Affine Shape Alignment Using Covariant Gaussian Densities: A Direct Solution

Csaba Domokos and Zoltan Kato, *Senior Member, IEEE*

***Abstract*—We propose a novel approach for the estimation of 2D affine transformations aligning a planar shape and its distorted observation. The exact transformation is obtained as a least-squares solution of a linear system of equations constructed by fitting Gaussian densities to the shapes which preserve the effect of the unknown transformation. In the case of compound shapes, we also propose a robust and efficient numerical scheme achieving near real-time performance. The method has been tested on synthetic as well as on real images. Its robustness in the case of segmentation errors, missing data, and modelling error has also been demonstrated. The proposed method does not require point correspondences nor the solution of complex optimization problems, has linear time complexity and provides an exact solution regardless of the magnitude of deformation.**

***Index Terms*—Registration, shape alignment, affine transformation, covariant function, Gaussian distribution.**

EDICS Category: ARS-RBS Region, Boundary, and Shape Analysis

I. Introduction

REGISTRATION is an important step in almost all image processing tasks where images of different views or sensors need to be compared or combined. There is a rich literature on registration methods, good surveys can be found in [1]–[3]. Common application areas include target tracking in video sequences, visual inspection, super resolution, shape matching [4] or medical image analysis [3]. In a general setting, we want to estimate transformation parameters aligning two images such that one image (called the *observation*) becomes similar to the second one (called the *template*).

Basically registration algorithms fall into two main categories: *Landmark-based* and *Area-based* methods. *Landmark-based* methods [4] aim at establishing point correspondences between two images and then use these points to compute the closest transformation. Therefore, to make this approach feasible, the correspondence problem must be solved first. The common assumption needed to find good matches is that the unknown transformation is close to identity (*i.e.* the strength of the deformation is limited). Searching for the best transformation usually requires an iterative method like the Iterative Closest Point (ICP) algorithm [5], [6]. Radiometric information plays a crucial role in establishing correspondences. However, since shapes are represented as binary images, radiometric information is not available in our case. As a result, the correspondence problem becomes quite challenging. One can only use geometric information but invariant geometric features (*e.g.* corners, junctions) might be difficult to extract (a circular shape, for instance). The main advantage of these methods is that as long as a sufficient number of point matches are available, one can usually find an optimal aligning transformation implying that these algorithms are less sensitive to occlusions. On the other hand, *Area-based* (or featureless) methods [7]–[12] work without attempting to detect salient objects. Instead, the problem is solved by computing global descriptors [12] or invariants of the objects [13], [14]. When the distortion is small, often Fourier descriptors [15] or mutual information [16] is used. The drawback of this family of methods is also the high computational cost and the restricted range of distortions.

A. State of the Art

Several techniques have been developed to address the affine registration problem. By thresholding the magnitude of the Fourier transform of the images, Zhang *et al.* [15] construct affine invariant features insensitive to noise, in order to establish point correspondence. Several Fourier domain based methods [17] represent images in a coordinate system in which the affine transformation is reduced to an anisotropic scaling factor, which can be computed using cross correlation methods. Govindu and Shekar [18] develop a framework that uses the statistical distribution of geometric properties of image contours to estimate the relevant transformation parameters. The main advantages of these methods are that they do not need point correspondences across views and the images may also differ by the overall level of illumination. For matching 2D feature points, [19] reduces the general affine case to the orthogonal case by using the means and covariance matrices of the point sets, then the rotation is computed as the roots of a low-degree complex coefficients polynomial. Another direct approach [20] extends the given pattern to a set of affine covariant versions, each carrying slightly different information, and then extract features for registration from each of them separately. In [21], the transformation is parameterized at different scales, using a decomposition of the deformation vector field over a sequence of nested (multiresolution) subspaces. An energy function describing the interactions between the images is then minimized under a set of constraints, ensuring that the transformation maintains the topology in the deformed image. Manay *et al.* [22] explore an optimization framework for computing shape distance and shape matching from integral invariants which are employed for robustness to

Manuscript received ????; revised ????.

Csaba Domokos and Zoltan Kato are with the Department of Image Processing and Computer Graphics, University of Szeged, P.O. Box 652, H-6701 Szeged, Hungary. Fax:+36 62 546 397, Tel:+36 62 546 399, Email: {dcs,kato}@inf.u-szeged.hu.

This work has been partially supported by the Hungarian Scientific Research Fund – OTKA K75637.

high-frequency noise. Shape warping by the computation of an optimal reparameterization allows this method to account for large localized changes, such as occlusions, and configuration changes. In [23], a method for identifying silhouettes from a given set of Radon projections is presented. The authors study how the Radon transform changes when a given 2D function is subjected to rotation, scaling, translation, and reflection. Using these properties, the parameters of the aligning transformation are expressed in terms of the Radon transform. In [12], an image registration algorithm based on affine moments is proposed. First, some representative regions are extracted which are matched based on the similarity of their moments. Then point correspondences are established as the centers of the region pairs and the transformation is recovered in a classical way by solving a system of equations constructed from the point correspondences. Kannala *et al.* [13] construct a system of equations by basically looking at the images at 3 different scales. Flusser *et al.* [11] propose a novel image normalization process with respect to unknown affine transformations based on affine moment invariants. In Section VI, comparative results are presented with these two methods.

In many situations, the variability of image features is so complex that the only feasible way to register such images is to reduce them to a binary representation and solve the registration problem in that context [24]. Binary registration, however, is an important problem in itself for many complex image analysis tasks. Belongie *et al.* proposed a novel approach for shape matching [4]. The method first extracts points by densely sampling the contour, then point matches between the two objects are established using a novel similarity metric, called *shape context*, which consists in constructing a log-polar histogram of surrounding edge pixels. Obviously, there is no guarantee that point pairs are exactly corresponding because of the sampling procedure. The correspondences are simply established by solving a linear assignment problem, which requires time consuming optimization methods. The advantage compared to traditional landmark based approaches is that landmarks need not be salient points nor radiometric information is involved. A novel segment-based shape matching algorithm is presented in [25] which avoids problems associated with purely global or local methods. This approach generalizes the idea of finding a point-to-point correspondence between two shapes to that of finding a segment-to-segment correspondence.

In many applications, registration appears as the problem of aligning a set of points. In [26], a robust approach is proposed, where each point set is represented by a mixture of spherical Gaussians and the point set registration is treated as a problem of aligning the two mixtures. For this purpose, the authors derived a closed-form expression for the L_2-distance between two Gaussian mixtures, which in turn leads to a computationally efficient registration algorithm. Although our approach also constructs a mixture of Gaussians, the fundamental difference is that we do not represent shapes while [26] basically recovers the underlying continuous shape from the discrete point set using a method similar to Kernel Density Estimation (KDE). An affine registration algorithm for matching 2D feature points is presented in [19], which recovers both the aligning affine transformation as well as the unknown correspondences. The algorithm conists of two steps: first the general affine case is reduced to the orthogonal case, then the unknown rotation is computed as the roots of a low-degree polynomial with complex coefficients. Coherent Point Drift (CPD) is introduced in [27], which is a probabilistic method for nonrigid registration of point sets. Coherent Point Drift (CPD) [27] is a probabilistic approach for non-rigid registration of point sets. CPD simultaneously recovers the non-rigid transformation and the correspondence between the point sets without making any prior assumption on the transformation model except that of motion coherence. The registration is treated as a Maximum Likelihood (ML) estimation problem with motion coherence constraint over the velocity field such that one point set moves coherently to align with the second set.

B. Contributions

The parametric estimation of two-dimensional affine transformations between two *gray-level* images has been addressed by Hagege and Francos in [28] which provides an accurate and computationally simple solution avoiding both the correspondence problem as well as the need for optimization. The basic idea is to reformulate the original problem as an *equivalent linear parameter estimation* one which can be easily solved. This solution, however, relies on radiometric information which is not available in binary images. In our previous work [29], [30], an extension of these ideas has been proposed to the binary case.

Herein, following our ideas outlined in [29], [30], we will propose a novel method which provides an accurate and computationally simple solution to the affine registration of planar shapes. The main benefits of the algorithmic solution presented in this paper are increased robustness compared to [29] and a numerically more efficient computation scheme compared to [30], allowing near real-time registration. The main difficulty with binary images is that they do not contain radiometric information, only the foreground pixel coordinates are available for the registration algorithm. We will show, that in spite of the missing radiometric information, we can still formulate the registration problem as the solution of a linear system of equations. The basic idea is to generate a pair of *covariant functions* that are related by the unknown transformation. The resulting algorithm is fast and provides a direct solution without establishing correspondences.

This paper organizes as follows. We introduce our registration problem in Section II followed by the theoretical description of the proposed approach in Section III and the discussion of its properties from various viewpoints in Section IV. We then introduce an efficient numerical implementation scheme in Section V. The performance and robustness of our method is analyzed in Section VI on a large synthetic benchmark dataset as well as in the context of traffic sign matching. Furthermore, comparative tests on synthetic data against state-of-the-art methods are also presented. Finally, Section VII concludes the paper.

II. Problem statement

Let us denote *template* and *observation* points by $\mathbf{x} = [x_1, x_2]^T \in \mathbb{R}^2$ and $\mathbf{y} = [y_1, y_2]^T \in \mathbb{R}^2$ respectively. The relation between the shapes is then as follows

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{t} \quad \Leftrightarrow \quad \mathbf{x} = \mathbf{A}^{-1}(\mathbf{y} - \mathbf{t}) = \mathbf{A}^{-1}\mathbf{y} - \mathbf{A}^{-1}\mathbf{t}, \tag{1}$$

where $(\mathbf{A}, \mathbf{t}) \in (\mathbb{R}^{2\times 2} \times \mathbb{R}^{2\times 1})$ is the unknown affine transformation that we want to recover. Classical landmark-based approaches would now identify at least 3 point pairs $\{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^{n\geq 3}$ then solve the system of linear equations obtained from Eq. (1). However, we are interested in a direct approach without solving the correspondence problem. For that purpose, shapes will be represented by their characteristic function $\mathbb{1} : \mathbb{R}^2 \to \{0, 1\}$, where 0 and 1 correspond to the background and foreground respectively. If we denote the *template* by $\mathbb{1}_t$ and the *observation* by $\mathbb{1}_o$, the following equality also holds

$$\mathbb{1}_t(\mathbf{x}) = \mathbb{1}_o(\mathbf{A}\mathbf{x} + \mathbf{t}) = \mathbb{1}_o(\mathbf{y}). \tag{2}$$

When we can observe some image features (*e.g.* gray-levels of the pixels [28]) that are invariant under the transformation, then we can define an additional relation

$$f(\mathbf{x}) = g(\mathbf{A}\mathbf{x} + \mathbf{t}) = g(\mathbf{y}), \tag{3}$$

where $f, g : \mathbb{R}^2 \to \mathbb{R}$ are *covariant functions* under the transformation $(\mathbf{A}, \mathbf{t})$, defined on those observed features. Furthermore, the above relations are still valid when a function is acting on both sides of Eq. (1) and Eq. (3) [10], [28]–[30]. Indeed, for properly chosen $\eta : \mathbb{R}^2 \to \mathbb{R}$ and $\omega : \mathbb{R} \to \mathbb{R}$, we get

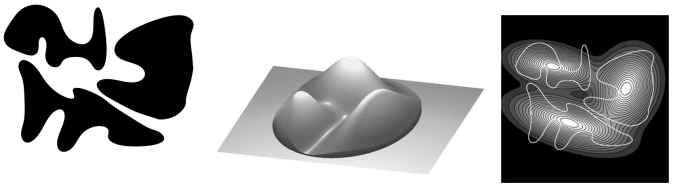
$$\eta\big(\mathbf{y}\big) = \eta\big(\mathbf{A}\mathbf{x} + \mathbf{t}\big) \tag{4}$$

$$\omega \circ g(\mathbf{y}) = \omega \circ g(\mathbf{A}\mathbf{x} + \mathbf{t}) = \omega \circ f(\mathbf{x}). \tag{5}$$

Starting from either Eq. (4) or Eq. (5), we can generate as many linearly independent equations as needed by making use of nonlinear η (resp. ω) functions. There is a fundamental difference between the above two equations though: the nonlinear function η is acting directly on the point coordinates and hence on the unknown parameters of $(\mathbf{A}, \mathbf{t})$ resulting in a *nonlinear* system of equations [10]; whereas ω is acting on the *covariant functions* f and g allowing for a *linear* system of equations [28]–[30]. Herein, we will explore the latter idea.

III. Estimation of affine transformations

The crucial step of the proposed approach is to construct a pair of *covariant functions* satisfying Eq. (3). Once these functions are established, we can adopt the direct method from [28], [29] to solve for the unknown transformation $(\mathbf{A}, \mathbf{t})$. When graylevel images are considered, the image functions themself serve as appropriate covariant functions [28]. Unfortunately, the construction of such functions for binary images is a quite challenging task due to the lack of radiometric information: These functions must be based on the only available geometric information.



(a) (b) (c)

Fig. 1. Gaussian PDFs fitted over a compound shape yield a consistent coloring. (a) Original shape; (b) 3D plot of the Gaussian PDFs over the elliptic domain with $r = 2$; (c) Gaussian densities as a grayscale image. The white contour shows shape boundaries.

A. Construction of Covariant Functions

Since we know that the *template* and *observation* are identical up to an affine transformation (this is stated in Eq. (1)), we do not need to represent shapes. Therefore we can safely consider the points of the *template* as a sample from a bivariate normally distributed random variable denoted by $X \sim N(\mu, \Sigma)$ with probability density function (PDF) [29]

$$p(\mathbf{x}) = \frac{1}{2\pi\sqrt{|\Sigma|}} \exp\Big(-\frac{1}{2}(\mathbf{x} - \mu)^T\Sigma^{-1}(\mathbf{x} - \mu)\Big).$$

Applying any linear transformation to X results also in a bivariate normal random variable $Y = \mathbf{A}X + \mathbf{t}$ with parameters

$$X \overset{(\mathbf{A},\mathbf{t})}{\mapsto} Y \sim N(\mu', \Sigma') = N(\mathbf{A}\mu + \mathbf{t}, \mathbf{A}\Sigma\mathbf{A}^T). \tag{6}$$

Obviously, the above equation is only valid when $\mathbf{A}$ is non-singular and positive definite. In our case, $(\mathbf{A}, \mathbf{t})$ is an affine transformation thus $\mathbf{A}$ is clearly non-singular. On the other hand, a negative determinant would mean that the transformation is not orientation-preserving. In practice, however, such transformations are usually excluded by physical constraints hence we can assume that $|\mathbf{A}|$ is always positive. The parameters of the probability densities $N(\mu, \Sigma)$ and $N(\mu', \Sigma')$ can be readily estimated as the sample means and covariances of the point coordinates, while $|\mathbf{A}|$ can be expressed from $\mathbf{A}\Sigma\mathbf{A}^T = \Sigma'$ as

$$|\mathbf{A}||\Sigma||\mathbf{A}^T| = |\Sigma'|, \quad \text{hence} \quad |\mathbf{A}| = \sqrt{|\Sigma'|/|\Sigma|}. \tag{7}$$

From a geometric point of view, the mean values μ and μ' represent the center of mass of the *template* and *observation* respectively, while Σ and Σ' capture the orientation and eccentricity of the shapes. Fig. 1 shows a compound shape and the fitted Gaussian densities of each component.

Now let us have a closer look at the relationship between $p(\mathbf{x})$ and $s(\mathbf{y})$, the PDF of Y. It follows from Eq. (6), that $\Sigma'^{-1} = \mathbf{A}^{-T}\Sigma^{-1}\mathbf{A}^{-1}$, furthermore

$$(\mathbf{y} - \mu') = \big(\mathbf{A}\mathbf{x} + \mathbf{t} - (\mathbf{A}\mu + \mathbf{t})\big) = (\mathbf{A}\mathbf{x} - \mathbf{A}\mu).$$

We thus get

$$s(\mathbf{y}) = \frac{\exp\Big(-\frac{1}{2}(\mathbf{A}\mathbf{x} - \mathbf{A}\mu)^T\mathbf{A}^{-T}\Sigma^{-1}\mathbf{A}^{-1}(\mathbf{A}\mathbf{x} - \mathbf{A}\mu)\Big)}{2\pi\sqrt{|\mathbf{A}||\Sigma|\mathbf{A}^T|}}$$

$$= \frac{\exp\Big(-\frac{1}{2}(\mathbf{x} - \mu)^T\mathbf{A}^T\mathbf{A}^{-T}\Sigma^{-1}\mathbf{A}^{-1}\mathbf{A}(\mathbf{x} - \mu)\Big)}{|\mathbf{A}|2\pi\sqrt{|\Sigma|}} = \frac{p(\mathbf{x})}{|\mathbf{A}|}.$$

Finally, substituting back $|\mathbf{A}|$ from Eq. (7), we get

$$\sqrt{|\Sigma|}p(\mathbf{x}) = \sqrt{|\Sigma'|}s(\mathbf{y}). \tag{8}$$

It is well known, that the normalizing constant $1/(2\pi\sqrt{|\Sigma|})$ in the density functions ensures that the integral of the PDF evaluates to 1. It is also the maximum value of the density function, which is inversely proportional to the area of the shape. This dependence on the shape size may cause numerical instabilities hence we define our *covariant functions* $\mathcal{P}, \mathcal{S} : \mathbb{R}^2 \rightarrow \mathbb{R}$ as the unnormalized densities

$$\mathcal{P}(\mathbf{x}) = 2\pi\sqrt{|\Sigma|}p(\mathbf{x}) = \exp\left(-\frac{1}{2}(\mathbf{x}-\mu)^T\Sigma^{-1}(\mathbf{x}-\mu)\right), \tag{9}$$

$$\mathcal{S}(\mathbf{y}) = 2\pi\sqrt{|\Sigma'|}s(\mathbf{y}) = \exp\left(-\frac{1}{2}(\mathbf{y}-\mu')^T\Sigma'^{-1}(\mathbf{y}-\mu')\right).$$

Since the covariance matrices and mean vectors can be computed from the images, both $\mathcal{P}$ and $\mathcal{S}$ is obtained directly from the input shapes and they are *covariant* satisfying Eq. (3):

$$\mathcal{P}(\mathbf{x}) = \mathcal{S}(\mathbf{A}\mathbf{x}+\mathbf{t}) = \mathcal{S}(\mathbf{y}). \tag{10}$$

B. Linear Estimation of Affine Parameters

Since point correspondences are not available, we cannot construct a system directly from Eq. (1) or Eq. (10). We know, however, that the finite domains of the *template* and *observation* , $\mathcal{F}_t = \{\mathbf{x} \in \mathbb{R}^2 | \mathbb{1}_t(\mathbf{x}) = 1\}$ and $\mathcal{F}_o = \{\mathbf{y} \in \mathbb{R}^2 | \mathbb{1}_o(\mathbf{y}) = 1\}$ respectively, are related by $(\mathbf{A}, \mathbf{t})$: $\mathbf{A}\mathcal{F}_t + \mathbf{t} = \mathcal{F}_o$ [29]. Therefore multiplying Eq. (1) and Eq. (10), we can integrate out individual point correspondences:

$$\int_{\mathcal{F}_t} \mathbf{x}\mathcal{P}(\mathbf{x})d\mathbf{x} = |\mathbf{A}|^{-1}\int_{\mathcal{F}_o} \mathbf{A}^{-1}(\mathbf{y}-\mathbf{t})\mathcal{S}(\mathbf{y})d\mathbf{y},$$

where we have used the integral transformation $\mathbf{x} = \mathbf{A}^{-1}(\mathbf{y}-\mathbf{t})$, $d\mathbf{x} = |\mathbf{A}|^{-1}d\mathbf{y}$. Note that the Jacobian $|\mathbf{A}|$ can be easily computed from the input images based on Eq. (7). In order to generate more linearly independent equations, we will adopt appropriate nonlinear functions $\omega : \mathbb{R} \rightarrow \mathbb{R}$ and generate new equations according to Eq. (5):

$$\int_{\mathcal{F}_t} \mathbf{x}\omega\big(\mathcal{P}(\mathbf{x})\big)d\mathbf{x} = |\mathbf{A}|^{-1}\int_{\mathcal{F}_o} \mathbf{A}^{-1}(\mathbf{y}-\mathbf{t})\omega\big(\mathcal{S}(\mathbf{y})\big)d\mathbf{y}. \tag{11}$$

If q_{ki} denotes the elements of $\mathbf{A}^{-1}$ and $-\mathbf{A}^{-1}\mathbf{t}$

$$\mathbf{A}^{-1} = \begin{bmatrix} q_{11} & q_{12} \\ q_{21} & q_{22} \end{bmatrix} \quad \text{and} \quad -\mathbf{A}^{-1}\mathbf{t} = \begin{bmatrix} q_{13} \\ q_{23} \end{bmatrix},$$

we can expand the integrals yielding the following linear system

$$|\mathbf{A}|\int_{\mathcal{F}_t} x_k\omega\big(\mathcal{P}(\mathbf{x})\big)d\mathbf{x} =$$

$$\sum_{i=1}^{2} q_{ki}\int_{\mathcal{F}_o} y_i\omega\big(\mathcal{S}(\mathbf{y})\big)d\mathbf{y} \;+\; q_{k3}\int_{\mathcal{F}_o} \omega\big(\mathcal{S}(\mathbf{y})\big)d\mathbf{y}, \;\; k = 1, 2.$$

Adopting a set of linearly independent functions $\{\omega_i\}_{i=1}^{\ell}$, we can rewrite the system in matrix form

$$\begin{bmatrix} \int_{\mathcal{F}_o} y_1\omega_1\big(\mathcal{S}(\mathbf{y})\big) & \int_{\mathcal{F}_o} y_2\omega_1\big(\mathcal{S}(\mathbf{y})\big) & \int_{\mathcal{F}_o} \omega_1\big(\mathcal{S}(\mathbf{y})\big) \\ \vdots & \vdots & \vdots \\ \int_{\mathcal{F}_o} y_1\omega_\ell\big(\mathcal{S}(\mathbf{y})\big) & \int_{\mathcal{F}_o} y_2\omega_\ell\big(\mathcal{S}(\mathbf{y})\big) & \int_{\mathcal{F}_o} \omega_\ell\big(\mathcal{S}(\mathbf{y})\big) \end{bmatrix} \times$$

$$\begin{bmatrix} q_{k1} \\ q_{k2} \\ q_{k3} \end{bmatrix} = |\mathbf{A}| \begin{bmatrix} \int_{\mathcal{F}_t} x_k\omega_1\big(\mathcal{P}(\mathbf{x})\big) \\ \vdots \\ \int_{\mathcal{F}_t} x_k\omega_\ell\big(\mathcal{P}(\mathbf{x})\big) \end{bmatrix}, \quad k = 1, 2. \tag{12}$$

The solution of this linear system provides the parameters of the registration. If $\ell > 3$ then the system is over-determined and the solution is obtained as a least squares solution. Note that independently of the number of systems, the coefficient matrix need to be computed only once. Hence the complexity of the algorithm depends linearly on the size of the shapes.

C. Choosing the Integration Domain

A trivial choice for the domains in our integral equation Eq. (11) is the foreground regions $\mathcal{F}_t$ and $\mathcal{F}_o$ [29]. Since the parameters of the transformation are estimated by integrating over the segmented domains, this approach works well as long as we have a near-perfect segmentation. Unfortunately, this is rarely encountered in reality [30]. Therefore a clear disadvantage of this approach is that any segmentation error will inherently result in erroneous integrals causing misalignment. Furthermore, even if the segmentation is perfect, the precision of these domains is always compromised by discretization error. On the other hand, image analysis often deals with the matching of objects composed of several parts, yielding a group of disjoint shapes when segmented. The topology of such compound shapes will not change under the action of the affine group. Herein, we will develop a robust method to define corresponding integration domains making use of the statistics of compound shapes.

Let us assume that the *template* consists of $m \geq 2$ disjoint shapes. This is the typical output of classical region-based segmentation algorithms, where the labelling of the different regions results in disjoint shapes, but similar results can be achieved by detecting the conneted components of a compound object. In both cases, the input of our method will be a pair of labeled images, where each component on the *template* has exactly one corresponding shape on the *observation*, *i.e.* there exists a bijective mapping between the *template* and *observation* components under the transformation $(\mathbf{A}, \mathbf{t})$. For each pair of corresponding components, we can establish *covariant functions* $\mathcal{P}_i$ and $\mathcal{S}_i$ similar to Eq. (9):

$$\mathcal{P}_i(\mathbf{x}) = \exp\Big(-\frac{1}{2}(\mathbf{x}-\mu_i)^T\Sigma_i^{-1}(\mathbf{x}-\mu_i)\Big)$$

$$\mathcal{S}_i(\mathbf{y}) = \exp\Big(-\frac{1}{2}(\mathbf{y}-\mu_i')^T\Sigma_i'^{-1}(\mathbf{y}-\mu_i')\Big),$$

wehere Σ_i,Σ_i' and μ_i,μ_i' are the covariance matrices and mean vectors of the i^{th} shape on the *template* and *observation*, respectively. This yields an equation similar to Eq. (11) for each $1 \leq i \leq m$. If the correspondence between components would be known then we could simply construct a system of m equations and solve for the unknowns. As such a matching is not known, we will sum these relations yielding

$$\sum_{i=1}^{m}\int_{\mathcal{F}_t} \mathbf{x}\omega\big(\mathcal{P}_i(\mathbf{x})\big)d\mathbf{x} = \int_{\mathcal{F}_t} \mathbf{x}\sum_{i=1}^{m}\omega\big(\mathcal{P}_i(\mathbf{x})\big)d\mathbf{x} = \tag{13}$$

$$|\mathbf{A}|^{-1}\int_{\mathcal{F}_o} \mathbf{A}^{-1}(\mathbf{y}-\mathbf{t})\sum_{i=1}^{m}\omega\big(\mathcal{S}_i(\mathbf{y})\big)d\mathbf{y}.$$

$$\begin{bmatrix} \int_{\mathcal{D}_o} y_1 \sum_{i=1}^{m} \omega_1\big(\mathcal{S}_i(\mathbf{y})\big) & \int_{\mathcal{D}_o} y_2 \sum_{i=1}^{m} \omega_1\big(\mathcal{S}_i(\mathbf{y})\big) & \int_{\mathcal{D}_o} \sum_{i=1}^{m} \omega_1\big(\mathcal{S}_i(\mathbf{y})\big) \\ \vdots & \vdots & \vdots \\ \int_{\mathcal{D}_o} y_1 \sum_{i=1}^{m} \omega_\ell\big(\mathcal{S}_i(\mathbf{y})\big) & \int_{\mathcal{D}_o} y_2 \sum_{i=1}^{m} \omega_\ell\big(\mathcal{S}_i(\mathbf{y})\big) & \int_{\mathcal{D}_o} \sum_{i=1}^{m} \omega_\ell\big(\mathcal{S}_i(\mathbf{y})\big) \end{bmatrix} \begin{bmatrix} q_{k1} \\ q_{k2} \\ q_{k3} \end{bmatrix} = |\mathbf{A}| \begin{bmatrix} \int_{\mathcal{D}_t} x_k \sum_{i=1}^{m} \omega_1\big(\mathcal{P}_i(\mathbf{x})\big) \\ \vdots \\ \int_{\mathcal{D}_t} x_k \sum_{i=1}^{m} \omega_\ell\big(\mathcal{P}_i(\mathbf{x})\big) \end{bmatrix}, \; k = 1, 2 \tag{16}$$

The next step is to get rid of the segmentation domains $\mathcal{F}_t$ and $\mathcal{F}_o$. Our goal is to to select appropriate domains $\mathcal{D}_t$ and $\mathcal{D}_o$ satisfying the following properties

1) they are related by the unknown transformation $\mathbf{A}\mathcal{D}_t + \mathbf{t} = \mathcal{D}_o$
2) the integrands are rich enough (*i.e.* have characteristic pattern) within the selected domains.

The key idea is using the statistics of the whole *template* and *observation* objects. Indeed, the overall shape (*i.e.* the whole foreground region) of the *template* and *observation* also gives rise to a pair of covariant Gaussian densities $p(\mathbf{x})$ and $s(\mathbf{y})$. Since the equidensity contours of these PDFs are ellipsoids centered at the mean, it is natural to chose a corresponding pair of these ellipses as the integration domain. Simplifying Eq. (8), we get the well known Mahalanobis-distance which defines a metric invariant under the unknown transformation $(\mathbf{A}, \mathbf{t})$:

$$(\mathbf{x} - \mu)^T \Sigma^{-1} (\mathbf{x} - \mu) = (\mathbf{y} - \mu')^T \Sigma'^{-1} (\mathbf{y} - \mu').$$

Corresponding domains can then be obtained by selecting points whose Mahalanobis-distance are less than r^2 from the mean:

$$\mathcal{D}_t = \{\mathbf{x} \in \mathbb{R}^2 | (\mathbf{x} - \mu)^T \Sigma^{-1} (\mathbf{x} - \mu) \leq r^2\} \tag{14}$$

$$\mathcal{D}_o = \{\mathbf{y} \in \mathbb{R}^2 | (\mathbf{y} - \mu')^T \Sigma'^{-1} (\mathbf{y} - \mu') \leq r^2\} \tag{15}$$

To satisfy property 2), we may choose an ellipse according to the *two sigma rule* (*i.e.* $r = 2$), which guarantees that about 95% of values are within the enclosed ellipsoid (see (Fig. 1)). Experiments show that good alignments can be achieved by settings ranging from $r = 1$ to $r = 3$. Another advantage is that these domains are analytical which allows for a quite efficient numerical implementation scheme, as discussed in Section V.

In summary, all we need to construct a system of linear equations are the means and covariances of the input shapes. Based on these statistics, we can select the integration domains and construct appropriate covariant functions yielding the system Eq. (16) similar to Eq. (12).

IV. Discussion

Herein, we give two alternative interpretations of the proposed framework, thereby relating our approach to other state-of-the-art approaches of shape alignment. First, we analyse the method and show its relation to the metric based framework proposed by Bronstein *et al.* [31]. Then the relation with classical moment-based approaches is discussed.

A. Relation to Metric-Based Approaches

Bronstein *et al.* proposed a generic framework for non-rigid shape matching in [31], [32], where the problem is studied from the perspective of metric geometry. The basic idea is to construct a so-called *canonical representation* of the original shape. Making use of this representation, they are able to either detect symmetry of non-rigid shapes [32] or solve a non-rigid object recognition problem [31]. The mathematical background of this approach is based on measure theory. Herein, we will discover the similarities and differences between our approach and the framework presented by Bronstein *et al.* . In particular, we will formally analyze the case of a single pair of shapes outlined in Section III-A and Section III-B.

Following [31], let the *template* and *observation* be modeled as metric spaces $(\mathcal{F}_t, d_t)$ and $(\mathcal{F}_o, d_o)$, where $d_t, d_o : \mathbb{R}^2 \to \mathbb{R}_0^+$

$$d_t(\mathbf{u}, \mathbf{v}) = (\mathbf{u} - \mathbf{v})^T \Sigma^{-1} (\mathbf{u} - \mathbf{v}) \quad \text{and}$$

$$d_o(\mathbf{u}, \mathbf{v}) = (\mathbf{u} - \mathbf{v})^T \Sigma'^{-1} (\mathbf{u} - \mathbf{v})$$

are the Mahalanobis-distances constructed on the *template* and *observation*, respectively. This kind of metric is often called an *extrinsic* metric [32]. Note that our Gaussians $\mathcal{P}$ and $\mathcal{S}$ are uniquely determined by the respective Mahalanobis-distances in their exponents, hence the above metrics are equivalent to our covariant functions in Eq. (9). Now, we will show that $(\mathcal{F}_t, d_t)$ and $(\mathcal{F}_o, d_o)$ are *isometric*, *i.e.* there exist a bijective homomorphism $g : (\mathcal{F}_t, d_t) \to (\mathcal{F}_o, d_o)$. Indeed, the map $g : \mathbb{R}^2 \to \mathbb{R}^2$ is given by the unknown affine transformation $(\mathbf{A}, \mathbf{t})$: $g(\mathbf{u}) = \mathbf{A}\mathbf{u} + \mathbf{t}$. g is clearly bijective since $\mathbf{A}$ is nonsingular, while it is also easily seen that g is a homomorphism:

$$\begin{aligned} d_t(\mathbf{u}, \mathbf{v}) &= (\mathbf{u} - \mathbf{v})^T \Sigma^{-1} (\mathbf{u} - \mathbf{v}) \\ &= (\mathbf{A}^{-1} g(\mathbf{u}) - \mathbf{A}^{-1}\mathbf{t} - \mathbf{A}^{-1} g(\mathbf{v}) + \mathbf{A}^{-1}\mathbf{t})^T \\ &\quad \mathbf{A}^{-T} \Sigma'^{-1} \mathbf{A}^{-1} \\ &\quad (\mathbf{A}^{-1} g(\mathbf{u}) - \mathbf{A}^{-1}\mathbf{t} - \mathbf{A}^{-1} g(\mathbf{v}) + \mathbf{A}^{-1}\mathbf{t}) \\ &= \big(g(\mathbf{u}) - g(\mathbf{v})\big)^T \Sigma'^{-1} \big(g(\mathbf{v}) - g(\mathbf{v})\big) \\ &= d_o\big(g(\mathbf{u}), g(\mathbf{v})\big). \end{aligned}$$

We remark that the only possible isometries are affine transformations, denoted by $\mathrm{Iso}(\mathcal{F}_t, d_t)$. This is obvious, because of the properties of the Mahalanobis-distance (equidistant points are lying on an ellipse, which can be obtained from an arbitrary ellipse by applying an affine transformation). Thus $\mathcal{F}_t$ and any $\mathcal{F} \subset \mathbb{R}^2$ are extrinsically isometric [32], if there exist $g \in \mathrm{Iso}(\mathcal{F}_t, d_t)$, such that $g(\mathcal{F}_t) = \mathcal{F}$. This means that $\mathcal{F}$ can

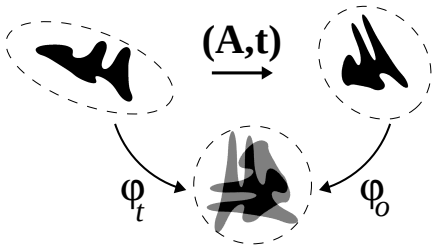


Fig. 2. Canonical representation of a *template* and its *observation*.

be obtained from $\mathcal{F}_t$ by applying an affine transformation. In this case $\mathcal{F}_t$ and $\mathcal{F}$ are called *congruent* [32].

For shape recognition [11], [14], [31], the *canonical form* of the shapes is used, because this representation is uniquely determined up to a rigid motion by the shape regardless of its deformation (see Fig. 2). Therefore, if two shapes are identical but deformed, then their *canonical forms* are related by a simple rigid body transformation which is easy to verify [33], [34] once *canonical representation* of the shapes are available. In the framework of Bronstein *et al.* [31], *canonical form* is obtained by embedding the shapes into the Euclidean space $(\mathbb{E}, d_{\mathbb{E}})$, where the only possible isometires are rigid motions and the ordinary Euclidean metric $d_{\mathbb{E}}(\mathbf{u}) = \mathbf{u}^T\mathbf{u}$ is used. In the affine case, the embedding of the *template* $\varphi : (\mathcal{F}_t, d_t) \rightarrow (\mathbb{E}, d_{\mathbb{E}})$ is given as $\varphi(\mathbf{u}) = \mathbf{R}^T\mathbf{u}$, where $\mathbf{R}$ is obtained by factorizing the inverse covariance matrix Σ^{-1} of $\mathcal{F}_t$:

$$\begin{aligned}\Sigma^{-1} = \mathbf{U}\mathbf{V}\mathbf{U}^T &= \mathbf{U}\sqrt{\mathbf{V}}\sqrt{\mathbf{V}}\mathbf{U}^T \\ &= (\mathbf{U}\sqrt{\mathbf{V}})(\mathbf{U}\sqrt{\mathbf{V}})^T = \mathbf{R}\mathbf{R}^T, \quad (17)\end{aligned}$$

where $\mathbf{V}$ is diagonal and $\mathbf{U}$ could be either orthogonal or upper triangular depending on whether we use *SVD* or *LR* decomposition. Note that the factorization of Σ^{-1} is not unique, but this ambiguity causes only different rigid transformations within the Euclidean space. Making use of $\mathbf{u} = \mathbf{R}^{-T}\varphi(\mathbf{u})$, it is easy to see that φ is a homomorphism, *i.e.* $d_t = d_{\mathbb{E}}(\varphi \times \varphi)$:

$$\begin{aligned} d_t(\mathbf{u},\mathbf{v}) &= (\mathbf{u}-\mathbf{v})^T\Sigma^{-1}(\mathbf{u}-\mathbf{v}) \\ &= \big(\mathbf{R}^{-T}\varphi(\mathbf{u}) - \mathbf{R}^{-T}\varphi(\mathbf{v})\big)^T\mathbf{R}\mathbf{R}^T \\ &\quad \big(\mathbf{R}^{-T}\varphi(\mathbf{u}) - \mathbf{R}^{-T}\varphi(\mathbf{v})\big) \\ &= \big(\varphi(\mathbf{u}) - \varphi(\mathbf{v})\big)^T\mathbf{R}^{-1}\mathbf{R}\mathbf{R}^T\mathbf{R}^{-T}\big(\varphi(\mathbf{u}) - \varphi(\mathbf{v})\big) \\ &= \big(\varphi(\mathbf{u}) - \varphi(\mathbf{v})\big)^T\big(\varphi(\mathbf{u}) - \varphi(\mathbf{v})\big) \\ &= d_{\mathbb{E}}\big(\varphi(\mathbf{u}), \varphi(\mathbf{v})\big). \end{aligned}$$

Since Σ^{-1} is positive definite, $\mathbf{R}^T$ is nonsingular, hence φ is injective because $\varphi(\mathbf{u})$ has a unique solution for all $\mathbf{u}$. Furthermore $\varphi|_{\mathcal{F}_t} = \varphi_t \in \mathrm{Iso}(\mathcal{F}_t, d_t)$, *i.e.* φ_t is bijective over $\mathcal{F}_t$ and $\mathcal{F}_c = \varphi_t(\mathcal{F}_t)$ is called the *canonical form* of the shape $\mathcal{F}_t$ (see Fig. 2).

While recognition deals with the question of whether two shapes are identical under a certain class of deformations, registration wants to recover an aligning transformation between two shapes that are known to be identical but deformed. Obviously, these are strongly related problems - solving either will implicitly solve the other. In particular, recognition via *canonical representation* will implicitly solve the alignment problem of the shapes because $(\mathbf{A}, \mathbf{t})$ is obtained as $\varphi_o^{-1} \circ E \circ \varphi_t$, where E is the rigid body transformation between the *canonical forms*. This is exploited in [11], [14].

The fundamental difference is that our approach doesn't recover *canonical shapes*. Instead of embedding the shapes into an Euclidean space, we apply nonlinear ω functions to the metrics (d_t, d_o) to get any number of *isometric* spaces. This is easily achieved since $\omega \circ d_t = \omega \circ d_o(g \times g)$ thus $g : (\mathcal{F}_t, \omega(d_t)) \rightarrow (\mathcal{F}_o, \omega(d_o))$ will also be an *isometry*.

B. Moment-based Interpretation

Image moments and invariants were introduced by Hu [35] for 2D pattern analysis. Since then, they became one of the most popular region-based descriptors because any shape can be reconstructed from its infinite set of moments [36]. Traditional two dimensional $(p+q)$th order moments of a function $\rho : \mathbb{R}^2 \rightarrow \mathbb{R}$ are defined as $m_{pq} = \int_{\mathbb{R}^2} x_1^p x_2^q \rho(\mathbf{x}) d\mathbf{x}$, where $p, q \in \mathbb{N}_0$. When ρ is an image function then these moments are also referred to as *image moments*. In the binary case, where objects are represented by their silhouette, ρ is a characteristic function yielding $m_{pq} = \int_{\mathcal{F}} x_1^p x_2^q d\mathbf{x}$, where $\mathcal{F} = \{\mathbf{x} \in \mathbb{R}^2 : \rho(\mathbf{x}) = 1\}$. This is often called a *shape moment*. Generally, orthogonal moments, such as Legendre [36] or Zernike moments [37], are numerically more stable than regular moments. We remark, however, that orthogonal moments can always be expressed by a finite set of regular moments.

In this sense, we can recognize first order *function moment*s $m_{10} = \int_{\mathcal{F}} x_1 \rho(\mathbf{x}) d\mathbf{x}$ in Eq. (16). A fundamental question is what kind of ρ function could be used instead of the characteristic function in order to solve the registration problem. As we pointed out, we need *covariant* functions. While invariants are immune to the action of the affine group, *covariant* functions vary with the actual transformation thus providing constraints on the unknown parameters. Indeed, from Eq. (5) we get: $\rho = \omega \circ f$, where ω can be any function satisfying Eq. (5). It is clear that higher order moments should be avoided in order to keep our equations *linear*. Instead, a set of linearly independent functions $\{\omega_i\}_{i=1}^{\ell}$ is adopted to generate appropriate moments. Theoretically almost any nonlinear ω could be used, but we will show in the next section, that *power* functions are computationally favorable.

V. Numerical implementation

We have constructed our equations in the continuum but in practice we only have a limited precision digital image. This means that the integrals can only be *approximated*. Another issue is large numeric errors caused by highly varying pixel coordinates. A standard technique to minimize this error is to normalize the *template* and *observation* into $[-1,1] \times [-1,1]$. Normalization is composed of a translation of the origin into the center of the shape followed by an appropriate scaling. If $s_k = \max_{\forall \mathbf{x} \in \mathcal{F}_t} (||x_k - \mu^{(k)}||)$, then

$$\mathbf{N}_t = \begin{bmatrix} 1/s_1 & 0 \\ 0 & 1/s_2 \end{bmatrix}, \text{ and } \mathbf{t}_t = \begin{bmatrix} -\mu^{(1)}/s_1 \\ -\mu^{(2)}/s_2 \end{bmatrix}, \quad (18)$$

where the coordinates of the mean vector are $\mu = [\mu^{(1)}, \mu^{(2)}]^T$. Normalization of the *observation* $\mathbf{N}_o$ is computed similarly. Note, that the normalized mean and covariance of the fitted Gaussians can also be computed from the unnormalized parameters: $\mu^* = 0$ because of the normalization and $\Sigma^* = \mathbf{N}_t \Sigma \mathbf{N}_t^T$; furthermore $\mu_i^* = \mathbf{N}_t\mu_i - \mathbf{N}_t\mu$ and $\Sigma_i^* = \mathbf{N}_t\Sigma_i\mathbf{N}_t^T$. Applying the normalizing transformation $(\mathbf{N}_t, \mathbf{t}_t)$ to the *template* and $(\mathbf{N}_o, \mathbf{t}_o)$ to the *observation*, the algorithm will recover the transformation $(\mathbf{A}^*, \mathbf{t}^*)$ which aligns the normalized shapes. Then the original transformation $(\mathbf{A}, \mathbf{t})$ is recovered by unnormalizing $(\mathbf{A}^*, \mathbf{t}^*)$:

$$\mathbf{A} = \mathbf{N}_o^{-1}\mathbf{A}^*\mathbf{N}_t, \text{ and } \mathbf{t} = \mathbf{N}_o^{-1}\mathbf{A}^*\mathbf{t}_t + \mathbf{N}_o^{-1}\mathbf{t}^* - \mathbf{t}_o. \quad (19)$$

Furthermore, the normalization has to be taken into account in our equations as it affects the measure of integrals. Therefore, when normalizing the images, the left and right hand sides of the equations in Eq. (16) have to be multiplied by $|\mathbf{N}_t|$ and $|\mathbf{N}_o|$ respectively and the Jacobian of the transformation is

$$|\mathbf{A}^*| = \frac{|\mathbf{N}_o|}{|\mathbf{N}_t|}|\mathbf{A}|. \quad (20)$$

A. Computing the Integrals

Herein, we will develop an efficient numerical scheme to compute the integrals over the elliptic domains $\mathcal{D}_t$ and $\mathcal{D}_o$ defined in Section III-C. For that purpose, let us consider power functions x^n as the applied ωs (see Fig. 4). One term of the integral from Eq. (13) over the normalized *template* domain $\mathcal{D}_t = \{\mathbf{x} | \mathbf{x}^T\Sigma^{-1}\mathbf{x} \le r^2\}$ is computed as follows

$$\int_{\mathcal{D}_t} \mathbf{x}\omega\big(\mathcal{P}_i(\mathbf{x})\big)d\mathbf{x} =$$

$$\int_{\mathcal{D}_t} \mathbf{x} \exp\Big(-\frac{1}{2}(\mathbf{x}-\mu_i)^T\Sigma_i^{-1}(\mathbf{x}-\mu_i)\Big)^n d\mathbf{x}. \quad (21)$$

The computation for $\mathcal{D}_o$ is the same with obvious substitutions. We can further simplify the integrand in Eq. (21) by translating the origin of the coordinate system into μ_i and diagonalizing

$$\Sigma_i^{-1} = \begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{12} & \sigma_{22} \end{bmatrix},$$

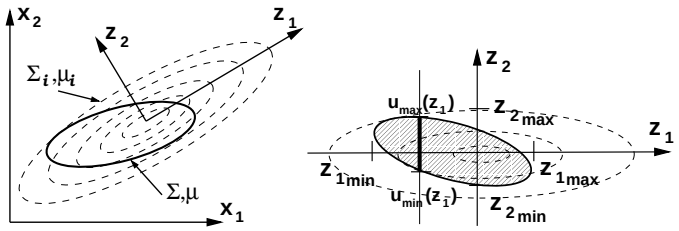
where μ_i and Σ_i are computed from the i^{th} input shape. The usual way to diagonalize a covariance matrix is by spectral decomposition. Unfortunately, it is inherently ambiguous as the applied rotation can be either α or $\alpha + \pi$ resulting in two different orientations of the coordinate system. In order to have a unique decomposition, we will diagonalize Σ_i^{-1} by a *shear transformation* $\mathbf{G}_i$: $\Sigma_i^{-1} = \mathbf{G}_i^T\mathbf{D}_i\mathbf{G}_i$, where

$$\mathbf{G}_i = \begin{bmatrix} 1 & \frac{\sigma_{12}}{\sigma_{11}} \\ 0 & 1 \end{bmatrix}, \text{ and } \mathbf{D}_i = \begin{bmatrix} \sigma_{11} & 0 \\ 0 & |\Sigma_i^{-1}|/\sigma_{11} \end{bmatrix}. \quad (22)$$

Actually, the diagonalization of Σ_i^{-1} results in a coordinate transformation (see Fig. 3) which in turn yields the following integral transformation in Eq. (21):

$$\mathbf{z} = \mathbf{G}_i(\mathbf{x} - \mu_i) \;\Rightarrow\; \mathbf{x} = \mathbf{G}_i^{-1}\mathbf{z} + \mu_i, \text{ and } d\mathbf{x} = |\mathbf{G}_i|^{-1}d\mathbf{z}. \quad (23)$$

Furthermore, let $\mathbf{z} = [z_1, z_2]^T$ and



(a) (b)

Fig. 3. We compute the integrals over an elliptic domain. (a) An example for the domain (bold line) and the integrated function (dashed lines) which is the ith *covariant function*. (b) In order to simplify the integral, we should translate the coordinate system to the mean of the ith object and diagonalize its covariance matrix.

$$\widehat{\mu} = \mathbf{G}_i\mu_i \quad \text{and} \quad \widehat{\Sigma}^{-1} = \mathbf{G}_i^{-T}\Sigma^{-1}\mathbf{G}_i^{-1} = \begin{bmatrix} \widehat{\sigma}_{11} & \widehat{\sigma}_{12} \\ \widehat{\sigma}_{12} & \widehat{\sigma}_{22} \end{bmatrix},$$

where $\widehat{\mu} = [\widehat{\mu}^{(1)}, \widehat{\mu}^{(2)}]^T$, yielding

$$\mathbf{x} = \mathbf{G}_i^{-1}\mathbf{z} + \mathbf{G}_i^{-1}\mathbf{G}_i\mu_i = \mathbf{G}_i^{-1}(\mathbf{z} - \widehat{\mu}).$$

The integration domain $\mathcal{D}_t$ in the new coordinate system becomes $\widehat{\mathcal{D}}_t = \{\mathbf{z} | (\mathbf{z}-\widehat{\mu})^T\widehat{\Sigma}^{-1}(\mathbf{z}-\widehat{\mu}) \le r^2\}$, because using $\Sigma^{-1} = \mathbf{G}_i^T\widehat{\Sigma}^{-1}\mathbf{G}_i$ we get

$$\mathbf{x}^T\Sigma^{-1}\mathbf{x} = (\mathbf{z}-\widehat{\mu})^T \underbrace{\mathbf{G}_i^{-T}\mathbf{G}_i^T}_{\mathbf{I}} \widehat{\Sigma}^{-1} \underbrace{\mathbf{G}_i\mathbf{G}_i^{-1}}_{\mathbf{I}}(\mathbf{z}-\widehat{\mu})$$
$$= (\mathbf{z}-\widehat{\mu})^T\widehat{\Sigma}^{-1}(\mathbf{z}-\widehat{\mu}).$$

Similarly, for $(\mathbf{x} - \mu_i)$ we get

$$(\mathbf{x}-\mu_i)^T\Sigma_i^{-1}(\mathbf{x}-\mu_i) = \mathbf{z}^T\mathbf{G}_i^{-T}\mathbf{G}_i^T\mathbf{D}_i\mathbf{G}_i\mathbf{G}_i^{-1}\mathbf{z} = \mathbf{z}^T\mathbf{D}_i\mathbf{z}.$$

Putting the above pieces together, Eq. (21) has the following form in the new coordinate system:

$$\int_{\mathcal{D}_t} \mathbf{x}\omega\left(\mathcal{P}_i(\mathbf{x})\right)d\mathbf{x} = \int_{\widehat{\mathcal{D}}_t} \mathbf{G}_i^{-1}(\mathbf{z}-\widehat{\mu}) \exp\Big(-\frac{n}{2}\mathbf{z}^T\mathbf{D}_i\mathbf{z}\Big)\,d\mathbf{z}. \quad (24)$$

Expanding the first term

$$\mathbf{G}_i^{-1}(\mathbf{z}-\widehat{\mu}) = \begin{bmatrix} 1 & -\frac{\sigma_{12}}{\sigma_{11}} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} z_1 - \widehat{\mu}^{(1)} \\ z_2 - \widehat{\mu}^{(2)} \end{bmatrix}$$
$$= \begin{bmatrix} z_1 - \frac{\sigma_{12}}{\sigma_{11}}z_2 + \frac{\sigma_{12}}{\sigma_{11}}\widehat{\mu}^{(2)} - \widehat{\mu}^{(1)} \\ z_2 - \widehat{\mu}^{(2)} \end{bmatrix} \quad (25)$$

and making use of the basic properties of integrals, it is easy to see that we only need to compute the following types of integrals ($k = 1, 2$):

$$c_k\int_{\widehat{\mathcal{D}}_t} z_k \exp\Big(-\frac{n}{2}\mathbf{z}^T\mathbf{D}_i\mathbf{z}\Big)d\mathbf{z}, \quad c_3\int_{\widehat{\mathcal{D}}_t} \exp\Big(-\frac{n}{2}\mathbf{z}^T\mathbf{D}_i\mathbf{z}\Big)d\mathbf{z}, \quad (26)$$

where $c_1 \in \{1, 0\}$, $c_2 \in \{-\frac{\sigma_{12}}{\sigma_{11}}, 1\}$, and $c_3 \in \{\frac{\sigma_{12}}{\sigma_{11}}\widehat{\mu}^{(2)} - \widehat{\mu}^{(1)}, -\widehat{\mu}^{(2)}\}$ are the coefficients of z_1, z_2, and the constants from Eq. (25). Now we will derive a closed form formula for computing the above integrals for $k = 1$. First, let us write explicitly the double integrals in Eq. (26). The bounds for the first variable z_1 are $z_{1_{\min}}$ and $z_{1_{\max}}$ (see also Fig. 3)

$$z_{1_{\min,\max}} = \frac{|\widehat{\Sigma}^{-1}|\widehat{\mu}^{(1)} \mp \sqrt{|\widehat{\Sigma}^{-1}|\widehat{\sigma}_{22}r^2}}{|\widehat{\Sigma}^{-1}|} = \widehat{\mu}^{(1)} \mp \sqrt{\frac{\widehat{\sigma}_{22}r^2}{|\widehat{\Sigma}^{-1}|}}.$$

$$c_1 \int_{z_{1_{\min}}}^{z_{1_{\max}}} z_1 \int_{u_{\min}(z_1)}^{u_{\max}(z_1)} \exp\left(-\frac{n}{2}\mathbf{z}^T\mathbf{D}_i\mathbf{z}\right) dz_2 dz_1 = \frac{c_1\sqrt{\pi}}{2C_1} \int_{z_{1_{\min}}}^{z_{1_{\max}}} z_1 \exp\left(-\frac{nd_{11}z_1^2}{2}\right) \left(\operatorname{erf}\left(C_1 u_{\max}(z_1)\right) - \operatorname{erf}\left(C_1 u_{\min}(z_1)\right)\right) dz_1 \quad (27)$$

$$c_2 \int_{z_{2_{\min}}}^{z_{2_{\max}}} z_2 \int_{v_{\min}(z_2)}^{v_{\max}(z_2)} \exp\left(-\frac{n}{2}\mathbf{z}^T\mathbf{D}_i\mathbf{z}\right) dz_1 dz_2 = \frac{c_2\sqrt{\pi}}{2C_2} \int_{z_{1_{\min}}}^{z_{1_{\max}}} z_2 \exp\left(-\frac{nd_{22}z_2^2}{2}\right) \left(\operatorname{erf}\left(C_2 v_{\max}(z_2)\right) - \operatorname{erf}\left(C_2 v_{\min}(z_2)\right)\right) dz_2 \quad (28)$$

$$c_3 \int_{z_{2_{\min}}}^{z_{2_{\max}}} \int_{v_{\min}(z_2)}^{v_{\max}(z_2)} \exp\left(-\frac{n}{2}\mathbf{z}^T\mathbf{D}_i\mathbf{z}\right) dz_1 dz_2 = \frac{c_3\sqrt{\pi}}{2C_2} \int_{z_{1_{\min}}}^{z_{1_{\max}}} \exp\left(-\frac{nd_{22}z_2^2}{2}\right) \left(\operatorname{erf}\left(C_2 v_{\max}(z_2)\right) - \operatorname{erf}\left(C_2 v_{\min}(z_2)\right)\right) dz_2 \quad (29)$$

The bounds for the second variable z_2 are functions of z_1 corresponding to a vertical slice of the ellipse $\widehat{\mathcal{D}}_t$ at z_1:

$$u_{\min,\max}(z_1) = \widehat{\mu}^{(2)} - \frac{\widehat{\sigma}_{12}(z_1 - \widehat{\mu}^{(1)}) \pm w_1}{\widehat{\sigma}_{22}},$$

where $w_1 = \sqrt{\widehat{\sigma}_{22}r^2 - |\widehat{\Sigma}^{-1}|(z_1 - \widehat{\mu}^{(1)})^2}$.

Now the first integral from Eq. (26) can be rewritten as Eq. (27), where $C_1 = \sqrt{nd_{22}/2}$ and $\operatorname{erf}(x) = (2/\sqrt{\pi})\int_0^x e^{-t^2}dt$. The erf function can be efficiency approximated by its Taylor sum. We thus reduced the original double integral to a single integral, which is easy and very fast to compute by a standard quadrature formula. We can derive similar formulas for the other two integrals in Eq. (26), as shown in Eq. (28)–(29), with $C_2 = \sqrt{nd_{11}/2}$. The only difference is that we change the order of integration variables and associated bounds will be

$$z_{2_{\min,\max}} = \widehat{\mu}^{(2)} \mp \sqrt{\frac{\widehat{\sigma}_{11}r^2}{|\widehat{\Sigma}^{-1}|}},$$

$$v_{\min,\max}(z_2) = \widehat{\mu}^{(1)} - \frac{\widehat{\sigma}_{12}(z_2 - \widehat{\mu}^{(2)}) \pm w_2}{\widehat{\sigma}_{11}}$$

with $w_2 = \sqrt{\widehat{\sigma}_{11}r^2 - |\widehat{\Sigma}^{-1}|(z_2 - \widehat{\mu}^{(2)})^2}$.

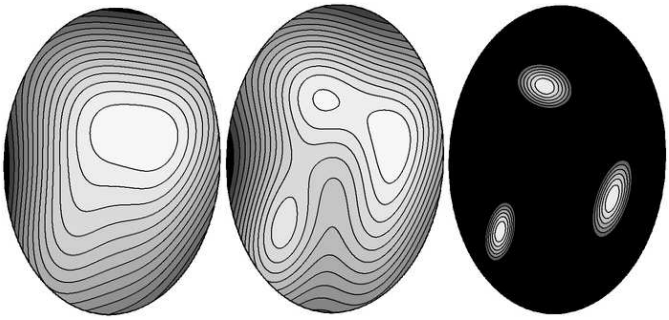
B. Choice of ω Functions

The closed form formulas Eq. (27)–(29) require that the applied set of $\{\omega_i\}_{i=1}^{\ell}$ functions be of the form

$$\omega_i : \mathbb{R} \to \mathbb{R} : \omega_i(x) = x^{n_i} \text{ with } n_i \in \mathbb{R}$$

While other choices are equally possible for ω_i (*e.g.* a trigonometric family has been successfully used in our previous work [29], [30]), only the above *power* functions allow a closed form computation of the integrals. A clear benefit of this numerical scheme is near-real time performance. We found empirically that the n^{th} power and n^{th} root functions with odd n, *i.e.* the set $\{x, x^3, x^5, \sqrt[3]{x}, \sqrt[5]{x}\}$ (see Fig. 4), provides satisfactory alignments in all of our test cases. It can be seen on Fig. 4 that the power functions concentrate the coloring to near by the center of mass of the shapes, producing sharp peaks, whereas the root functions have an opposite effect resulting in a smooth mixture surface.

The steps of the proposed algorithm are summarized in Algorithm 1. Note that the solution is obtained in a single pass without any loop or optimization. Although we have to compute the integrals and solve a linear system, the complexity of these steps are constant and, most importantly, independent of the image size. The images need to be scanned only once, when computing the means and covariance matrices of the



(a) $\omega(x) = \sqrt[5]{x}$ (b) $\omega(x) = \sqrt[3]{x}$ (c) $\omega(x) = x^3$

Fig. 4. The effect of the applied ωs on the compound shape from Fig. 1.

Algorithm 1: Pseudo-code of the proposed algorithm.

Input : *Template* and *observation* are labelled images ($\mathbb{I}_t$, $\mathbb{I}_o$); m = number of components

Output: Estimated transformation $(\tilde{\mathbf{A}}, \tilde{\mathbf{t}})$

Choose a set of exponents $\{n_i \in \mathbb{R} | i = 1 \dots \ell\}$

Initialize $r \in [1, 3]$

Normalize the input images by $(\mathbf{N}_t, \mathbf{t}_t)$ and $(\mathbf{N}_o, \mathbf{t}_o)$

`/*Constructing the system of Eq. (16): `$\mathbf{C}_o$` (resp. `$\mathbf{C}_t$`) denotes the coefficient matrix (resp. constants) on the left and right hand sides of the system. */`

$\{\mathbf{C}_t, |\Sigma|\} \leftarrow$ `Coeffs` $(\mathbb{I}_t, m, r, \{n_i\}_{i=1}^{\ell}, 2)$

$\{\mathbf{C}_o, |\Sigma'|\} \leftarrow$ `Coeffs` $(\mathbb{I}_o, m, r, \{n_i\}_{i=1}^{\ell}, 3)$

$\mathbf{C}_t \leftarrow \mathbf{C}_t\sqrt{|\Sigma'|/|\Sigma|}$

return

$$\left[\begin{bmatrix} \mathbf{N}_t & \mathbf{t}_t \\ \mathbf{0} & 1 \end{bmatrix}^{-1} \begin{bmatrix} (\mathbf{C}_o^+ \cdot \mathbf{C}_{t_{1..3,1}})^T \\ (\mathbf{C}_o^+ \cdot \mathbf{C}_{t_{1..3,2}})^T \\ [0\ 0\ 1] \end{bmatrix} \begin{bmatrix} \mathbf{N}_o & \mathbf{t}_o \\ \mathbf{0} & 1 \end{bmatrix}\right]^{-1}$$

Function `Coeffs` computes $\mathbf{C}_t$ and $\mathbf{C}_o$ for Algorithm 1.

Input : Labeled image; m = number of components; r; $\{n_i\}_{i=1}^{\ell}$; c = number of columns of the matrix $\mathbf{C}$
Output: $\mathbf{C}$, $|\Sigma|$

Set all element of $\mathbf{C} \in \mathbb{R}^{\ell \times c}$ to 0
Compute μ, and Σ over the whole foreground region
for $i \leftarrow 1$ **to** m **do**
    Compute μ_i and Σ_i of the i^{th} component
    /*Diagonalize Σ_i^{-1} */
    Compute $\mathbf{G}_i$, $\mathbf{D}_i$ based on Eq. (22)
    $\widehat{\Sigma}^{-1} \leftarrow \mathbf{G}_i^{-T}\Sigma^{-1}\mathbf{G}_i^{-1}$, $\widehat{\mu} \leftarrow \mathbf{G}_i\mu_i$
    /*Computing the coefficients */
    for $j \leftarrow 1$ **to** ℓ **do**
        $n \leftarrow n_j$
        $\mathbf{T} \leftarrow [Eq.\ (27), Eq.\ (28), Eq.\ (29)]$
        $\mathbf{C}_{j,1..c} \leftarrow \mathbf{C}_{j,1..c} + \mathbf{T}_{1..c}$
    end
end
return $\mathbf{C}$,$|\Sigma|$

Gaussian densities. Once these parameters are computed, the rest of the algorithm runs in constant time independently of the input size.

VI. Experimental results

The proposed algorithm has been implemented in Matlab 7.2 and all tests have been ran on a Pentium IV 3.2 GHz under Linux operating system. The benchmark datasets and the demo implementation of our method are available for download at http://www.inf.u-szeged.hu/~kato/software/.

For quantitative evaluation, we created several benchmark databases of $\approx$ 1500 synthetically generated observations of 60 different compound shapes. The applied transformations were randomly composed of $0°, 10°, \ldots, 350°$ rotations; $0, 0.4, \ldots, 1.2$ shearings; $0.5, 0.7, \ldots, 1.9$ scalings, and $-20, 0, 20$ translations along both axes. The resulting images are of size $\approx 1500 \times 1500$, some typical examples can be seen in Fig. 5. For the evaluation of registration results, we defined two kinds of error measure: The first one (denoted by ϵ) measures the average distance between the true $(\mathbf{A}, \mathbf{t})$ and the estimated $(\tilde{\mathbf{A}}, \tilde{\mathbf{t}})$ transformation, while the second one is the absolute difference (denoted by δ) between the *observation* and the *registered* image:

$$\epsilon = \frac{1}{|F_t|} \sum_{\mathbf{p} \in F_t} \|(\mathbf{A} - \tilde{\mathbf{A}})\mathbf{p}\|, \quad \text{and} \quad \delta = \frac{|F_r \triangle F_o|}{|F_r| + |F_o|} \cdot 100\%,$$

where $\triangle$ means the symmetric difference, while F_t, F_o, and F_r denote the set of pixels of the *template*, *observation*, and the *registered* shape. Note that ϵ can only be used when the true transformation $(\mathbf{A}, \mathbf{t})$ is also known, while δ can always be computed. On the other hand, ϵ gives a better characterization of the transformation error as it directly evaluates the misalignment, whereas δ sees only the percentage of non-overlapping area between the *observation* and *registered* shape. The performance of our algorithm on the benchmark dataset has been evaluated based on these measures. A summary of these results is presented in Table I.

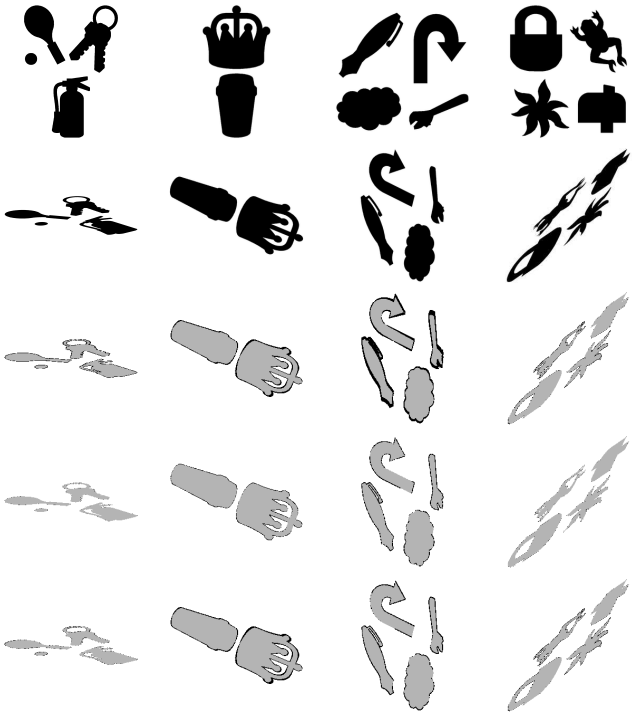


Fig. 5. Some typical registration results on the synthetic data set. The *template* and corresponding *observation* are shown in the first and second row, respectively. Subsequent rows contain the registration results provided by the method of Kannala *et al.* [13], Suk & Flusser [11], and the proposed method.

TABLE I
Median of error measures and runtime on 1435 images.

	Runtime (sec.)	ϵ (pixel)	δ (%)
Kannala *et al.* [13]	28.64	2.68	1.64
Suk & Flusser [11]	4.03	0.43	0.06
Proposed method	0.33	0.54	0.19

The proposed approach has also been compared to some recent binary registration approaches [11], [13]. One of the most closely related approach is proposed by Kannala *et al.* [13]. This method constructs a linear system of equations by basically looking at the images at 3 different scales, therefore the solution is inherently less precise as in each equation they can only use part of the available information. In contrast to [13], our approach always uses all the information available in the images. The method proposed by Suk & Flusser [11] computes normalization parameters based on image moments. The aligning transformation can then be directly computed from these parameters. We have obtained the Matlab implementation from the authors of both methods and conducted a comparative test on our benchmark datasets. The results presented in Table I and Fig. 5 show that our method outperforms these approaches in terms of computing time. Furthermore, in contrast to [13], it provides almost perfect alignments, while the registration quality of [11] is

slightly better at the price of a more than 10 times higher CPU time. On the other hand, our method clearly dominates [11] in terms of robustness as it is demonstrated in the next section.

A. Robustness

The robustness of the proposed algorithm has been analyzed against missing data, segmentation as well as modelling errors. Besides using real images inherently subject to such errors, we have also conducted a systematic test on simulated data: In the first testcase, $10\%, \ldots, 90\%$ of the foreground pixels has been removed from the *observation* before registration to simulate *missing data*. In the second case, square-shaped regions of a total size $1\%, \ldots, 5\%$ of the shape have been randomly added to or removed from the boundary of foreground regions to simulate *segmentation error*. Note that we do not include cases where erroneous foreground regions appear as disconnected regions, because such false regions can be efficiently removed by appropriate morphological filtering. We therefore concentrate on cases where segmentation errors cannot be filtered out. Another issue is *modelling error*. Obviously, the method would fail for anything which is far from the affine case. It is however expected to be robust when an affine deformation model is a good *approximation* to the true transformation. This will definitely undermine the identity relation in Eq. (1) yielding an error in our system. Therefore our third testcase consists in adding a zero mean Gaussian noise with σ deviation to each pixel coordinate (basically a random displacement), resulting in a non-linear deformation. See samples of these errors in Fig. 6.

Table II summarizes the achieved performance with respect to the above three cases. Considering the fact that a $\delta < 6\%$ corresponds to a visually acceptable alignment, our method proved to be robust in all three cases, although at different levels: It can tolerate up to as high as 50% missing pixels, 2.5% segmentation error, and a $\sigma = 2.5$ modelling error. Note also that [13] is consistently outperformed by the proposed approach in all testcases, whereas [11] is quite sensitive to missing data and fails completely for modelling errors, but it is robust against segmentation errors.

Finally, we remark that occlusion yields a rather high error rate for both the proposed as well the other state-of-the-art methods. This is because they are relying on quantities obtained by integrating over the whole object area. Thus large missing parts would drastically change these quantities resulting in false registrations. Nevertheless, in many application areas one can take images under controlled conditions which guarantees that observations are not occluded (*e.g.* medical imaging, industrial inspection).

B. Real Images

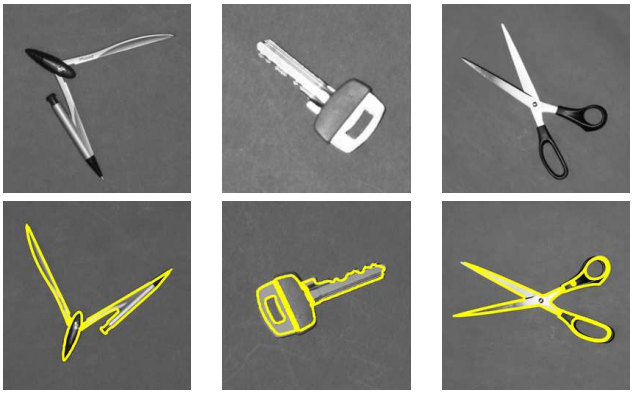
The performance of our method has also been evaluated on real images. Fig. 7 shows some examples of these images. The segmentation was performed via thresholding. The main challenges are segmentation errors and slight projective distortion between the image pairs. In summary, when reasonably good segmentations are available and the true transformation is close enough to an affine one then our method performs quite well, as it is shown by the δ error values displayed below each image pair.

(a) Missing data (b) Segmentation error (c) Modelling error

Fig. 6. Sample observations for testing robustness. In c) the true contour is overlayed in blue.

TABLE II
MEDIAN OF ERROR MEASURES VS. VARIOUS TYPE OF ERRORS.

	Kannala *et al.* [13]		Suk & Flusser [11]		Proposed	
Missing data (% of removed pixels)						
%	ϵ	δ	ϵ	δ	ϵ	δ
10	3.66	2.38	19.28	9.86	2.16	1.26
50	7.65	4.71	110.23	50.54	6.06	3.67
90	26.22	14.75	258.12	92.2	18.33	10.04
Segmentation error (size of randomly added/removed squares in %)						
%	ϵ	δ	ϵ	δ	ϵ	δ
1	5.84	3.85	1.65	0.88	4.97	2.96
2.5	10.77	6.67	3.21	1.74	9.27	5.27
5	18.37	11.06	5.68	3.08	17.96	9.86
Modelling error (σ of random displacements)						
σ	ϵ	δ	ϵ	δ	ϵ	δ
1	6.19	3.96	73.11	34.86	5.19	3.08
2.5	12.74	7.82	75.09	35.67	10	5.7
5	27.33	16.73	73.63	35.02	18.44	10.48



$\delta = 2.64\%$ $\delta = 5.55\%$ $\delta = 9.42\%$

Fig. 7. Registration results on real images. **Top**: the images used as *templates*. **Bottom**: the corresponding *observations* with the overlayed contour of the registration results.

a) Registration of Traffic Signs: Nowadays, modern cars include many safety systems. Automatic traffic sign recognition is a major challenge of such intelligent systems, where one of the key tasks is the real-time matching of an observed sign with its *template*. Fig. 8 shows typical registration results on such images. Herein, we have used classical thresholding and some morphological operations for segmentation but automatic detection/segmentation is also possible [38]. Note that these images are inherently corrupted by several type of error:

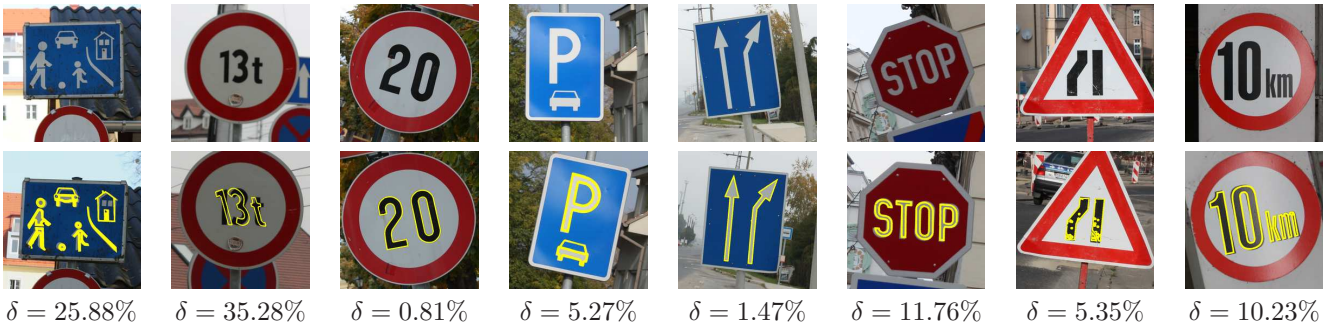


Fig. 8. Registration results on traffic signs. **Top**: the images used as *templates*. **Bottom**: the corresponding *observations* with the overlayed contour of the registration results

First, the true transformation is *projective*, but -since signs has to be detected and recognized from a larger distance- the affine model is a valid assumption here. Second, thresholding produced imperfect segmentations in the presence of specular reflections. Finally, various surface errors resulted in missing data inside foreground regions (see the last two images in Fig. 8). In spite of these problems, our method performs quite well when reasonably good segmentations are available and the true transformation is close enough to an affine one.

VII. Conclusions

In this paper, we have presented a novel approach for planar shape alignment. The fundamental difference compared to classical image registration algorithms is that our model works without any landmark, feature detection or optimization by adopting a novel idea where the transformation is obtained as a solution of a set of linear equations. The complexity of the algorithm is linear and, by adopting an efficient numerical scheme, it is capable of registering images at near real-time speed. Experimental results show that the proposed method provides good alignments on both real and synthetic images. Furthermore, its robustness has been demonstrated against missing data, segmentation and modelling errors. In general, our method will perform well as long as the first and second order statistics of shapes do not change dramatically, therefore its superiority can be fully exploited in applications where occlusion can be kept at a minimum (*e.g.* medical imaging or industrial inspection). Comparative tests show the efficiency and accuracy of our model compared to state-of-the-art methods.

References

[1] B. Zitová and J. Flusser, "Image registration methods: A survey," *Image and Vision Computing*, vol. 21, no. 11, pp. 977–1000, Oct. 2003.

[2] J. Salvi, C. Matabosch, D. Fofi, and J. Forest, "A review of recent range image registration methods with accuracy evaluation," *Image and Vision Computing*, vol. 25, no. 5, pp. 578–596, May 2007.

[3] J. B. A. Maintz and M. A. Viergever, "A survey of medical image registration," *Medical Image Analysis*, vol. 2, no. 1, pp. 1–36, Mar. 1998.

[4] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape context," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 509–522, Apr. 2002.

[5] P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, Feb. 1992.

[6] A. W. Fitzgibbon, "Robust registration of 2D and 3D point sets," *Image and Vision Computing*, vol. 21, no. 13, pp. 1145–1153, Dec. 2003.

[7] S. Mann and R. W. Picard, "Video orbits of the projective group a simple approach to featureless estimation of parameters," *IEEE Transactions on Image Processing*, vol. 6, no. 9, pp. 1281–1295, Sept. 1997.

[8] P. M. Q. Aguiar, "Unsupervised simultaneous registration and exposure correction," in *Proceedings of International Conference on Image Processing*. Atlanta, GA, USA: IEEE, Oct. 2006, pp. 361–364.

[9] R. Hagege and J. M. Francos, "Parametric estimation of multi-dimensional affine transformations:an exact linear solution," in *Proceedings of International Conference on Acoustics, Speech, and Signal Processing*, vol. 2. Philadelphia, USA: IEEE, Mar. 2005, pp. 861–864.

[10] C. Domokos and Z. Kato, "Parametric estimation of affine deformations of planar shapes," *Pattern Recognition*, vol. 43, no. 3, pp. 569–578, Mar. 2010.

[11] T. Suk and J. Flusser, "Affine normalization of symmetric objects," in *Proceedings of International Conference on Advanced Concepts for Intelligent Vision Systems*, ser. Lecture Notes in Computer Science, J. Blanc-Talon, W. Philips, D. Popescu, and P. Scheunders, Eds., vol. 3708. Antwerp, Belgium: Springer, Sept. 2005, pp. 100–107.

[12] J. Flusser and T. Suk, "A moment-based approach to registration of images with affine geometric distortion," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 32, no. 2, pp. 382–387, Mar. 1994.

[13] J. Kannala, E. Rahtu, J. Heikkilä, and M. Salo, "A new method for affine registration of images and point sets," in *Proceedings of Scandinavian Conference on Image Analysis*, ser. Lecture Notes in Computer Science, vol. 3540, Pattern Recognition Society of Finland. Joensuu, Finland: Springer, June 2005, pp. 224–234.

[14] J. Heikkilä, "Pattern matching with affine moment descriptors," *Pattern Recognition*, vol. 37, no. 9, pp. 1825–1834, Sept. 2004.

[15] Y. Zhang, C. Wen, and Y. Zhang, "Recognition of symmetrical images using affine moment invariants in both frequency and spatial domains," *Pattern Analysis & Applications*, vol. 5, no. 3, pp. 316–325, Aug. 2002.

[16] J. Kim and J. A. Fessler, "Intensity-based image registration using robust correlation coefficients," *IEEE Transactions on Medical Imaging*, vol. 23, no. 11, pp. 1430–1444, Nov. 2004.

[17] A. Kadyrov and M. Petrou, "Affine parameter estimation from the trace transform," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 10, pp. 1631–1645, Oct. 2006.

[18] V. Govindu and C. Shekhar, "Alignment using distributions of local geometric properties," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 10, pp. 1031–1043, Oct. 1999.

[19] J. Ho, M.-H. Yang, A. Rangarajan, and B. Vemuri, "A new affine registration algorithm for matching 2D point sets," in *Proceedings of IEEE Workshop on Applications of Computer Vision*. Austin, Texas, USA: IEEE, Feb. 2007, pp. 25–31.

[20] E. Rahtu, M. Salo, J. Heikkilä, and J. Flusser, "Generalized affine moment invariants for object recognition," in *Proceedings of International Conference on Pattern Recognition*, vol. 2. Hong-kong: IEEE, Aug. 2006, pp. 634–637.

[21] O. Musse, F. Heitz, and J.-P. Armspach, "Topology preserving deformable image matching using constrained hierarchical parametric models," *IEEE Transactions on Image Processing*, vol. 10, no. 7, pp. 1081–1093, July 2001.

[22] S. Manay, D. Cremers, B.-W. Hong, A. J. Yezzi Jr., and S. Soatto, "Integral invariants for shape matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 10, pp. 1602–1618, Oct. 2006.

[23] F. Hjouj and D. W. Kammler, "Identification of reflected, scaled,

translated, and rotated objects from their Radon projections," *IEEE Transactions on Image Processing*, vol. 17, no. 3, pp. 301–310, Mar. 2008.

[24] K. M. Simonson, S. M. Drescher, and F. R. Tanner, "A statistics-based approach to binary image registration with uncertainty analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, pp. 112–125, Jan. 2007.

[25] G. McNeill and S. Vijayakumar, "Hierarchical procrustes matching for shape retrieval," in *Proceedings of Computer Vision and Pattern Recognition*, B. Werner, Ed., vol. 1. New York, NY, USA: IEEE, June 2006, pp. 885–894.

[26] B. Jian and B. C. Vemuri, "A robust algorithm for point set registration using mixture of gaussians," in *Proceedings of IEEE International Conference on Computer Vision*, vol. 2. Beijing, China: IEEE, Oct. 2005, pp. 1246–1251.

[27] A. Myronenko, X. Song, and M. A. Carreira-Perpiñán, "Non-rigid point set registration: Coherent point drift," in *Proceedings of Conference on Neural Information Processing Systems*, B. Schölkopf, J. Platt, and T. Hoffman, Eds. Vancouver, Canada: MIT Press, Dec. 2006, pp. 1009–1016.

[28] R. Hagege and J. M. Francos, "Linear estimation of sequences of multi-dimensional affine transformations," in *Proceedings of International Conference on Acoustics, Speech, and Signal Processing*, vol. 2. Toulouse, France: IEEE, May 2006, pp. 785–788.

[29] C. Domokos and Z. Kato, "Binary image registration using covariant Gaussian densities," in *Proceedings of International Conference on Image Analysis and Recognition*, ser. Lecture Notes in Computer Science, A. Campilho and M. Kamel, Eds., vol. 5112. Póvoa de Varzim, Portugal: Springer, June 2008, pp. 455–464.

[30] ——, "Affine alignment of compound objects: A direct approach," in *Proceedings of International Conference on Image Processing*, IEEE. Cairo, Egypt: IEEE, Nov. 2009, pp. 169–172.

[31] A. M. Bronstein, M. M. Bronstein, R. Kimmel, M. Mahmoudi, and G. Sapiro, "A gromov-hausdorff framework with diffusion geometry for topologically-robust non-rigid shape matching," *International Journal of Computer Vision*, vol. 89, no. 2-3, pp. 266–286, Sept. 2010.

[32] D. Raviv, A. M. Bronstein, M. M. Bronstein, and R. Kimmel, "Full and partial symmetries of non-rigid shapes," *International Journal of Computer Vision*, vol. 89, no. 1, pp. 19–39, Aug. 2010.

[33] K. S. Arun, T. S. Huang, and S. D. Blostein, "Least-squares fitting of two 3-D point sets," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 9, no. 5, pp. 698–700, Sept. 1987.

[34] S. Umeyama, "Least-squares estimation of transformation parameters between two point patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 4, pp. 376–380, Apr. 1991.

[35] M.-K. Hu, "Visual pattern recognition by moment invariants," *IRE Transactions on Information Theory*, vol. 8, no. 2, pp. 179–187, Feb. 1962.

[36] A. Foulonneau, P. Charbonnier, and F. Heitz, "Multi-reference shape priors for active contours," *International Journal of Computer Vision*, vol. 81, no. 1, pp. 68–81, Jan. 2009.

[37] M. R. Teague, "Image analysis via the general theory of moments," *Journal of the Optical Society of America*, vol. 70, pp. 920–930, Aug. 1980.

[38] M. S. Prieto and A. R. Allen, "Using self-organising maps in the detection and recognition of road signs," *Image and Vision Computing*, vol. 27, no. 6, pp. 673–683, May 2009.

Csaba Domokos (SM'07) received the M.Sc. degrees in computer science from the University of Szeged, Hungary in 2006. He is currently pursuing the Ph.D. degree in the Institute of Informatics at University of Szeged.

His current research interests include image registration, image transformations and shape matching.

Zoltan Kato received the M.S. degree in Computer Science from the University of Szeged, Hungary in 1990, and the Ph.D. degree from the University of Nice doing his research at INRIA Sophia Antipolis, France in 1994. Since then, he has been a visiting research associate at the Computer Science Department of the Hong Kong University of Science & Technology, Hong Kong; an ERCIM postdoc fellow at CWI, Amsterdam, The Netherlands; and a visiting fellow at the School of Computing, National University of Singapore, Singapore. Currently, he is head of the Department of Image Processing and Computer Graphics at the University of Szeged, Hungary. He was an Associate Editor of IEEE TRANSACTIONS ON IMAGE PROCESSING from 2003 to 2008. He is a Senior Member of the IEEE and IEEE Signal Processing Society. His research interests include statistical image models, MCMC methods, and shape modeling.