

# Learning Shape Correspondence for n-D curves

Alain Pitiot, Hervé Delingette and Paul M. Thompson

February 7, 2006

## Abstract

We present a learning method that introduces explicit knowledge into the shape correspondence problem. Given two input curves to be matched, our approach establishes a dense correspondence field between them, where the characteristics of the matching field closely resemble those in an *a priori* learning set. We build a shape distance matrix from the values of a shape descriptor computed at every point along the curves. This matrix embeds the correspondence problem in a highly expressive and redundant construct and provides the basis for a pattern matching strategy for curve matching. We selected the previously introduced observed transport measure as a shape descriptor, as its properties make it particularly amenable to the matching problem. Synthetic and real examples are presented along with discussions of the robustness and applications of the technique.

## 1 Introduction

“As for example in geometry [...] I define figure [shape] to be that in which the solid ends; or, more concisely, the limit of solid”. This excerpt from a dialogue between Socrates and Meno as fictitiously reported by Plato [37] illustrates how difficult it is to characterize the notion of shape, a common concept that is still in need of a satisfying definition, centuries after the first inquiries into its nature.

As pointed out by Veltkamp [44], shape is classically considered as “something geometrical”, a definition that leaves a large part to psychovisual analysis. Shape has therefore been studied both from the perspective of the human visual system and from a more abstract mathematical one (see [29, 47, 17] for reviews).

Among the studies dealing with the visual system aspect of shape, psycho-cognitive approaches such as the holistic *Gestalt* [25, 47] or reductionist approaches [19, 14] have focused on developing an artificial equivalent to the human shape analysis system, capable of both visual perception and cognition. Despite their success, they are inherently qualitative and difficult to translate into algorithms. Several more computational theories have also been proposed: Marr’s “primal sketch” theory [30, 31, 32], the so-called “shape from x” techniques (shape from shading [20], from contour [42], etc.), the dynamic hierarchical approach [24] or Leyton’s Symmetry-Curvature evolution theorem [28] to name a few.

From a more abstract mathematical point of view, a shape can be defined as an equivalence class under a group of transformations. Given a similarity measure (or measure of resemblance), which must be invariant under this transformation group, the shape of a pattern is the pattern modulo the action of the group [16]. The competing statistical theories of Bookstein [5] and Kendall [23] are two attempts to establish a mathematical framework that rigorously defines the notion of shape distance or shape similarity.

### 1.1 Shape Correspondence

Both of these mathematical approaches require that correspondences between shapes are already available *a priori*, a limitation shared by most methods in the literature. This correspondence problem arises in a broad range of image-related fields, from signal processing to pattern recognition. In computer vision for instance, the search for target patterns often requires a given template to be matched to

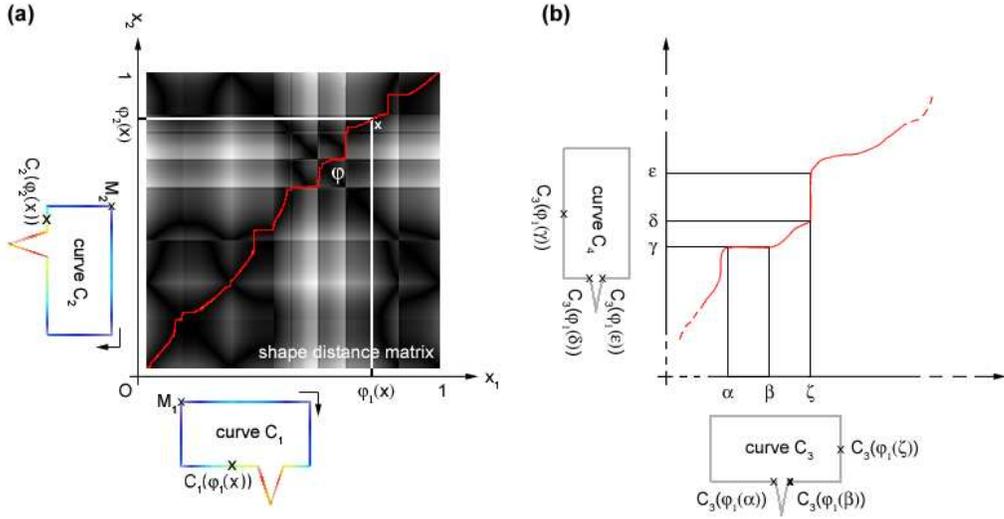


Figure 1: Illustration of the matching framework. (a) Given a corresponding pair of matched points,  $(M_1, M_2)$ , which defines the origins of axis  $x_1$  and  $x_2$ , a reparameterization corresponds to the red path  $\phi$  between  $(0,0)$  and  $(1,1)$ . As  $x$  travels along  $\phi$ ,  $\phi_1(x)$  in  $C_1$  is matched to  $\phi_2(x)$  in  $C_2$ . (b) Such a framework allows for the triangular indentation  $[\alpha-\beta]$  in  $C_3$  to be matched to a single point  $\gamma$  in  $C_4$  and conversely for  $[\delta-\epsilon]$  in  $C_4$  to map to point  $\zeta$  in  $C_3$ .

pictorial elements in an input image [33]. In computer graphics, matched curves may be used to derive a series of intermediate shapes to “morph” one into the other [21]. In medical image analysis, the objects to be matched may be instances of a given anatomical structure, for which a statistical model, or time-dependent model, is required [9]. In this paper, we approach the issue of curve matching as a process of computing a dense correspondence field between two *a priori* parameterized curves.

At a glance, defining a correspondence between two curves entails finding in them pairs of corresponding elements that share specific similarities in shape, position, or both. More formally, given two curves  $C_1$  and  $C_2$  with any *a priori* parameterizations represented by two functions  $C_1$  and  $C_2$ :

$$C_1 : \begin{cases} I_1 \subset \mathbb{R} & \rightarrow \mathbb{R}^n \\ x_1 & \mapsto C_1(x_1) \end{cases}, C_2 : \begin{cases} I_2 \subset \mathbb{R} & \rightarrow \mathbb{R}^n \\ x_2 & \mapsto C_2(x_2) \end{cases} \quad (1)$$

we are looking for a reparameterization of  $C_1$  and  $C_2$ , that is, for two functions  $f_1$  and  $f_2$ , such that  $C_1^* = C_1 \circ f_1$  and  $C_2^* = C_2 \circ f_2$  and

$$\forall x_1 \in I_1, \forall x_2 \in I_2, x_1 \text{ “close to” } x_2 \Rightarrow C_1^*(x_1) \text{ “very similar in shape to” } C_2^*(x_2) \quad (2)$$

where “very similar in shape to” is defined with respect to a given shape similarity metric.

Following [41], to allow multiple points in  $I_1$  to be matched to a single point in  $I_2$  and conversely, we restate our problem as that of finding a continuous function

$$\varphi : \begin{cases} I \subset \mathbb{R}^2 & \rightarrow I_1 \times I_2 \\ x & \mapsto (\varphi_1(x), \varphi_2(x)) \end{cases} \quad (3)$$

such that:

$$\forall x \in I, C_1(\varphi_1(x)) \text{ “very similar in shape to” } C_2(\varphi_2(x)) \quad (4)$$

Figure 1(a) illustrates this approach on a pair of rectangles with triangular indentations. The inherent symmetry of this framework is shown in (b) where an entire subcurve in  $C_3$  is matched to a single point in  $C_4$ , and the other way round.

In addition, as in [41], we would like the matching function to satisfy, the following properties:

**symmetry:** the function  $\varphi_{C_1 \rightarrow C_2}$  associated with the matching of  $C_1$  to  $C_2$  should be the inverse of  $\varphi_{C_2 \rightarrow C_1}$ , the function associated with the matching of  $C_2$  to  $C_1$ .

**consistent self-matching:** when trying to match an object to itself, the optimal reparameterization should be the identity:  $\forall \text{curve } \mathcal{C}, \varphi_{\mathcal{C} \rightarrow \mathcal{C}} = (Id, Id)$ . In general, we would like  $\varphi$  to be not too dissimilar from the identity.

## 1.2 Object matching approaches

A number of automated methods for parameterized curve or surface matching have been presented in the literature, that tackle some or all of the above issues. Extensive reviews of these approaches are available elsewhere (see [45], [29] or [26] for instance). We present in the following section a selection of techniques and comment in Section 1.3 on their lack of explicit control, an issue that we solve via a learning paradigm (see Section 3).

Trouvé and Younes detailed in [41] an axiomatic formulation for 1-D matching: they introduced, among others, the concepts of symmetry and consistent self-matching and proposed a matching framework for 2-D piecewise lines that satisfies their axioms.

One of the many suggested PDE approaches, Cohen’s approach [7] compared the bending and stretching energies of one curve,  $C_1$ , and a reparameterization of the other,  $C_2^*$ , to find the best match. In [8], several PDEs were introduced to introduce geometric information when computing a set of geodesic paths between the curves to be matched.

Techniques may also be derived from a direct combinatorial search for point correspondence. Arkin *et al.* [1] proposed an efficient polygonal line matching algorithm where the turning function (measure of the angle between the tangent to a curve and a pre-determined axis as a function of arc length) was used as a shape measure. Chui and Rangarajan’s “soft assign” algorithm [6] offers a robust way to find correspondences between noisy input point sets. Wang *et al.* [46] used geodesic interpolation to compute the dense correspondence field between two surfaces once an initial sparse set of corresponding points had been obtained with another automated shape-based matching algorithm.

Decomposition of the input objects in the spatial or frequency domains has also proved popular. In [22], the first elliptical harmonics of the expansion of the input objects (which must have spheroidal shapes) served to establish a correspondence. Similar Fourier-based techniques were reported earlier in [34].

In [39], Sebastian *et al.* used a dynamic programming approach similar to that of [41] to find the best match between two 2-D curves, using a similarity measure based on “alignment” between segments of the curves. A similar dynamic programming scheme was proposed by Gdalyahu *et al.* in [13] where the edit distance (minimum number of elementary changes required to change the string-like descriptor of one curve into that of the other curve) served as a similarity measure.

Registration and elastic warping approaches have also been investigated. In [40], Thompson *et al.* mapped the input surfaces to two spheres whose coordinates were then warped under anatomical feature curve constraints using spherical harmonic interpolation. Davatzikos *et al.* [10] also identified corresponding points on object boundaries in two images before aligning them using elastic warping. Fleuté *et al.* [12] minimized the Euclidean distance between an input shape and a registered template, which assumed smooth transition paths between them.

From an optimization point of view, Davies *et al.* [11] presented a curve matching method, in the context of searching for the most compact statistical shape model. An information-theoretic criterion was designed and controlled the correspondence between objects.

These approaches generally rely on computing a shape descriptor or a set of descriptors, either locally or globally. If objects to be matched are considered as filled contours (or regions), spatial moments can be used for matching. These are global measures, and quantify the shape of the object considered as a whole. A number of moment invariants can be derived from them: area, circularity,

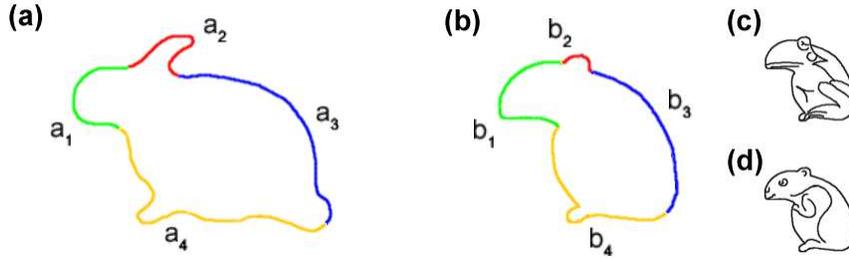


Figure 2: Matching two animal silhouettes: (a) outline of a rabbit; (b) outline could either be that of frog (c) or of squirrel (d). The matching of segments  $a_i$  and  $b_i$  therefore depends on external knowledge to choose between these two alternatives.

compactness, eccentricity, etc. (see [38] for details). Shape matrices have also been used as a compact representation of global and local shape properties Goshtasby [15] binned pixels based on their polar coordinates. Similarly, Belongie’s shape context [4] outputs for each point of an object the histogram of the positions of all the other points expressed in a log-polar coordinate system (centered on the given point).

Often, object boundaries are used instead of their entire area. In the case of curves or surfaces, a number of analytic functions can be estimated at any point: tangent, acceleration, tangent angle, curvature, or torsion for instance [43]. Signature functions and “time warps” may also be computed [45].

### 1.3 Common caveat

Despite their diversity, these matching approaches share the same drawback: their lack of control over the similarity measure introduced in equation 2, which is often defined *a priori*, once and for all, and uses only limited domain-based information (or information learned implicitly from examples). Typically, these matching processes can be reduced to optimizing a functional whose minimum corresponds to a “good” correspondence field. The difficulty of designing an adequate functional comes from the difficulty of characterizing an adequate correspondence field. In [7] for instance, the authors assume that points with similar curvature should be matched. This may suit some applications, but is not always desirable. Figure 2 illustrates such a situation where a series of animal silhouettes is available and we wish to compute an average shape:

- Suppose that the database of delineations contains both rodents and batrachians. Figure 2(a) could then be the outline of a rabbit and Figure 2(b) the outline of frog (c). We would probably not like the frog’s eye  $b_2$  to be matched with the rabbit’s ear  $a_2$ . Instead we would like segments  $\{a_1, (b_1, b_2)\}$ ,  $\{(a_2, a_3), b_3\}$  and  $\{a_4, b_4\}$  to be matched, in spite of the fact that the curvature signature of segment  $a_2$  more closely resembles that of  $b_2$  than that of  $b_3$ .
- On the other hand, we may know beforehand that the database contains only rodents and that Figure 2(b) is actually the outline of squirrel (d). Then, we would like the following segment pairs to match:  $\{a_1, b_1\}$ ,  $\{a_2, b_2\}$ ,  $\{a_3, b_3\}$  and  $\{a_4, b_4\}$ .

Clearly, choosing between these two scenarios requires *explicit knowledge* to be introduced into the matching algorithm, as designing a suitable matching algorithm to handle each new case separately would be particularly ineffective.

### 1.4 Learning the correspondence field

To overcome this issue, we propose a learning approach where an *a priori* learning set of correspondence fields helps the matching algorithm compute a correspondence field, between the input curves, whose characteristics resemble those in the learning set. Our method uses a shape distance matrix

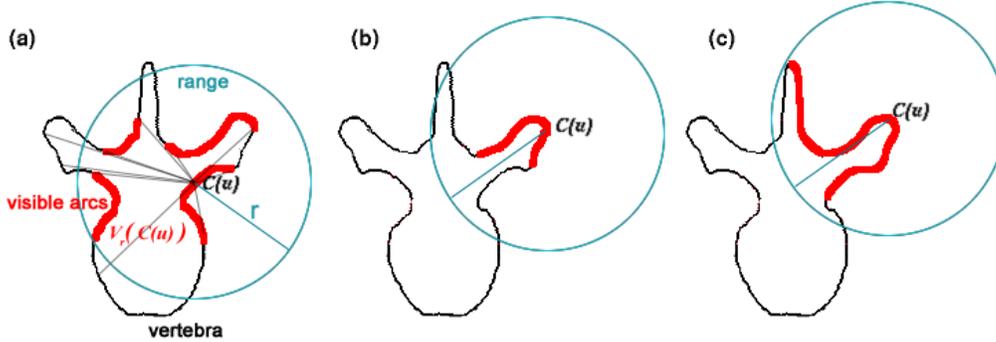


Figure 3: Observed transport descriptor principle demonstrated on a human vertebra outline: the visible arcs (thick red lines) are defined for three locations on  $C(u)$ .

[41, 39], the matrix of the differences between the values of a local shape descriptor computed on every pair of points of the curves to be matched (see Figure 1 for an illustration and Section 3 for a mathematical description). We argue that this shape distance matrix embeds the matching problem in a highly expressive and redundant construct that is easier to manipulate than the reparameterizations of the input curves. This matrix is visually interesting since it allows visual inspection of the reparameterization problem. It also recasts the matching problem as a search for a geodesic in another metrizable space, the space of reparameterizations (which is a group).

We briefly present in the following Section 2 the observed transport descriptor that we previously introduced in [36], and recall the properties that make it suitable for our matching problem. We then elaborate on a classical non-learning technique in Section 3 and introduce our learning-based matching method in Section 4. Results are discussed in Section 5 where we also investigate the robustness of our approach. Note that a preliminary version of this study was published in [36].

## 2 The observed transport shape descriptor

Before computing a shape distance matrix, we first have to select a shape descriptor. Evaluating the quality of a given descriptor is difficult, as it depends both on the characteristics of the shapes to be described and on the specific application. As we could not find a descriptor to suit our needs in the literature and bearing in mind the rodent/batrachian example detailed above, we developed our own: the observed transport descriptor (OT). Here, we formulate it in the continuous 1-D and discrete  $n$ -D cases and illustrate its behavior and notable properties on a few examples. Extensions to higher-dimensional spaces, additional properties, theoretical proofs and accompanying experiments can be found in the thesis [35].

### 2.1 Definitions

- continuous case:

Let  $C : \begin{cases} I \subset \mathbb{R} & \rightarrow \mathbb{R}^n \\ u & \mapsto C(u) \end{cases}$  be a curve in  $n$ -D (open or closed), parameterized with respect to a scalar  $u$ .

We define the observed transport measure  $\rho_r$  as follows:

$$\forall u \in I, \rho_r(C(u)) \doteq \int_{V_r(C(u))} \|C(u) - C(v)\| \cdot |C'(v)| \cdot dv \quad (5)$$

where  $\|C(u) - C(v)\|$  is the Euclidean distance in  $\mathbb{R}^n$ , and  $V_r(C(u))$  is the set of the “visible” arcs

of  $C$ , within distance (or range)  $r \in \mathbb{R}^+$  from  $C(u)$ :  $V_r(C(u)) = \{v \text{ s.t. } \|C(u)C(v)\| \leq r \text{ and line segment } [C(u)C(v)] \text{ intersects } C \text{ only in } C(u) \text{ and } C(v)\}$

By considering only the points *visible* from  $C(u)$ , we obtain a measure whose value changes drastically with the sudden apparition of vastly different surroundings within the range  $r$ , which acts as a locality control (compare Figure 3(a), (b) and (c)). This helps achieve greater discrimination since those points are visually remarkable.

$\rho_r(C(u))$  may be regarded as the minimal total amount of work it takes to transport the elementary elements  $dv$  with mass  $|C'(v)| \cdot dv$  that are visible within range  $r$  from point  $C(u)$ , from their location  $C(v)$ , to  $C(u)$  in the fashion of a Monge-Kantorovich transport problem [18], hence its name.

- discrete approximation on a regular grid:

We define a discrete version of curve  $C$  as an unsorted collection of  $n$ -D isotropic voxels of size  $s^n$ :  $C = \{C_i \in \mathbb{R}^n\}_{i=1}^N$ . In this case, we do not assume any *a priori* parameterization.

We then derive a discrete version of  $\rho_r$  in the  $n$ -D discrete case:

$$\forall i \in 1 \dots N, \rho_r(C_i) \doteq \sum_{j=1, C_j \in V_r(C_i)}^N \|C_i - C_j\| \cdot dV_j(C_i) \quad (6)$$

where  $dV_j(C_i)$  is the surface of voxel  $C_j$  visible from voxel  $C_i$  which we approximate by  $ns^{n-1}$ .

## 2.2 Examples and properties

Figure 4 shows how  $\rho$  behaves with respect to the classical curvature descriptor on a few 2-D curves. All values were recorded with a range  $r = +\infty$ . Both descriptor values for the hand outlines in (b) and (d) were saturated at a fifth of their respective maximal value to better illustrate the discriminating power of the observed transport approach. We recall here the main properties of our descriptor (see [35] for details):

**Large scope:** The OT descriptor has a fairly large scope of application since it can be computed on both connected and disconnected  $n$ -D manifolds and  $n$ -D sets of voxels or points. Note that for disconnected manifolds, a parameterized representation of the object might be difficult to obtain.

**Increased discriminating power:** As illustrated in curve (a), even though  $\rho$  evidently depends on the curvature at the point at which it is computed, it actually takes into account a much larger neighborhood around it. We then get  $\rho(A) \neq \rho(B)$  and  $\rho(C) \neq \rho(D)$ , which correctly reflects the differences in the shape landscape surrounding those points. Note that, of course,  $A$  &  $B$  and  $C$  &  $D$  have, respectively, the same curvature.

Curves (a) and (b) also demonstrate that high curvature points are adequately discriminated. The intersection points of the circle and the straight line in curve (a) have distinctive values, the tips and junctions of the fingers in curve (b) and (d) are clearly distinguished, etc. These matter since they are significant for visual perception (see the psychological experiments of Attneave and Arnoult [2, 3]).

**Adjustable locality:** The observed transport descriptor is both a local measure of shape and an indicator of *context* (for large values of  $r$ ): for instance, it adequately discriminates between the inferior and superior aspects of the corpus callosum in Figure 4(c). A local shape modification therefore only affects its immediate neighborhood, as illustrated with curve (d): the cut middle finger mostly modifies the descriptor values of the immediately surrounding fingers, slightly alters its values for points directly opposite on the other side of the hand (since the cut finger was visible from them), and, evidently, does not affect the left-most and right-most fingers.

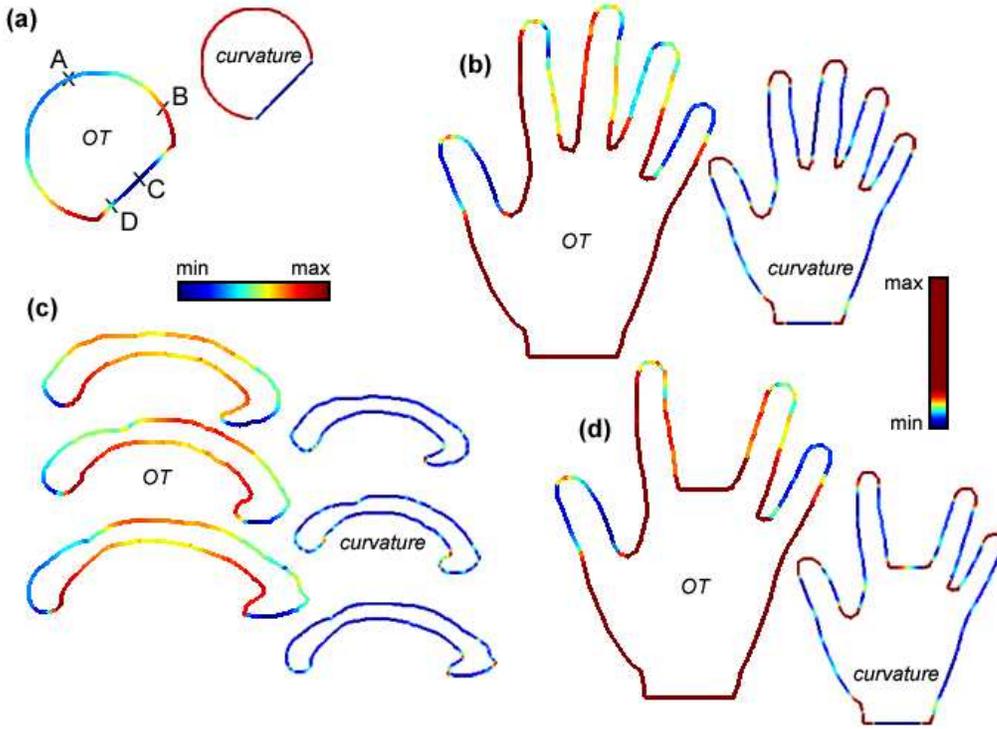


Figure 4: Observed transport (OT) and curvature descriptors computed over a variety of curves: (a) truncated circle (u-parameterized 2-D curve); (b) 5-fingered hand (set of 2-D points); (c) 3 corpora callosa (set of 2-D points) and (d) hand with missing finger (set of 2-D points).

**Invariance:** Our descriptor is also invariant with respect to rigid transformations since both the visible sets and the distances between visible points are invariant with respect to rigid transformations. OT is not scale invariant *per se* as we believe that scale is an important shape characteristic. Scale invariance is however desirable in a curve reparameterization approach (see Section 3 for an implementation). Finally, the use of arc length  $|C'(v)| \cdot dv = ds$  ensures that the descriptor is invariant under reparameterization.

### 3 A non-learning reparameterization approach

Equipped with a suitable shape descriptor, we can tackle the correspondence problem. We first recall, and elaborate on, a standard non-learning algorithm for curve matching [41, 39]. We will detail in Section 4 how a learning strategy can be derived from it. Note that all the methods we discuss could equally be applied to a different shape descriptor (curvature, for instance). The enhanced discriminating power of the observed transport descriptor makes it a more appropriate candidate however.

We consider curves defined on an interval of  $\mathbb{R}$ , taking values in  $\mathbb{R}^n$  with  $n = 2$  or  $3$  (i.e. planar or space curves). Following Trouvé *et al.* [41] (see also Sebastian *et al.* [39] or Gdalyahu *et al.* [13]) we define the best reparameterization  $\varphi_{C_1 \rightarrow C_2}^*$  between curves  $C_1$  and  $C_2$  to be that which minimizes the overall cumulative distance between descriptor values computed for all pairs of matched points (see Figure 1):

$$\varphi_{C_1 \rightarrow C_2}^* \doteq \arg \min_{\varphi=(\varphi_1, \varphi_2)} \left\{ \int_I |\rho_r(C_1(\varphi_1(s))) - \rho_r(C_2(\varphi_2(s)))| \cdot ds \right\} \quad (7)$$

where  $ds$  is the arc length.

In practice, we often need to discretize both curves to compute  $\varphi_{C_1 \rightarrow C_2}^*$ . This sampling process must be performed carefully so as not to discard from the discrete set of pixels important curve features. In particular, one should pay attention to the sampling rate required to capture the curve details deemed important.

Let  $\Lambda$  be the shape distance matrix associated with the discretized curves  $C_1 = \{C_1^i\}_{i=1}^{N_1}$  and  $C_2 = \{C_2^j\}_{j=1}^{N_2}$  where  $N_1$  is the number of points in the discrete  $C_1$  and  $N_2$  the number of points in  $C_2$ :

$$\Lambda = [\lambda_{ij}] \quad \begin{matrix} i = 1 \dots N_1 \\ j = 1 \dots N_2 \end{matrix}, \quad \forall (i, j) \quad \lambda_{ij} \doteq \left| \rho_r(C_1^i) - \rho_r(C_2^j) \right| \quad (8)$$

Finding the best reparameterization then boils down to finding in  $\Lambda$  the minimal cost path between points  $(0, 0)$  and  $(N_1, N_2)$ , which requires that a single matching pair of points  $(M_1 \in C_1, M_2 \in C_2)$  be given. This can be done with a dynamic programming approach [41, 39].

Note that the discrete  $C_1$  and  $C_2$  may sometimes have very different sizes. For instance, they may have been sampled at the same rate from images with different resolutions. In this cases, non scale invariant descriptors will take vastly different ranges of values. This may hamper the computation of an adequate correspondence field. One way to alleviate this issue is to replace  $\Lambda$  by a normalized version,  $\tilde{\Lambda}$ , computed as follows:

$$\forall (i, j) \quad \tilde{\lambda}_{ij} \doteq \left| \tilde{\rho}_r(C_1^i) - \tilde{\rho}_r(C_2^j) \right| \quad \text{with} \quad \tilde{\rho}_r(C_k^l) \doteq \frac{\rho_r(C_k^l) - \mu_{C_k}}{\sigma_{C_k}} \quad (9)$$

where  $\mu_{C_k}$  and  $\sigma_{C_k}$  are respectively the mean and the standard deviation of  $\{\rho_r(C_k^l)\}_{l=1}^{N_k}$ .

### 3.1 Choosing $M_1$ and $M_2$

The matching pair of points can be chosen *a priori* or they may be optimized as well. For open curves, a natural choice would be to select one of the corresponding extremities. The dynamic programming approach then yields an  $\mathcal{O}(N_1.N_2)$  complexity. For closed curves, a pair of remarkable homologous points (one in each curve) could be selected based on their particular characteristics: extrema of curvature, left-most/right-most points, etc. Complexity is also  $\mathcal{O}(N_1.N_2)$ . In the general case where no pair of matching points is given, the complexity becomes  $\mathcal{O}(N_1.N_2^2)$  as we apply the  $\mathcal{O}(N_1.N_2)$  dynamic programming algorithm to all possible pairs  $(M_1 \in C_1, M)$ ,  $\forall M \in C_2$  and select the matching with lowest cost.

Much better control over this selection process can be obtained with the learning approach we introduce below (see Section 4).

### 3.2 Consistent self-matching

When a number of consecutive points have the same shape descriptor (in a circle for instance, for both OT and curvature), there is no unique best path with respect to the above criterion. To bias the search towards “natural” reparameterizations (the “consistent self-matching” axiom), we introduce in equation 7 a constraint to prevent the path from deviating too much from the diagonal of  $\Lambda$ , *i.e.* for some  $\alpha \in \mathbb{R}$ :

$$\varphi_{C_1 \rightarrow C_2}^* \doteq \arg \min_{\varphi} \left\{ \int_I |\tilde{\rho}_r(C_1(\varphi_1(s))) - \tilde{\rho}_r(C_2(\varphi_2(s)))| .ds + \alpha . \int_I |\varphi_1(s) . C_2'(s) - \varphi_2(s) . C_1'(s)| .ds \right\} \quad (10)$$

The larger the  $\alpha$ , the closer the path is to the diagonal. A circle matched against itself then yields  $\phi_1 = \phi_2 = Id$  when  $\alpha > 0$ . Consistent self-matching is particularly desirable for shape descriptors with only average discriminating power, for which large parts of the input objects map to the same descriptor value. It also proves useful as a regularization term when the chosen shape descriptor is noisy. Clearly, there are only a handful of different curvature values for the rectangles of Figure 5 (g) and (h). The associated shape distance matrix then presents large uniformly colored squares within which all paths would be equiprobable if  $\alpha$  were zero. In practice, we take  $\alpha = 0$  for the OT descriptor as it is sufficiently discriminating in and of itself.

### 3.3 A few minimal cost path examples

Figure 5 illustrates the non-learning approach on four pairs of curves (one pair per column), for the observed transport and the curvature descriptors. For each pair, we show the OT maps in (a,b) and the curvature maps in (g,h), along with the point-by-point Euclidean averages derived from them (f,i), and the shape distance matrix associated to the OT descriptor (c). We used  $\alpha = 0$ , and  $r = +\infty$  to yield maximum discriminating power. Both OT and the curvature descriptor were normalized with the mean/standard deviation technique described above. Color is used both to display the values of the descriptors in (a),(b),(g) and (h), and to render corresponding points in (d) and (e). Overall, the OT descriptor behaves more adequately than curvature on the synthetic and real examples.

## 4 A learning reparameterization approach

We present in this section the learning algorithms we developed to bias the search for a correspondence field between two  $n$ -D curves towards instances that are admissible with respect to an *a priori* given learning set. It is through this learning set that we introduce *a priori* knowledge and expertise in the matching process.

### 4.1 Method

An interesting feature of the shape distance matrix is that it embeds, in a highly redundant way, information about all possible reparameterizations between the two input objects. In Figure 8 for instance, we can notice clear patterns corresponding to the triangular indentations of the rectangles (highlighted areas). A local matching scenario, e.g. “matching the rodents’ ears” in Figure 2, or “matching the triangles together” in Figure 5, then corresponds to a sub-path in a sub-matrix extracted from the shape distance matrix of the objects.

We derive the following 4-step algorithm (see Figure 6):

**Step 1 (*a priori*).** Given a number of desired local matching scenarios, a human operator first forms a learning set by selecting instances (i.e. examples) for each scenario. An instance consists of a 2-D sub-matrix  $M_{i,j} \in \mathcal{M}_{m_{i,j} \times n_{i,j}}$ , and its associated connected sub-path  $P_{i,j} = \left\{ \left( x_{i,j}^k, y_{i,j}^k \right) \right\}_{k=1}^{m_{i,j} + n_{i,j}}$ , where  $i$  is the scenario index, and  $j$  is the instance index.

Sub-paths may be defined in many ways. For instance, if the operator is satisfied with the characteristics of a match obtained between two curves with the non-learning matching approach, and wants to enforce similar characteristics over a series of similar curves, he can extract sub-paths from the complete minimum cost path computed via dynamic programming between these two curves. Alternatively, sub-paths can be drawn by hand on the sub-matrices (this is in essence what we did in Figure 8) or generated algorithmically.

Similarly, sub-matrices are extracted from shape distance matrices computed from curves that should look similar to the ones the operator wants to reparameterize. As always with a learning approach, the operator should ensure that the learning set adequately represents the desired matching characteristics by selecting appropriate instances.

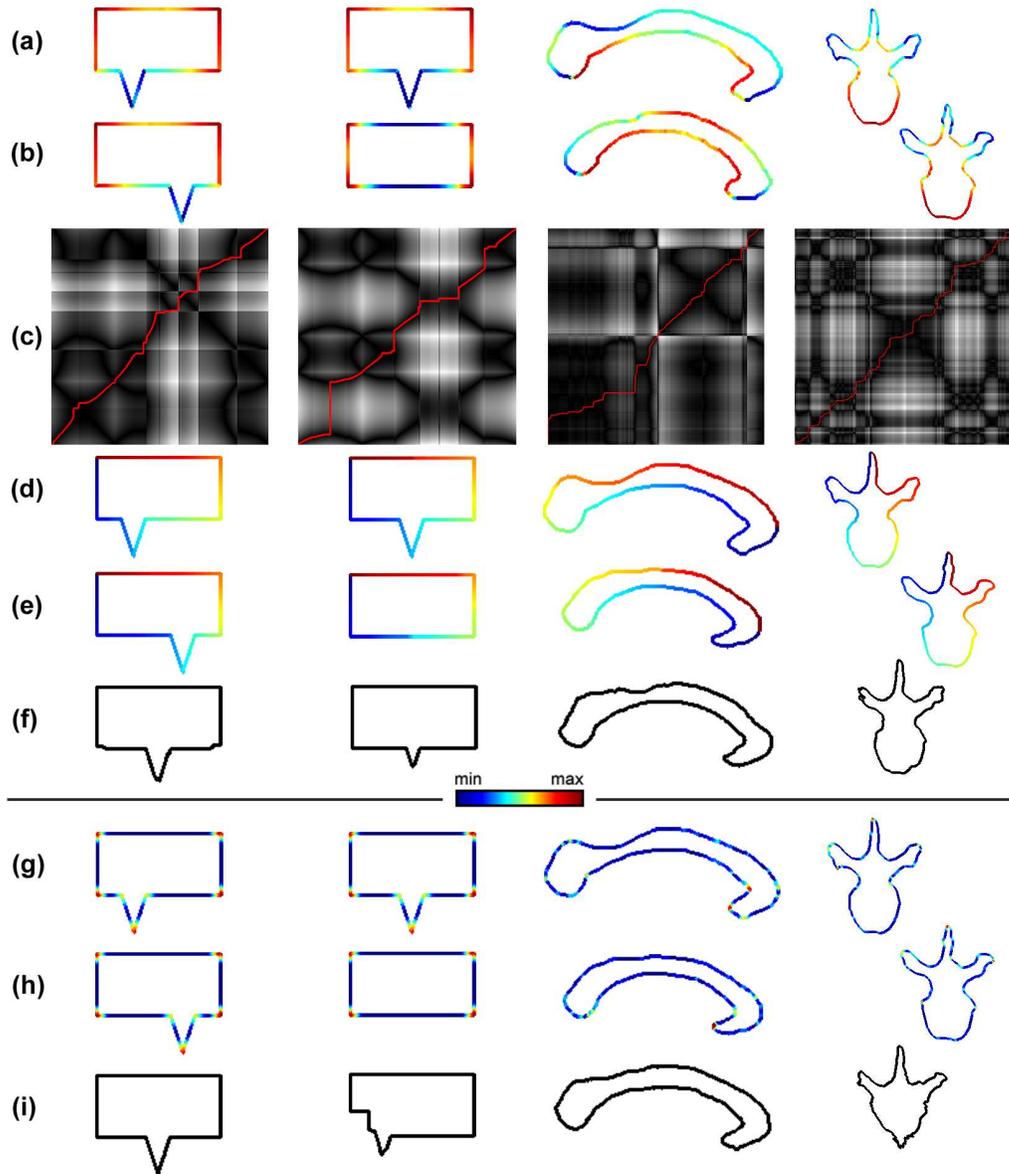


Figure 5: Non-learning reparameterization: (a,b) Observed Transport maps; (c) associated shape distance matrices (scaled to a square) and optimal paths (in red); (d,e) reparameterized curves with OT (corresponding points are rendered in the same color); (f) point-by-point Euclidean averages for OT; (g,h) curvature maps; (i) point-by-point Euclidean averages for curvature.

$P_{i,j}$  is then the sub-path in  $M_{i,j}$  which represents a local matching scenario, in the same fashion that the optimal cost path in Section 3 represents the “optimal” global matching scenario.

Figure 6 illustrates that process. We want to reparameterize a series of corpora callosa (the corpus callosum is a C-shaped structure of the brain which contains nerve fibres that connect the hemispheres). Two representative pairs of callosa are first selected (A,B). Note that we could have selected only one pair, or more than two pairs, depending on the heterogeneity of the dataset. The shape distance matrix of each pair is computed (C,D). Then, the operator identifies in the shape matrices the sub-matrices  $M_{i,j}$  that correspond to the matching scenarios of interest (for instance, matching the posterior part of the callosa and matching the anterior part) to form the learning set (E). In this example, we have one instance,  $S_{1,1}$ , for the first matching scenario and two instances,  $S_{2,1}$  and  $S_{2,2}$ , for the second scenario. For each instance  $S_{i,j}$ , we also compute the distance map  $D_{i,j}$  of its sub-path inside the domain defined by its sub-matrix ( Figure 6(F) ).

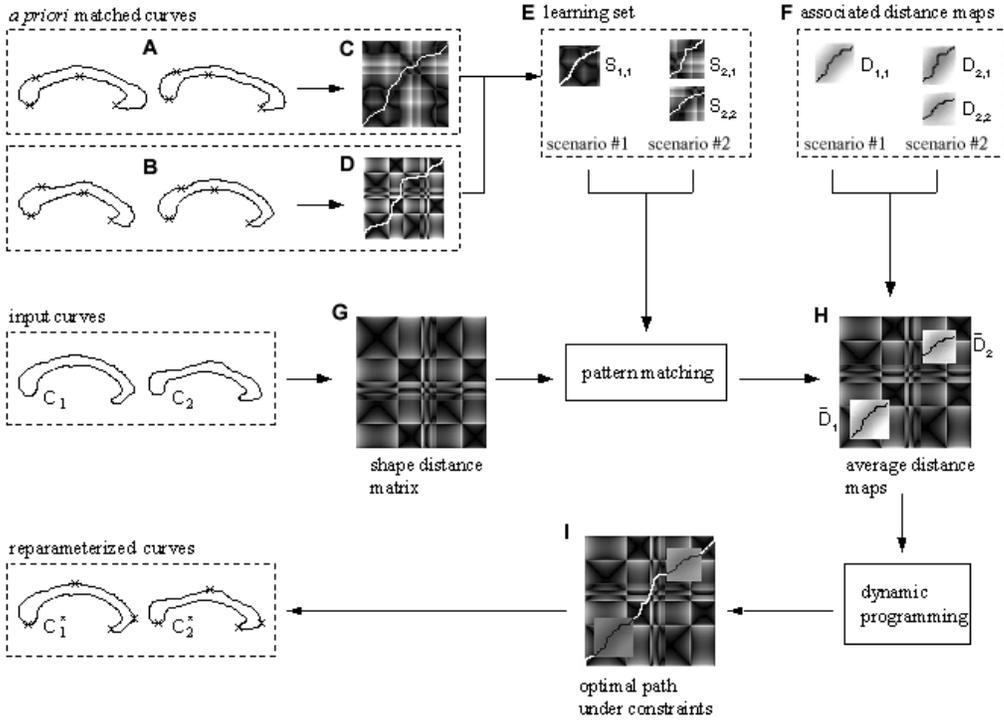


Figure 6: Pattern matching strategy.

Let  $S_1 = \{S_{1,1}, \dots, S_{1,N_1}\}, \dots, S_K = \{S_{K,1}, \dots, S_{K,N_K}\}$  be the  $K$  scenarios, with their instances  $S_{i,j} = (M_{i,j}, D_{i,j})$ .

**Step 2.** Once we have computed the shape distance matrix  $M \in \mathcal{M}_{m,n}$  from the two input curves to be reparameterized  $C_1$  and  $C_2$  ( Figure 6(G) ), a pattern matching algorithm is used to find sub-matrices in  $M$  that closely resemble those of the learning set. We use a straightforward multi-scale registration approach: for each sub-matrix  $M_{i,j}$  in the learning set, we compute the similarity between its translated and scaled image and the underlying sub-matrix in  $M$  of same dimensions. The range of translations and scales explored by the registration algorithm defines an “exploration neighborhood” in the shape distance matrix whose size can be controlled by the user. To account for a possible mismatch between  $M_1$  and  $M_2$ , we consider  $M$  to be closed under vertical and horizontal circular shifts. That is, when the translated and scaled image of  $M_{i,j}$  is not entirely included in  $M$ , missing values are taken from the corresponding other side of the matrix  $M$ .

For each  $M_{i,j}$ , we then record the translation  $t_{i,j}^*$  and scale  $s_{i,j}^*$  for which the maximal similarity is achieved:

$$(t_{i,j}^*, s_{i,j}^*) = \arg \max_{t,s} (\text{similarity}(T(t,s)M_{i,j}, M)) \quad (11)$$

where  $T(t,s)M_{i,j}$  is the image of  $M_{i,j}$  translated by vector  $t$  and scaled by factor  $s$ . We also discard instances for which the associated similarity measure is too low (that is, below an application-dependent threshold  $\theta$ , typically 0.2 for the correlation coefficient).

**Step 3.** For each scenario  $S_i$  in the learning set, we then apply translation  $t_{i,j}^*$  and scale  $s_{i,j}^*$  to the distance maps  $D_{i,j}$  associated with each of its instances and average them, to allow for every instance of the scenario to equally influence the matching ( Figure 6(H) ). Let  $\bar{D}_i$  be the average distance map associated with scenario  $S_i$ . The averaging process is done pixel by pixel:

$$\forall(x,y), \bar{D}_i(x,y) = \frac{1}{N_i} \sum_{j=1}^{N_i} T(t_{i,j}^*, s_{i,j}^*)D_{i,j}(x,y) \quad (12)$$

with the convention that  $D_{i,j}(x, y) = 0$  outside of the domain defined by the sub-matrix  $M_{i,j}$  inside which the sub-path  $P_{i,j}$  is inscribed.

For example in Figure 6, we average the scaled and translated maps  $T(t_{2,1}^*, s_{2,1})D_{2,1}$  and  $T(t_{2,2}^*, s_{2,2})D_{2,2}$  of the two instances  $S_{2,1}$  and  $S_{2,2}$  of scenario  $S_2$ ; no averaging is required for scenario  $S_1$  since it only has one instance,  $S_{1,1}$ . The zero-level sets of the average transformed distance maps [27] then give the average transformed sub-paths,  $\bar{P}_i$ .

Note that the quality of the match between the sub-matrices from the learning set and the matrix  $M$  could be used to compute a weighted average distance map instead of an equal-weight one.

Also, there is no theoretical guarantee that the zero-level set of an average map is a single open curve, as averages of distance maps are not distance maps. However in practice, it is highly unlikely that a topologically different path would be obtained as the sub-paths being averaged are fairly similar to each other.

**Step 4.** We then use dynamic programming to find the optimal sub-paths in the shape distance matrix  $M$  *in between* the average sub-paths,  $\bar{P}_i$ , computed at step #3 (i.e., those imposed by the scenarios).

Alternatively, we could simply *bias* the search for the global optimal path towards those defined by the average sub-paths instead of forcing it to pass through them. We would then replace  $\Lambda$  by  $\bar{\Lambda}$ , the combination of the shape distance matrix  $\Lambda$  with each average distance map  $\bar{D}_i$  defined as follows:

$$\forall(x, y), \bar{\Lambda}(x, y) = \Lambda(x, y) + \sum_{i=1}^K (\bar{D}_i(x, y) + (1 - \chi_{\bar{D}_i}(x, y)) \cdot \alpha_i) \quad (13)$$

where  $\alpha_i$  is the minimum value of  $\bar{D}_i$  and  $\chi_{\bar{D}_i}$  is the indicator function such that:  $\chi_{\bar{D}_i}(x, y) =$

1 if $(x, y) \in$ domain of $\bar{D}_i$
0 otherwise

Adding the minimal value of the average distance map to all the other pixels in the shape distance matrix prevents the dynamic programming algorithm from systematically avoiding the sub-matrices around the sub-paths, as each of the sub-matrices' pixels would otherwise add the value of the distance map to the shape distance matrix of the input curve underneath. Here again, a weighted sum could be considered to favor the sub-paths associated to the best matching sub-matrices.

Even with a shape descriptor invariant under reparameterizations, pairs of curves with substantially different initial parameterizations will produce different looking shape distance matrices (for instance, if one of the curves were parameterized in the reverse direction). Care should thus be taken to use similar *a priori* parameterizations for the objects to be matched and the ones in the learning set (arc length for instance, if it can be computed, or similarly ordered sets of pixels sampled in a similar fashion). This initial parameterization issue is somewhat typical of learning approaches. Clearly, there must be a sufficient degree of similarity between the items of the learning set and those onto which a previously learnt procedure has to be applied.

## 4.2 Determining $M_1$ and $M_2$

As mentioned above, the dynamic programming framework requires that a pair of matching points,  $M_1$  and  $M_2$ , be chosen *a priori*. Our matching approach provides an alternative means to select those points, with the proviso that they exhibit particular characteristics. Namely, we can extract from the shape distance matrix a sub-matrix that contains the matched  $(M_1, M_2)$  and use pattern matching to find its translated/scaled image in the shape distance matrix of the pair of input curves

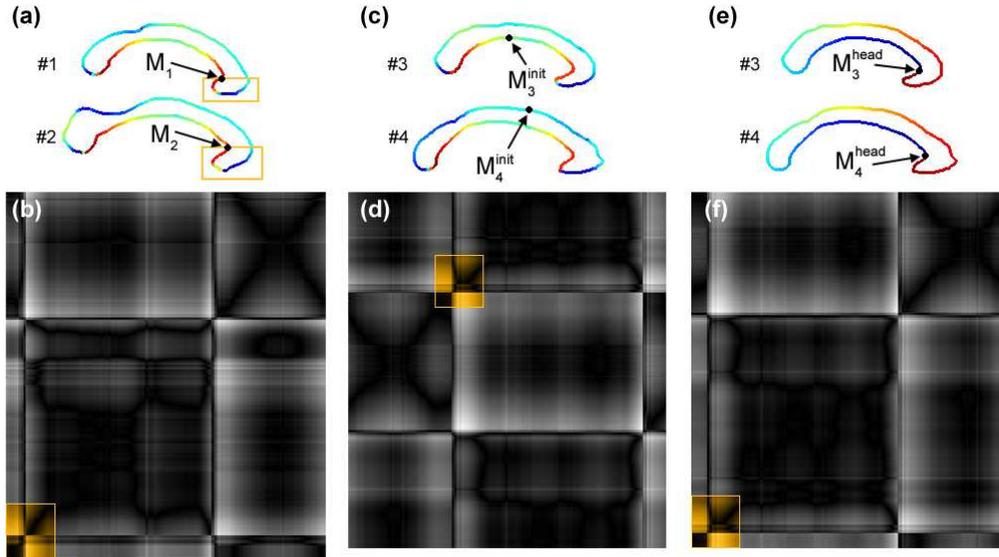


Figure 7: Determining the matching pairs of points: (a) selected representative outlines (color is used to render OT values) with callosal head highlighted in yellow and matching pair ( $M_1, M_2$ ); (b) corresponding shape distance matrix with highlighted head sub-matrix; (c) another pair of callosal outlines with random initial matching pair ( $M_3^{init}, M_4^{init}$ ); (d) associated shape distance matrix with highlighted matched sub-matrix; (e) resampled outlines (corresponding points are rendered in the same color) with new matching points *after* pattern matching; (f) associated shape distance matrix.

to be reparameterized. As the shape distance matrix is redundant, this provides us with a selection technique more robust than those using maxima of curvature or position, for instance.

We applied this technique to a series of corpus callosum outlines (#1 and #2 in Figure 7(a), #3 and #4 in (b), the remaining callosa are displayed in Figure 11(c)). For each pair of outlines, we are looking for an adequate matching pair of points. The user starts by choosing a pair of representatives outlines (#1 and #2) and determines a pair of matching point on them (a). We then compute the shape distance matrix (b). The user then determines in this matrix the sub-matrix corresponding to the head of the corpus callosum (highlighted in yellow). Given a new pair of outlines (c), we select  $M_3^{init}$  and  $M_4^{init}$  at random and compute the shape distance matrix (d). The pattern matching approach detailed in Section 4.1 is then used to find the callosal head sub-matrix from which we extract a proper pair of matched points ( $M_3^{head}, M_4^{head}$ ) in (e). We used the correlation coefficient as a similarity measure and  $\theta = 0.2$ .

The new shape distance matrix (f) is the image of the matrix (d) by the combination of a horizontal and a vertical circular shift (hence the need to take these into account in step 2 of the learning approach).

## 5 Results

We present here some synthetic examples as well as biomedical ones (our initial motivation) before commenting on the robustness of the learning approach.

### 5.1 Matching rectangles and triangles

Figure 8 illustrates our approach on the familiar rectangles with sliding triangular indentations. In the first case/scenario (first row), we wanted to match the triangular indentations together, whereas in the second case (second row), we wished to discard them as noise, and match them against the directly corresponding rectangle pieces. The learning set consisted in each case of the same sub-matrix taken from shape distance matrix (b) but with a different sub-path. The range of scales explored

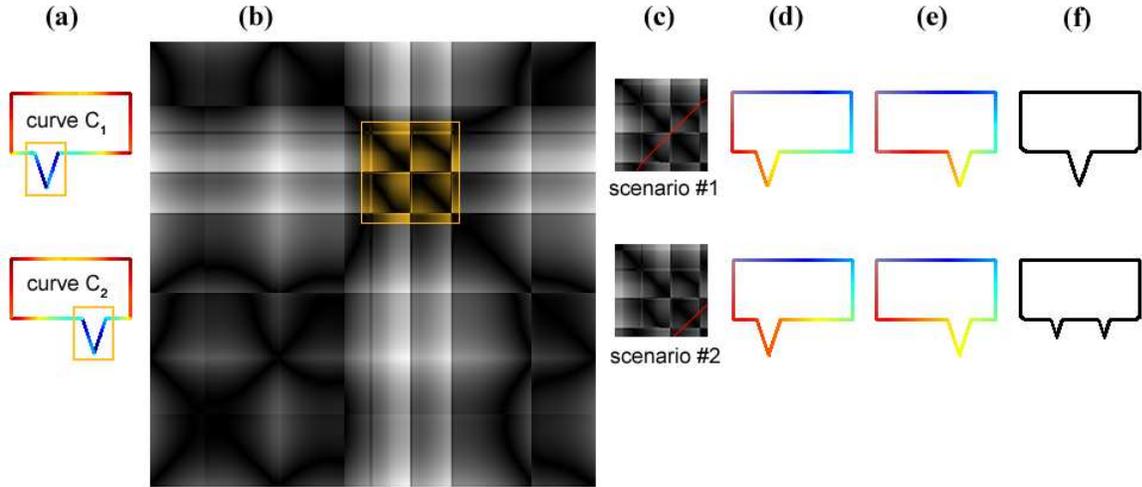


Figure 8: Learning reparameterization: (a) pair of curves used to build the learning set (color is used to render OT values); (b) shape distance matrix with highlighted learning sub-matrix; (c) learning sub-matrix with sub-paths for two matching scenarios: matching the triangular indentations (top row), discarding triangles as noise (bottom row); (d and e) input curves with color used to render corresponding points after the reparameterization; (f) the resulting point-by-point Euclidean average curves.

by the matching algorithm was  $\{0.5, 1.0, 1.5\}$ . We used the correlation coefficient as a similarity measure and  $\theta = 0.2$ . Figure 8(f) shows that the point-to-point Euclidean averages obtained for each scenario conformed to the expected matching characteristics: we obtained a triangle positioned halfway between the ones in the input curves on the first row (reflecting the fact that the triangles were matched together, as demonstrated by the color matches in (d) and (e)), and two attenuated triangles on the second row.

Note that the results of the first scenario exhibit characteristics similar to those of the Euclidean average obtained with the classical non-learning approach (see left-most Euclidean average in Figure 5(f)) with the notable difference that the average triangular indentation is better defined with the learning approach. This was to be expected since the learning approach ensured that the tips of the triangular indentations in the input curves were in exact correspondence whereas the non-learning technique can only aim for it.

The same learning sets were used to compute the mean shape of the set of noisy irregular rectangles with protruding indentations in Figure 9(a). The similarity measure threshold was set slightly lower to 0.1 as the input curves were somewhat perturbed. Notice how the indentation was indeed perceived as an important feature with the first learning set in (b) or discarded as noise in (c).

## 5.2 Preventing matches

Our learning approach to reparameterization can also be used to prevent sub-matches. In this case, we give to all points in the sub-matrix found by the pattern matching technique of step 2 an arbitrarily high value: the dynamic programming algorithm will then avoid this area in search for the overall minimum cost path. This could be helpful to prevent the matching of an animal’s eyes with another’s ears in Section 1.3, for instance, when we know that there are mistakes in the database.

This technique was used in Figure 10 to prevent the fishes’ fins from being matched. We selected among the shape distance matrices computed between 20 fish delineations (6 of them are shown in (a)) three sub-matrices corresponding to the fin-to-fin matching areas. While the classical non-learning technique fairly adequately matched the scales in (b), our learning algorithm successfully discarded them as a non-preminent feature in (c). The strain imposed on the dynamic programming algorithm by these avoidance constraints unfortunately prevented it from correctly matching the fish tails. An

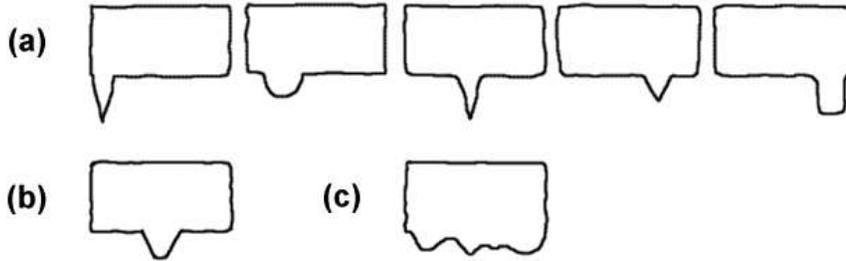


Figure 9: Matching irregular indentations: (a) set of irregular curves with a sliding indentation; (b) learning based Euclidean point-by-point average with scenario #1's learning set; (c) learning based average with scenario #2's learning set.

improved average outline could be obtained by introducing the fish tail area into the learning set, with the appropriate sub-path.

### 5.3 Biomedical example.

Figure 11 demonstrates the behavior of our matching approach on a series of corpus callosum delineations. Two pairs of corpus callosum outlines were *a priori* manually reparameterized (a): a number of pairs of corresponding points were manually selected, and a continuous correspondence field was interpolated between them. We computed their associated shape distance matrix (b) and selected in it a number of scenarios (yellow highlights) to serve as a learning set. We applied our learning approach to a test set of 20 callosa manually delineated by an expert in T1-weighted MR volumes. The matching pair of points were obtained using the pattern matching strategy described in Section 4.2. Here also, the range of scales explored by the matching algorithm was  $\{0.5, 1.0, 1.5\}$ . We used the correlation coefficient as a similarity measure and  $\theta = 0.2$ . Figure 11(c) shows 5 out of the 20 reparameterized corpora callosa, and the overall callosum average in (d), computed as the point-by-point Euclidean mean of the reparameterized callosa. Clearly, both the head and the tail of the callosum were correctly matched across the set of instances. This outline compares favorably against the Euclidean mean computed without learning (e) for which, as could be expected, the fairly irregular shape of the callosal head fooled the dynamic programming search.

### 5.4 Robustness of the matching approach

A classical dilemma of learning approaches, shared by ours, is whether the learning set is representative. Namely, a trade-off must be found between too exhaustive a learning set that might induce overfitting and too small a learning set which will not represent the target shapes sufficiently well, both of them yielding poor generalizability. Representativity aside, the robustness of our method mostly depends on that of the pattern matching algorithm used in step 2 which in turn is determined by the characteristics

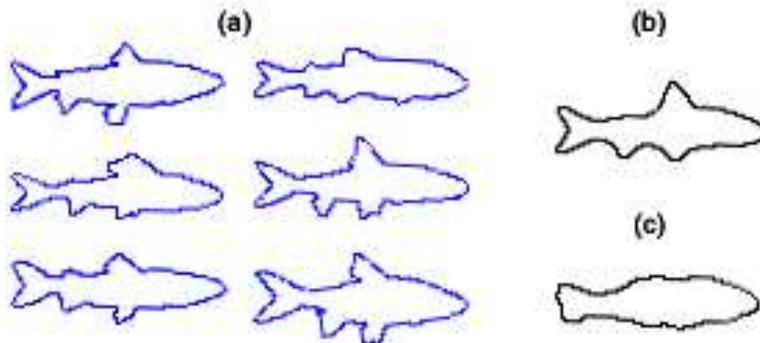


Figure 10: Preventing matches: (a) 6 out of 20 fish delineations; (b) non-learning Euclidean point-by-point average; (c) learning based average where fins were discarded as noise and prevented from matching.

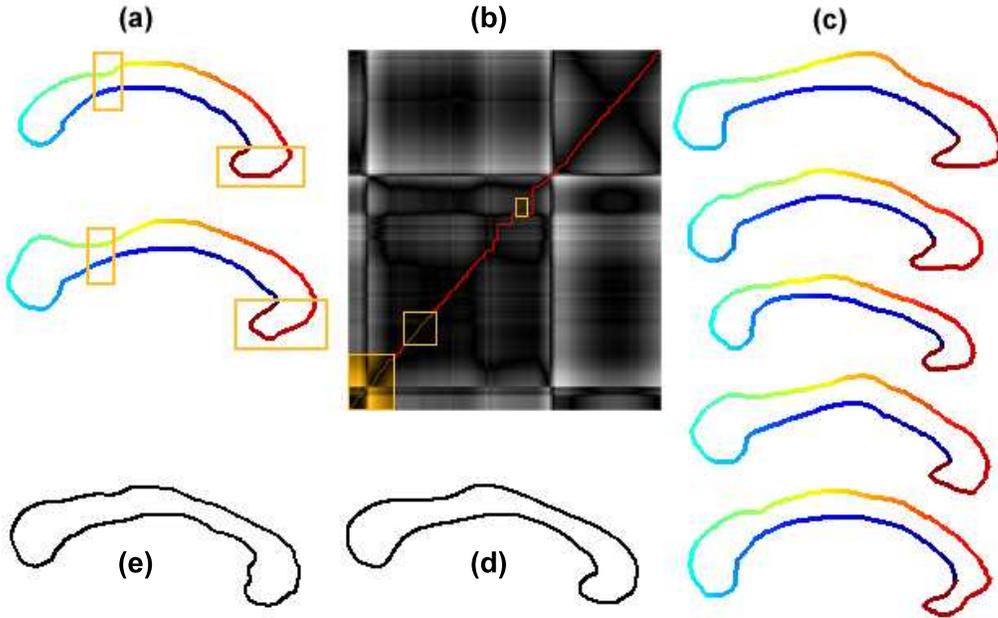


Figure 11: Pattern matching reparameterization of a few corpus callosum: (a) Two pairs of *a priori* parameterized callosum (corresponding points are rendered in the same color); (b) their associated shape distance matrix with the corresponding path shown in red and the selected scenario/shape distance sub-matrices in yellow; (c) a few reparameterized corpus callosum; (d) the overall point-by-point Euclidean average callosum with learning; (e) Euclidean average without learning.

of the chosen shape descriptor, the similarity measure used, and the neighborhood explored.

#### 5.4.1 Similarity measure

Since we are matching shape distance sub-matrices against each other, a similarity measure designed for mono-modal registration should be used. Here, we selected the correlation coefficient, which assumes an affine relationship between the intensities of both sub-matrices. When a greater dissimilarity is expected (that is, when the samples from the learning set might be fairly different from the objects to be matched), more general purpose measures could be used, the correlation ratio or mutual information for instance. These could only work properly on sufficiently large sub-matrices as they rely on the estimation of probability densities.

To increase robustness, we discard the poor matches between the sub-matrices in the learning set and the shape distance matrix. No match actually had to be discarded in the experiments reported in this article, though.

#### 5.4.2 Exploration neighborhood

The size of the exploration neighborhood depends on the differences between the *global* shapes of the objects to be matched and those of the *a priori* matched objects in the learning set:

- when they differ substantially, the exploration neighborhood must be relatively large (it could even be extended to the entire shape distance matrix) as the sub-matrix in the shape distance matrix of the objects to be matched might occupy a completely different position from that of the corresponding sub-matrix in the learning set. Adding noise to one curve (see Section 5.4.4) usually also implies increasing the size of the exploration neighborhood.
- when the objects are globally similar in shape, the shape distance matrices must also look alike: the exploration neighborhood can consequently be reduced as the corresponding sub-matrices should occupy close locations.

$\mu$	<i>distance (pixel)</i>	triangular indentation	all other points
0	mean	2	3
	max	3	10
2	mean	2.7	4
	max	4	5
4	mean	3	5
	max	3	7
6	mean	3.7	9
	max	5	14
8	mean	4.3	14
	max	6	15
10	mean	6.7	16
	max	9	23
12	mean	10	19
	max	15	26

Table 1: Noise sensitivity of the matching algorithm.

### 5.4.3 Scale

The range of scales explored also depends on the expected differences between the learning set and the objects to be matched:

- if the sizes between these are expected to be about the same (as with anatomical structures, for instance) then a small range of scales should be used; this should only be large enough to allow for variations caused by noise;
- when the sizes are expected to be more different, then obviously a larger range should be used. The use of a normalized share distance matrix is then highly recommended.

### 5.4.4 Noise

An experiment was designed to assess the sensitivity of the matching algorithm to small variations of the input curves. The objects considered were again the pair of rectangles with a sliding triangular indentation. 14 “virtual” landmark points were chosen along each curve: one at each corner and one half-way between every corner. These landmarks were not attached to a point on the curves - they only represent areas of interest in the curve. Curve 2 was then perturbed by a uniform noise of varying amplitude  $\mu$  (we added the noise, variance  $\sigma^2 = \mu^2/3$ , to each point of the delineation). The curves were subsequently matched with the learning set of the first row of Figure 8 (“matching triangle” scenario).

Table 1 reports the average and maximal Euclidean distance between the landmarks of curve 2 and the points of the perturbed curve 2 associated after matching to the corresponding landmark in curve 1 for various values of  $\mu$ . We made a distinction between points which belonged to the triangle from the others as their matching was supposed to be enforced by our algorithm. We used the correlation coefficient as a similarity measure and  $\theta = 0.2$ . The range of scales explored by the matching algorithm was  $\{0.5, 1.0, 1.5\}$  and we considered as an exploration neighborhood a rectangular window twice the size of the considered sub-matrix, centered on it.

As expected, the matching error was lower in the triangular indentation and, more importantly, stayed relatively constant until  $\mu = 8$ , whereas the error continuously increased along with  $\mu$  for points where the matching was not enforced via the learning set.

When  $\mu$  became too high, the matching algorithm could not manage to find in the shape distance matrix a sub-matrix similar enough to the one in the learning set.

## 5.5 Building the learning set

Our learning method requires that the correspondences between the objects of the learning set be established *a priori*. This may not always be a trivial task for complex shapes. However, it only has to be specified once and for a small number of instances. Also a sparse subset of the correspondence field could be specified by the user to generate a learning set. Most of the fully automated techniques presented in the introduction could produce a meaningful set that could then be manually corrected with a simple interface if need be.

Note that using a learning set implies that the objects we want to reparameterize should not be too different from those in the learning set. In fact, similarities between objects do not matter so much as similarities between the pairs of objects to be reparameterized and the pairs of objects in the learning set. Of course, the former is a sufficient condition for the latter. However, a unique advantage of our approach is that it can learn a matching strategy for even very dissimilar objects (e.g., across species in comparative allometry applications), provided that we apply it to the same dissimilar matching situations.

As such, this technique could prove particularly useful to put into correspondence the “odd looking” instances of an object set as they might require more attention than the “ordinary” ones which could be treated with direct non-learning algorithms.

## 6 Conclusion

A careful analysis of the object matching problem motivated the need for *explicit control* over the reparameterization process. We consequently developed a learning approach to the matching problem, based upon a classical shape difference geodesic search technique [41, 39]. A pattern matching algorithm drives the matching algorithm towards a correspondence field whose characteristics closely resemble those in a learning set of *a priori* corresponding instances. This allows the user to exert explicit control over the reparameterization.

We chose as a shape descriptor the previously introduced observed transport descriptor, in view of its increased discriminating power.

We have demonstrated the adequate behavior and flexibility of this approach on a number of synthetic and real examples. We also illustrated its inherent simplicity. Given a matching scenario, the visual qualities of the shape distance matrix make it straightforward to build a learning set by selecting sub-matrices in it. To satisfactorily handle the same scenario, a classical approach would require the user to actually design a suitable similarity measure and its associated set of constraints.

We are currently investigating the extensions of this technique to (1) higher dimensional spaces to match surfaces and volumes (see [36] for a preliminary sketch) and (2) to the reparameterization of groups of curves for atlas building.

## Acknowledgments

The authors wish to thank the reviewers for their many helpful insights. P.T. was supported by the National Institute on Aging, the National Library of Medicine, the National Institute for Biomedical Imaging and Bioengineering, and the National Center for Research Resources (grants AG016570, LM05639, EB01651, RR019771).

## References

- [1] E. Arkin, P. Chew, D. Huttenlocher, K. Kedem, and J. Mitchel. An efficient computable metric for comparing polygonal shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(3):703–716, 1991.

- [2] F. Attneave. Some informational aspects of visual perception. *Psychological Review*, 61(3):183–193, 1954.
- [3] F. Attneave and M.D. Arnoult. The quantitative study of shape and pattern perception. L. Uhr, editor, *Pattern Recognition*, pages 123–141, 1966.
- [4] S. Belongie, J. Malik, and J. Puzicha. Shape Matching and Object Recognition Using Shape Context. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):509–522, 2002.
- [5] F.L. Bookstein. *Morphometric Tools for Landmark Data: Geometry and Biology*. Cambridge University Press, 1991.
- [6] Haili Chui and Anand Rangarajan. A new algorithm for non-rigid point matching. In *Proceedings of Computer Vision and Pattern Recognition (CVPR’00)*, pages 2044–2052, 2000.
- [7] I. Cohen, N. Ayache, and P. Sulget. Tracking Points on Deformable Objects using Curvature Information. In *Proceedings of European Conference on Computer Vision (ECCV’92)*, pages 458–466, 1992.
- [8] I. Cohen and I. Herlin. Curves Matching Using Geodesic Paths. In *Proceedings of Computer Vision and Pattern Recognition (CVPR’98)*, pages 741–746, 1998.
- [9] T.F. Cootes, G.J. Edwards, and C.J. Taylor. Active Appearance Models. In *Proc. of ECCV*, pages 484–498, 1998.
- [10] C. Davatzikos, J. Prince, and N. Bryan. Image Registration Based on Boundary Mapping. *IEEE Transactions on Medical Imaging*, 15(1):212–215, 1996.
- [11] R.H. Davies, C.J. Twining, T.F. Cootes, J.C. Waterton, and C.J. Taylor. A Minimum Description Length Approach to Statistical Shape Modelling. *IEEE Transactions on Medical Imaging*, 21(5):525–537, 2002.
- [12] M. Fleuté, S. Lavallée, and R. Julliard. Incorporating a Statistically Based Shape Model into a System for Computer-Assisted Anterior Cruciate Ligament Surgery. *Medical Image Analysis*, 3(3):209–222, 1999.
- [13] Y. Gdalyahu and D. Weinshall. Flexible Syntactic Matching of Curves and its Application to Automatic Hierarchical Classification of Silhouettes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(12):1312–1328, 1999.
- [14] E. Goldmeier. Similarity in visually perceived forms. *Psychological Issues*, 8(1), 1972.
- [15] A. Goshtaby. Description and discrimination of planar shapes using shape matrices. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7:738–743, 1985.
- [16] U. Grenander, Y. Chow, and D. Keenan. *HANDS: A Pattern Theoretic Study of Biological Shapes*. Springer, 1991.
- [17] H.W. Hake. From discrimination and the invariance of form. In L. Uhr, editor, *Pattern Recognition: Theory, Experiments, Computer Simulations, and Dynamic Models of Form, Perception, and Discovery*, pages 142–173. 1966.
- [18] S. Haker, S. Angenent, and A. Tannenbaum. Minimizing Flows for the Monge-Kantorovich Problem. *SIAM Journal of Mathematical Analysis*, 35(1):61–97, 2003.
- [19] D.O. Hebb. *The organization of behavior*. John Wiley, 1949.

- [20] B. Horn. Height and gradient from shading. *International Journal of Computer Vision*, 5(1):37–75, 1990.
- [21] Takashi Kanai, Hiromasa Suzuki, and Fumihiko Kimura. Metamorphosis of Arbitrary Triangular Meshes. *IEEE Computer Graphics and Applications*, 20(2):62–75, 2000.
- [22] A. Kelemen, G. Szekely, and G. Gerig. Three-Dimensional Model-based Segmentation of Brain MRI. *IEEE Transactions on Medical Imaging*, 18(10):838–849, 1999.
- [23] D. Kendall. Shape Manifolds, Procrustean Metrics and Complex Projective Spaces. *Bulletin of the London Mathematical Society*, 16:81–121, 1984.
- [24] J. Koenderink and A. Van Doorn. Dynamic shape. *Biological Cybernetics*, 53:383–396, 1986.
- [25] K. Koffka. *Principles of Gestalt psychology*. Harcourt Brace Jovanovic, 1935.
- [26] L.J. Latecki, R. Lakamper, and U. Eckhardt. Shape Descriptors for Non-Rigid Shapes with a Single Closed Contour. In *Proceedings of Computer Vision and Pattern Recognition (CVPR'00)*, pages 424–429, 2000.
- [27] M. Leventon, E. Grimson, and O. Faugeras. Statistical Shape Influence in Geodesic Active Contours. In *Proceedings of Computer Vision and Pattern Recognition (CVPR'00)*, pages 4–11, 2000.
- [28] M. Leyton. Symmetry-curvature duality. *Computer Vision, Graphics, and Image Processing*, 38:327–341, 1987.
- [29] S. Loncaric. A survey of shape analysis techniques. *Pattern Recognition*, 31(8):983–1001, 1998.
- [30] D. Marr. A theory for cerebral neocortex. In *Proceedings of the Royal Society of London*, pages 161–234, 1970.
- [31] D. Marr. Early processing of visual information. In *Proceedings of the Royal Society of London*, pages 483–519, 1976.
- [32] D. Marr and T. Poggio. A computational theory of human stereo vision. In *Proceedings of the Royal Society of London*, pages 301–328, 1979.
- [33] T. McInerney and D. Terzopoulos. Deformable Models in Medical Image Analysis: A Survey. *Medical Image Analysis*, 1(2):91–108, 1996.
- [34] E. Persoon and K. Fu. Shape Discrimination Using Fourier Descriptors. *IEEE Transactions on Systems, Man and Cybernetics*, 7(3):170–179, 1977.
- [35] A. Pitiot. *Automated Segmentation of Cerebral Structures Incorporating Explicit Knowledge*. PhD thesis, École des mines de Paris, November 2003. URL: <ftp://ftp-sop.inria.fr/epidaure/Publications/Pitiot/pitiot-thesis-2003.pdf>.
- [36] A. Pitiot, H. Delingette, A. Toga, and P. Thompson. Learning Object Correspondences with the Observed Transport Shape Measure. In *Proceedings of Information Processing in Medical Imaging IPMI'03*, 2003.
- [37] Plato. *Meno (380 BC) in W.Lamb (ed.), Plato in Twelve Volumes, vol. 3*. Harvard University Press, 1967.
- [38] R.J. Prokop and A.P. Reeves. A survey of moment-based techniques for unoccluded object representation and recognition. *CVGIP: Graphics Models and Image Processing*, 54(5):483–460, 1992.
- [39] T.B. Sebastian, J.J. Crisco, P.N. Klein, and B.B. Kimia. Constructing 2D Curve Atlases. In *Proc. of CVPR*, pages 70–77, 2000.

- [40] P.M. Thompson and A.W. Toga. Detection, Visualisation and Animation of Abnormal Anatomic Structure with a Deformable Probabilistic Brain Atlas Based on Random Vector Field Transformations. *Medical Image Analysis*, 1(4):271–294, 1997.
- [41] Alain Trouvé and Laurent Younes. Diffeomorphic Matching Problems in One Dimension: Designing and Minimizing Matching Functionals. In *Proceeding of European Conference on Computer Vision (ECCV'00)*, pages 573–587, 2000.
- [42] F. Ulupinar and R. Nevatia. Inferring shape from contour for curved surfaces. In *Proceedings of the International Conference on Pattern Recognition (ICPR'90)*, pages 147–154, 1990.
- [43] P.J. van Otterloo. *A Contour-Oriented Approach to Shape Analysis*. Prentice Hall, 1992.
- [44] R.C. Veltkamp. Shape matching: similarity measures and algorithms. In *Proceedings of the International Conference on Shape Modeling and Applications*, pages 188–199, 1991.
- [45] R.C. Veltkamp and M. Hagedoorn. State of the Art in Shape Matching. *Technical Report UU-CS-1999-27*, 1999.
- [46] Y. Wang, B.S. Peterson, and L.H. Staib. Shape-Based 3D Surface Correspondence using Geodesics and Local Geometry. In *Proceedings of Computer Vision and Pattern Recognition (CVPR'00)*, pages 644–651, 2000.
- [47] L. Zusne. *Visual Perception of Form*. Academic Press, 1970.