ with respect to $\alpha$ is $w(\tau) = u(\tau) - \tau \ddot{u}(\tau)$, where $u$ is the covariant integral of the vector field $\dot{\alpha}(\tau)$ with zero initial value at $\tau = 0$ (i.e., $\frac{\partial u}{\partial \tau} = \frac{\partial u}{\partial \tau}$) and $\ddot{u}(\tau)$ is the vector field obtained from backward parallel translating the vector $u(1)$ along $\alpha$. Note that by construction $w(0) = w(1) = 0$, and thus the endpoints of $\alpha$, $p_1$ and $p_2$, remain fixed in a gradient descent iteration.

We have adapted this path straightening approach for computing geodesics on $M = \mathbb{C}$. An integral step in calculating geodesic paths between elements of $\mathcal{C}$ is obtaining a basis for the normal space of $\mathcal{C}$ inside $\mathcal{B}$ at any point $q$. In order to define the normal space, we must first find the directional derivative of $\Psi$ (see Eqn. 3) at any point $q \in B$. Let $w \in T_q(\mathcal{B})$. We write the derivative $d\Psi_{j,q}(w) = \frac{d}{ds} \langle \Psi_j(q(t) + sw(t)) \rangle_{s=0}$, for $j = 1, 2, 3, 4$. By expressing the directional derivative as an inner product as follows: $d\Psi_{j,q}(w) = \langle w(t), h_j(t) \rangle$ for some functions $h^1, ..., h^4$, where $\langle \cdot, \cdot \rangle$ is the standard $L^2$ inner product, these functions will serve as a basis for the normal space. The restriction of these functions inside $T_q(\mathcal{B})$ is obtained by removing the projection along the function $q_i(t) = h^i(t) - g(t)(q(t), h^i(t))$ and $\{b_i\}$ span the normal space $\mathcal{C}$ inside $\mathcal{B}$ at $q$. We refer the reader to the Appendix for analytical expressions of the $h^i(t)$’s. Next we describe three additional subroutines needed to implement the path straightening algorithm on $\mathcal{C}$.

(1) Projection Onto the Manifold $\mathcal{C}$: For any point $\mathcal{S} \in \mathcal{C}$, we need a tool to project $q$ to the nearest point in $\mathcal{C}$. This procedure is different from the standardization presented in Section 2, although they accomplish a similar task. (Standardization is restricted to the same orbit while the projection is not.) The numerical technique for the projection algorithm is again a Newton-Raphson method. The $4 \times 4$ Jacobian matrix for this projection is defined as $J_{ij}(q) = \langle b^i(t), b^j(t) \rangle$, for $i, j = 1, 2, 3, 4$ and where the $b^i$’s are the basis vectors for the normal space of $\mathcal{C}$ at $q$.

Algorithm 1 (Projection onto $\mathcal{C}$): Let $\epsilon > 0$.

(i) Compute the residual vector $r(q) = \Psi(q)$. If $||r(q)|| < \epsilon$, stop. Otherwise continue to step (ii).

(ii) Calculate the basis vectors $b^j$, $j = 1, 2, 3, 4$ for the normal space at $q$.

(iii) Calculate the Jacobian matrix $J_{ij}(q)$ and solve the equation $J_{ij}(q)y = -r(q)$.

(iv) Define $dq = \sum_{j=1}^4 y_j b^j$. Update $q \rightarrow \cos(||dq||)q + \sin(||dq||) \frac{dq}{||dq||}$. Go to step (i). Some examples of this projection are shown in Fig. 3.

(2) Projection Onto the Tangent Space $T_q(\mathcal{C})$: Given any function $w \in L^2([0, 1], \mathbb{R}^2)$, we need a procedure to project $w$ into $T_q(\mathcal{C})$, the tangent space of $\mathcal{C}$ at any point $q \in \mathcal{C}$. We accomplish this task in two steps:

Algorithm 2 (Projection onto $T_q(\mathcal{C})$): Given a basis $\{b^j\}$, $j = 1, 2, 3, 4$, for the normal space $N_q(\mathcal{C})$.

(i) If $w \notin T_q(\mathcal{B})$, then project $w$ to $T_q(\mathcal{B})$ via $w \rightarrow w - \langle w, q \rangle q$.

(ii) Then, compute $\{b^j\}$, a Gram-Schmidt orthonormalization of $\{b^j\}$. The projection of $w$ into $T_q(\mathcal{C})$ is defined as $\tilde{w} = w - \sum_{j=1}^4 \langle w, b^j \rangle b^j$.

Note that we can skip the first step even if $w \notin T_q(\mathcal{B})$, but it will compromise numerical stability.

(3) Parallel Translation: Given two nearby points $q_1, q_2 \in \mathbb{C}$, and a tangent vector $w \in T_{q_1}(\mathcal{C})$, we need a tool to obtain the parallel translation of $w$ to $\tilde{w} \in T_{q_2}(\mathcal{C})$. Remember that $\mathbb{C} \subset \mathcal{B}$, the unit hypersphere, so in this algorithm we make use of the analytic expression for parallel translation of a vector in the tangent space of $\mathcal{B}$.

Algorithm 3 (Parallel Translation): Given two points $q_1, q_2 \in \mathbb{C}$ and $w \in T_{q_1}(\mathcal{C})$.

(i) Compute the analytic expression for parallel translation on $\mathcal{B}$ as $w \rightarrow \tilde{w} \equiv \frac{2(\langle w, q_2 \rangle)}{||w + \langle w, q_2 \rangle q_2||^2}(q_1 + q_2)$.

(ii) Let $l = ||\tilde{w}||$. Project $\tilde{w}$ onto the tangent space $T_{q_2}(\mathcal{C})$ using Algorithm 2 to obtain $\tilde{w}$.

(iii) Rescale $\tilde{w}$ via $\tilde{w} \rightarrow \frac{\tilde{w}}{||\tilde{w}||}$.

Now that we have these important subroutines, we will present a general numerical procedure for the path straightening method on $\mathcal{C}$. To implement a path straightening approach on a computer, one must work with a discretized version of the path $\alpha$. We assume a uniform partition $\{0, \frac{1}{n}, \frac{2}{n}, ..., 1\}$ of the interval $[0, 1]$ on which the calculations will be performed. Since path straightening is an iterative procedure, we need to initialize a path in $\mathcal{C}$ be-
between the two given points \( q_1 \) and \( q_2 \). We do so by computing a geodesic between \( q_1 \) and \( q_2 \) in the larger space \( B \) and projecting it point-by-point in \( C \) using Algorithm 1. If the path is discretized at time points \( \tau = k/n \), then for all \( k = 0, 1, \ldots, n \), we initialize

\[
\alpha(k/n) = \Pi\left(\frac{1}{\sin(\theta)} \left[\sin((1-(k/n))\theta)q_1 + \sin((k/n)\theta)q_2 \right]\right),
\]

where \( \theta = \cos^{-1}\left((q_1, q_2)\right) \) and \( \Pi \) is the projection on to \( C \) via Algorithm 1. It is important to note that if \( q_1 \) and \( q_2 \) are not in \( C \), we must use the affine standardization technique presented in Section 2 to select the shapes in \( C \) that lie in the same affine orbits as \( q_1 \) and \( q_2 \).

We now have all the numerical tools necessary to implement the path straightening algorithm for computing geodesics on \( C \).

**Algorithm 4 (Path Straightening in \( C \)):** Given two shapes \( q_1 \) and \( q_2 \) in \( B \), initialize a path \( \alpha \) in \( C \) using a geodesic in \( B \) and projecting via algorithm 2.

1. Compute \( \frac{d\alpha}{d\tau} \) along \( \alpha \) using finite differencing.
2. Compute covariant integral \( u \) of \( \frac{d\alpha}{d\tau} \) along \( \alpha \) (uses Algorithm 2).
3. Backward parallel translate \( u(1) \) along \( \alpha \) using Algorithm 3 to obtain \( \tilde{u} \) (uses Algorithm 2).
4. Compute the gradient vector field \( \gamma(k/n) = u(k/n) - (k/n)\tilde{u}(k/n) \) for \( k = 0, 1, \ldots, n \).
5. Update the path \( \alpha'(k/n) = \alpha(k/n) - \epsilon \gamma(k/n) \) for some \( \epsilon > 0 \), then use Algorithm 1 to obtain \( \alpha(k/n) = \Pi(\alpha'(k/n)) \).
6. Return to step 2 unless \( \|u\| \) is small.

Fig. 4 shows an example of this algorithm for constructing geodesics between a pair of sample shapes in \( C \).

![Figure 4](image)

Figure 4. Left: An example of path straightening. Each row of shapes is the result of one iteration of the path straightening algorithm from the top row (initial) to the bottom row (final). Right: Evolution of \( E \) during path straightening.

### 3.3. Geodesics in Affine Elastic Shape Space

So far we have an algorithm for computing geodesics in the pre-shape space. In order to compute geodesics in the shape space \( S = C/(SO(2) \times \Gamma) \), we simply optimize over the re-parameterization and the rotation groups. Let \( q_1, q_2 \in C \) be the SRVF’s of any two (affine-standardized) closed curves, and solve for:

\[
(\gamma^*, O^*) = \text{argmin}_{(\gamma, O) \in \Gamma \times SO(2)} \|q_1 - O(q_2, \gamma)\|^2,
\]

and then compute a geodesic path using path-straightening algorithm in \( C \) between \( q_1 \) and \( O^*(q_2, \gamma^*) \). This results in a geodesic path in \( S \) between the given curves in the SRVF space. Each point along the path can be integrated to result in the corresponding parameterized curve.

We give an example of the benefit of affine invariant elastic shape analysis by computing the geodesic path between two shapes \( \beta_1 \) and \( \beta_2 \) in various shape spaces. \( \beta_1 \) is the outline of a horizontal bar with two bumps close together on top of it. \( \beta_2 \) is constructed from \( \beta_1 \) by spreading the bumps farther apart and then applying a random affine transformation. Fig. 5 shows geodesics between \( \beta_1 \) and \( \beta_2 \) in the following shape spaces: (a) closed curve shape space with a non-elastic, bending only metric [9], (b) closed curve shape space with the elastic metric, (c) affine shape space with the elastic metric.

![Figure 5](image)

Figure 5. Geodesic paths between \( \beta_1 \) and \( \beta_2 \) in different spaces. Spaces (a), (b), and (c) are given in the text above. The path in (d) is the result of undoing any rotation or standardization in either (b) or (c).

We can see that the amount of deformation in geodesic paths (a) and (b) in Fig. 5 is much greater than that of path (c). In fact, the geodesic distances computed are (a) 0.85, (b) 0.52, and (c) 0.19. Path (d) is a smooth path from \( \beta_1 \) to \( \beta_2 \) obtained by applying to path (c) a smooth path in \( GL(2) \) from \( A_1^{-1} \) to \( O^* A_2^{-1} \). (Fig. 6 shows more examples of this technique). An important point is that since path (c) is in the space of affine standardized shapes, it will be invariant to any affine transformation of \( \beta_1 \) or \( \beta_2 \), and thus the measure of deformation will not change.

### 4. Statistics in Affine Shape Space

In the previous section, we defined the shape space of affine standardized curves as a Riemannian manifold and outlined a procedure to calculate geodesic paths, and therefore distances, between points on \( S \). Since this distance is in fact a proper distance on the shape space, we can speak of computing a sample mean and covariance, allowing us to
build shape probability models in affine shape space. Let 
\([q] \in \mathcal{S}\) be the orbit of a shape \(q \in \mathcal{C}\) under the group action 
of \(\Gamma \times SO(2)\). Given a set of shapes \(Q = \{[q_1], ..., [q_n]\}\) in 
\(\mathcal{S}\), we obtain a mean shape \([\mu] \in \mathcal{S}\) by calculating the intrin-
sic mean, or Karcher mean. The Karcher mean is defined
as the shape that minimizes the sum of square distances to all 
the shapes in \(Q\), i.e. \(\mu = \arg\min_{q \in \mathcal{C}} \sum_{i=1}^{n} d([q], [q_i])^2\),
where \(d(\cdot, \cdot)\) is the geodesic distance. An iterative algo-
rithm to obtain the Karcher mean modulo rotation and re-
parameterization is outlined in [12] and is not repeated here.
Fig. 4 shows an example by plotting given curves at arbi-
trary affine transformations (left), after the standardization
(middle), and the resulting mean shape (right).

We compute pairwise distances with three techniques:
(1) without affine standardization and with elasticity, (2) 
with affine standardization and without elasticity, and (3)
with both affine standardization and elasticity. In test (3) the
classification rates are essentially perfect since standardiza-
tion is unique up to rotation and parameterization, and the
elastic metric is invariant to rotation and parameterization.

To quantify the performance of each shape comparison
 technique, we compute a precision-recall curve to visualize
classification rate with respect to different sized retrieval
sets. Let \(Q\) denote the set of all curves i.e. queries in the
dataset. For each row \(i\) in the distance matrix, let \(A_i(Q)\)
be the set of \(K\) retrievals, i.e. the shapes that yield the \(K\)
smallest distance values. Let \(R_i(Q)\) be the group of shapes
belonging to the relevant shape class of shape \(i\). Precision
is defined as \(P_i = \frac{|A_i(Q) \cap R_i(Q)|}{|A_i(Q)|}\), and recall is defined as
\(C_i = \frac{|A_i(Q) \cap R_i(Q)|}{|R_i(Q)|}\), where \(|\cdot|\) denotes cardinality. For each
\(K = 1, ..., 200\) we compute the average value of \(P_i\) and \(C_i\)
over each row and plot the resultant precision versus recall
curve. Note that when \(K\) is equal to the class size, precision
and recall are equal. When the class size \(K = 10\), metric (1)
yields 57% precision and recall, (2) yields an improvement
to 70%, and (3) achieves near perfection at 99% precision
and recall. Shown in Fig. 8(top) is the precision-recall curve
for the three metrics. Some examples of query-results are
shown in Fig. 8(bottom) for two metrics: the elastic metric
for similarity-only invariance and the elastic affine-invariant
metric. From these results, we conclude that affine elastic
shape metric is most appropriate for shape retrieval in situ-
avations where shapes have undergone affine transformations
within shape classes.

**MCD Database:** The Multiview Curve Database (MCD)
has been constructed from the MPEG-7 database to test the
performance of shape analysis methods in the presence of
perspective distortions [15]. 40 shapes were selected from
the MPEG-7 database and printed on white paper as bi-
nary images. Seven variations of each curve were recorded
by photographing the binary images under seven different
camera angles. It is our proposal that we can approximate
via affine transformation the perspective distortions brought
about by imaging an object from different camera locations.

From the precision-recall curves, we observe better clas-
sification performance with our affine standardization algo-
rithm compared with Zuliani’s affine invariant shape matrix
descriptor in [15], which achieved about 80% precision and
recall on the MCD database. Without affine standardiza-
tion, we report about 60% precision and recall, and with
affine standardization we see a large improvement to about
90%. This result suggests that affine standardization com-
bined with elastic shape analysis is effective in classifying
shapes under perspective distortions.
Pose-Invariant Activity Classification: An important application of affine invariant shape analysis is in the field of human activity, or human motion analysis. A major issue in certain automated recognition procedures is invariance of an activity under differing pose or camera angles. Note that if the pose changes so much that certain body parts are occluded and new parts become visible, then the shape changes are too complicated to be modeled as a simple transformation. However, in case of moderate changes (<45°), one can model these changes using affine transformations. In that situation the proposed metric can be used for a pose-invariant activity classification.

In this experiment we use the UMD activity dataset [14], which consists of 100 sequences of 80 shapes each, where each sequence represents a frame-by-frame outline of a person performing a task. The dataset is divided into 10 classes, or “activities,” of 10 sequences each. Our goal is to classify a test sequence under an arbitrary viewing angle; we simulate different viewing angles by applying appropriate affine transformations on a given sequence. Since these data were captured using broadside imaging, we simulate test sequences for different views: original (camera level and broadside), narrowside (camera at level height and slightly facing the subject), top view (camera at an elevated position and broadside), top-left (camera elevated and slightly facing the subject). To generate a test sequence, we simulate a stochastic process on $GL(2)$ with an appropriate mean and apply each point of the process to the corresponding shape in the sequence. Fig. 10 shows an example of an activity sequence under these simulated views. Then, we classify this test sequence using the nearest neighbor classifier under different metrics. (The distance for classifying a sequence of shapes is the sum of the distances for individual shapes.)

<table>
<thead>
<tr>
<th>Similarity</th>
<th>Broadsided Level</th>
<th>Narrowside Level</th>
<th>Broadsided Elevated</th>
<th>Narrowside Elevated</th>
</tr>
</thead>
<tbody>
<tr>
<td>Affine</td>
<td>98%</td>
<td>97%</td>
<td>98%</td>
<td>98%</td>
</tr>
<tr>
<td></td>
<td>98%</td>
<td>97%</td>
<td>98%</td>
<td>98%</td>
</tr>
</tbody>
</table>

Table 1. Leave-one-out classification rates for human activity sequence dataset under various simulated camera angles.

Note that the classification rate of the affine standardized test sequences matches that of the original, un-transformed sequences. Since classification rate decreases under different camera angles without standardization, we conclude that...
applying our affine standardization algorithm helps in classifying sequences of human activity under various camera angles in this simulated environment. From the drastic improvement in classification rate from 51% to 98% in the simulated top camera angle, we can see that standardization would be especially useful when applied to sequences imaged from high mounted surveillance cameras.

6. Summary

Shape analysis techniques that are invariant to affine transformation are desirable in many applications of computer vision. Here we have presented a contour-based shape analysis framework based on Riemannian geometry that is invariant to affine transformation and re-parameterization of contours. By expressing 2-D shapes in a standard form that is invariant to affine transformation and re-parameterization of contours, we eliminate the skew effect brought about by the action of GL(2). The standardized shapes are then compared using geodesic length of unit-length, standardized curves modulo the rotation and re-parameterization groups. Using this Riemannian framework, we can compute sample statistics and probability models for shape classes in an affine-invariant manner.

Acknowledgements

This work was supported in part by the grants ONR N00014-09-1-0665 and NSF DMS 09-15003 to AS. The authors would also like to acknowledge the NSWC PCD ILIR N00014-09-1-0665 and NSF DMS 09-15003 to AS. The authors would also like to acknowledge the NSWC PCD ILIR Program and the support of Dr. Kerry Commander and Dr. Quyen Huynh.

A. Appendix

Here we show the analytical formulas for the functions \( h_j(t) \), \( j = 1, \ldots, 4 \) that serve as a basis for the normal space of \( C \) at \( q \). These expressions arise from the calculation of the directional derivative of \( \Psi \). Let \( f_i(t) = \|q(t)\|e_i + \frac{q(t)}{\|q(t)\|}g(t) \) for \( i = 1, 2 \), where \( e_i \) is the \( i \)th standard basis vector in \( \mathbb{R}^2 \). Let \( Q(t) = \int_0^t \|q(\tau)\|^2d\tau \). Let \( G_i(t) = \int_0^t \|q(\tau)\|^2 x_i(q; \tau)d\tau \) for \( i = 1, 2 \). Now

\[
 h^1(t) = 4q(t)a_1x_1(q; t) - 4q(t)a_2x_2(q; t) \tag{5}
 + 2a_1f_1(t)(1 - Q(t)) - 2a_2f_2(t)(1 - Q(t)) + 2q(t)x_1(q; t)^2 - 2q(t)x_2(q; t)^2
 + 2f_1(t)(a_1 - G_1(t)) - 2f_2(t)(a_2 - G_2(t)),
\]

\[
 h^2(t) = 2a_2q(t)x_1(q; t) + 2a_1q(t)x_2(q; t) \tag{6}
 + a_2f_1(t)(1 - Q(t)) + a_1f_2(t)(1 - Q(t)) + 2q(t)x_1(q; t)x_2(q; t) + f_1(t)(a_2 - G_2(t))
 + f_2(t)(a_1 - G_1(t)),
\]

\[
 h^3(t) = f_1(t), \text{ and } h^4(t) = f_2(t).
\]

References


