

Appearance modeling under geometric context*

Jian Li¹, Shaohua Kevin Zhou², and Rama Chellappa¹

¹Center for Automation Research & ECE Department
University of Maryland, College Park, MD 20742

²Integrated Data Systems Department
Siemens Corporate Research, Princeton, NJ 08540

Abstract

We propose a unified framework based on a general definition of geometric transform (GeT) for modeling appearance. GeT represents the appearance by applying designed functionals over certain geometric sets. We show that image warping, Radon transform, trace transform, etc. are special cases of our definition. Moreover, three different types of GeTs are designed to handle deformation, articulation and occlusion and applied to fingerprinting the appearance inside a contour. They include the contour-driven GeT, the feature curve based GeT and selecting functionals to model the appearance inside the convex hull of the contour. A multi-resolution representation that combines both shape and appearance information is also proposed. We apply our approach to image synthesis and object recognition. The proposed approach produces promising results when applied to fingerprinting the appearance of human and body parts despite the challenges due to articulated motion and deformations.

1. Introduction

We present a generic approach for incorporating geometric prior information, or the so called *geometric context*, into appearance modeling. The context can be based on a model, inference from the contour, or prior knowledge of the motion, etc. In particular, we want to model the appearance inside a contour, such as the appearance of humans with articulated motion. In this case, objects may have very large deformations and self-occlusion, so simple transformations such as 2D affine cannot generate correct pixel-wise correspondences. We aim at relieving the burden on the machine learning algorithm to deal with such variations and improving the recognition rate by incorporating geometric context.

The most comprehensive way to incorporate geometric context is to have a full 3D generative model, which has been used successfully in face recognition [2]. However this approach is computationally intensive and requires too much prior information. A simplified 2D model is found in Active Appearance Model (AAM) [3], where a set of feature points are tracked and used to generate a *normalized*

appearance in the mean shape based on image warping. It requires feature points and can only deal with small deformations that obey a Gaussian distribution, so it is unapplicable for tasks like pedestrian recognition. Elastic Graph Matching (EGM) [9] is also a popular method to extract an appearance signature with some prior knowledge of the geometric structure. Similar to AAM, it requires feature points and an explicit model. In [12], the trace transform is used to generate invariant features with respect to a group of affine transformations. Their methods handle 2D rigid motion very well but have limited use for non-linear deformations. It has the advantage of not using any model, but does not have the capacity to include complicated geometric context.

In this paper, we take a transform based approach and provide a general framework that models the object with large deformations. When modeling the appearance inside a contour, we also look for implicit models that can be generated from the contour itself. In a word, the transform is used to represent visual patterns properly before recognizing them. Four major contributions are made.

(1) A unifying framework is proposed in section 2 based on a generally defined *Geometric Transform* (GeT). Section 2 also shows that image warping [17], Radon transform (RT) [4, 7, 8], and trace transform [12] are special instances of GeT. Our method is very flexible and does not require an explicit model or correspondences of feature points.

(2) We propose in sections 3,4 and 5 different ways and strategies of designing geometric sets and functionals used in GeT. In particular, we propose selection schemes based on the contour itself to handle large deformations and articulated motion. Going beyond the feature points based warping used in AAM, we also use feature curves/skeletons to find point-to-point correspondences. A functional that handles occlusion is also introduced.

(3) A multi-resolution version of GeT is proposed in section 6 and used for the combined representation of shape and appearance information at different scales.

(4) GeT is a very useful tool to obtain deformation invariant appearance models. In this paper we focus on obtaining the signature of the appearance inside a contour, which is essentially a 2D function with compact support. Section 7 shows the application of our approach for recognition of

*Most of this work was done when Zhou was at UMD. E-mails: {lij,rama}@cfar.umd.edu, kzhou@scr.siemens.com.

human and body parts under articulation.

2. Geometric transform

In this section, we present a framework for appearance modeling under geometric context. Our unifying definition includes as a motivating example the Radon transform [7, 8]

$$\mathbf{R}(\theta, p) = \int \int I(x, y) \delta(x \cos \theta + y \sin \theta - p) dx dy, \quad (1)$$

which applies an integral operation to image $I(x, y)$ along each line. RT carries two important elements: the geometric sets of straight lines, and the functionals defined over those sets, which are integrals. Through arbitrary choices of those two elements, we provide a general definition of geometric transform.

2.1. A unifying definition

Definition: Given any set $S \subset R^p$ and any function defined over the set S , that is $f : S \mapsto R^q$, a *geometric functional* G_S is a functional that takes as input the f value over the set S , that is $G_S : f \mapsto R^r$. We call S a *geometric set*. If we have a collection of sets $\{S(\theta)\}$ parameterized or indexed by θ , where each $S(\theta) \subset R^p$ is a geometric set, the *geometric transform* of the function $f : R^p \mapsto R^q$ is the mapping of $\{S(\theta)\}$ to R^r by applying $G_{S(\theta)}(\cdot)$ to f , i.e.,

$$\mathbf{R}(\theta) = \mathbf{R}(S(\theta)) = G_{S(\theta)}(f). \quad (2)$$

Because \mathbf{R} depends both on $\{S(\theta)\}$ and f , the definition in Eq. (2) is two-fold. If $\{S(\theta)\}$ is fixed, \mathbf{R} is the GeT of f . If we fix f , \mathbf{R} is the mapping of $\{S(\theta)\}$ to R^r . The parameter θ can be viewed as the coordinate in the transform domain.

In GeT, each set is associated with a vector (or a number when $r = 1$) that depends on the functional $G(\cdot)$ operating on a function f over set S . Ideally the vector is the signature of function f . The geometric context is embedded in the selection of sets $\{S(\theta)\}$. We provide details of how to select the sets in the following sections. Functional $G(\cdot)$ is chosen to generate the desired statistics in the set $S(\theta)$.

Note that the transform can be regarded as either the mapping $\mathbf{R} : \{S(\theta)\} \rightarrow R^r$ or the mapping $\mathbf{R} : \theta \rightarrow R^r$ according to function f and functional $G_{S(\theta)}(\cdot)$. In most cases of this paper, f is chosen as the image intensity defined on a compact region $\Omega \subset R^2$, which is usually inside a contour. We call f the appearance inside Ω , sometimes denoted as A . However, the domain of interest in Eq. (2) is not limited to an image plane which lies in R^2 . It can be generalized to $x - y - t$ plane in spatial-temporal domain or $x - y - n$ domain where n refers to the index of cameras when we have multiple cameras.

2.2. Special instances

We now show that many existing transforms and methods are special cases of the general definition in Eq. (2).

2.2.1 Radon transform and trace transform

Now we take a close look at the relation between RT and GeT. In n -dimensional Radon transform [4, 7, 8], the collection of sets $S(\theta)$ are hyperplanes parameterized by \mathbf{n} and p , such that $S(\theta) = \{x \in R^n | \mathbf{x}^T \mathbf{n} - p = 0\}$. So $\theta = \{\mathbf{n}, p\}$. The functional G is an integral operating on the set S .

$$\mathbf{R}(\mathbf{n}, p) = G_{S(\mathbf{n}, p)}(f) = \int_S f(\mathbf{x}) d\mathbf{x} = \int_{\Omega} f(\mathbf{x}) \delta(\mathbf{x}^T \mathbf{n} - p) d\mathbf{x} \quad (3)$$

RT has been well studied in computer tomography (CT) [8]. The image can be fully reconstructed from its RT using the filtered back-projection algorithm according to Fourier slice theorem. In computer vision, the basic use of RT has been for line and shape detections, which is also referred to as Hough transform [5].

By changing the functionals defined on the line set, RT is generalized to the trace transform. In 2D trace transform, the geometric set remains straight lines. Denoting points $\mathbf{x} = (x, y)$ in line set $S(\mathbf{s}, p)$ as

$$x = ps_1 + ts_2, y = ps_2 - ts_1, -\infty < t < +\infty,$$

the function $f(\mathbf{x})$ becomes a function of t , $f(t)$, for $\mathbf{x} \in S$. Some examples of geometric functional G include

$$G_S(f) = \left(\int |f(t)|^q dt \right)^r, G_S(f) = \frac{\int t f(t) dt}{\int f(t) dt},$$

and so on. Different choices of functional G correspond to different statistics. In [12], the authors focus on designing the combinations of functionals to extract affine-invariant features, which is very useful for recognition of appearance inside a contour. However because they limit the geometric sets to be straight lines, their methods lack the ability to model appearance for objects with large non-rigid motions.

2.2.2 Image warping

Consider the case when point sets are selected as the geometric sets, i.e., $S(\theta) = \{\mathbf{x}\}$. If the functional is an identity mapping, i.e., $G_{S(\theta)}(f) = f(\mathbf{x})$, then the definition in Eq. (2) generalizes the traditional definition of geometric transformation of images [17], which includes affine transformation or perspective transformation etc. In that case, θ can simply be the new coordinate in the transform domain, say, $\theta = (\tilde{x}, \tilde{y})$, and the transformation of coordinate system is implemented in the typically one-to-one mapping $S(\theta) = S((\tilde{x}, \tilde{y})) = \{(x, y)\}$. For example, if

$$\begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}, \quad (4)$$

then the GeT becomes a 2D affine transformation. More complicated image warping can also be included in GeT

such as the feature point based warping used in AAM [3]. First a set of feature points are registered, i.e., in terms of geometric set, $S(\tilde{\mathbf{x}}_p^c) = \{\mathbf{x}_p^c\}$ for known $\{\mathbf{x}_p^c\}_{p=1}^P$ and $\{\tilde{\mathbf{x}}_p^c\}_{p=1}^P$. Then an equivalent GeT to image warping uses point sets

$$S(\tilde{\mathbf{x}}) = \left\{ \sum_{i=1}^P h_i(\tilde{\mathbf{x}}) \mathbf{x}_i \right\}, \quad (5)$$

where $h_i(\cdot)$ is an interpolation kernel that satisfies $h_i(\tilde{\mathbf{x}}_j) = 1$ for $i = j$, $h_i(\tilde{\mathbf{x}}_j) = 0$ for $i \neq j$. In [3], the authors show that by properly choosing h_i , the warping can be piecewise affine or thin plate splines. So, AAM uses the feature points based GeT.

2.3. Designing GeT for appearance modeling

As seen in section 2.2, the geometric set is a very crucial aspect of generalization from Eq. (3), which can be written as

$$\mathbf{R}(S(\theta)) = \int f(\mathbf{x}) \chi_{S(\theta)}(\mathbf{x}) d\mathbf{x}, \quad (6)$$

where the integral takes place in an arbitrary set $S(\theta)$ and χ is the indicator function whether \mathbf{x} belongs to $S(\theta)$.

All the geometric information is embedded in the definition of $S(\theta)$ in Eq. (2), which essentially reflects correspondences inferred from geometric context. As seen in image warping, coordinate changes and point-to-point mapping are required. But in GeT, S can be arbitrarily selected to reflect the correspondence of curves or regions. In the case of modeling appearance inside two contours, our focus is on finding a transform domain representation that is invariant to the relative motion between the two contours.

Several typical kinds of motions studied in this paper and their preferred GeTs are as follows:

Bending: Such as human arms in Figure 1. GeT based on level set can be used as described in 3.1.

Local deformation: For small local deformations, a multi-resolution GeT discussed in section 6 can be applied. For larger ones, we can apply a GeT based on shape matching in section 3.2 or AAM [3] with a set of feature points to generate the point sets as discussed in section 2.2.2.

Articulated motion of parts: Objects with articulated motion (walking human) can be divided into parts. The geometric set can be generated from those segmented parts or from a skeleton model in section 4.

Other possible ways of finding the geometric set can be through an analysis of appearance such as color-based segmentation, from temporal dynamics such as using a motion model, or from multi-view relations when we have multiple cameras. They are beyond the scope of this paper.

We discuss generating the set from the contour itself and using feature curves in sections 3 and 4, respectively. A functional helpful to handle occlusions is proposed in section 5.

3. Contour-driven GeT

An interesting way of generating the set is from the contour itself. Neither explicit models nor feature points are required. Both curve sets and point sets can be generated from the contour as follows.

3.1. GeT based on level set

Suppose we have a level set representation of the contour $\phi(\mathbf{x})$. $\phi(\mathbf{x})$ can be obtained from the signed distance transform [11], or from Poisson equations [6]. The geometric set can be generated from the level set function as follows:

$$\mathbf{R}(c) = \int f(\mathbf{x}) \delta(\phi(\mathbf{x}) - c) d\mathbf{x}. \quad (7)$$

Since inside the contour $\phi(\mathbf{x}) > 0$, then for $c > 0$, the integral is over the level set inside the contour. One can easily see that this transform is translation and rotation invariant, and it can be easily made scale invariant as well. Not only that, it is not sensitive to bending of the contour. See Fig. 1.

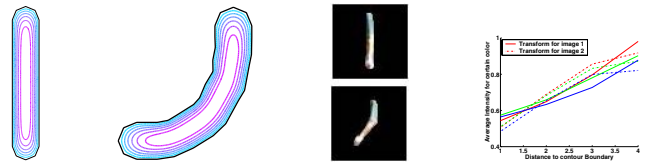


Figure 1: Illustration of contour-driven set applied to bending. Images 1 and 2: using the level set as the geometric set makes it insensitive to bending. Images 3 and 4: two arm images, and average intensity of each color along each level set for the two images of arms.

We show that this selection of sets is particularly useful for modeling the appearance of a single component contour with bending and small distortions, such as modeling human arms in section 7.1. In Fig. 1, we plot the curves of $\mathbf{R}(c)$ displaying the average intensity of arm images along different levels c . We observe that these curves are clustered together. This indicates that our transform is somewhat insensitive to bending.

3.2. GeT based on shape matching

Point sets can be generated through matching between two contours. In this case the mapping $S(\tilde{\mathbf{x}}) = \{\mathbf{x}\}$ reflects the dense point-to-point correspondences inferred from shape matching. Region-based shape matching [16] can be directly applied to infer the correspondences inside two contours. But here we focus on the contour based shape matching, among which a descriptor called shape context raises the benchmark in [1][10] and finds correspondences without feature points. We formulate the idea into a GeT and apply it to modeling the appearance of pedestrians, where the objects have articulated motion of parts and self occlusions.

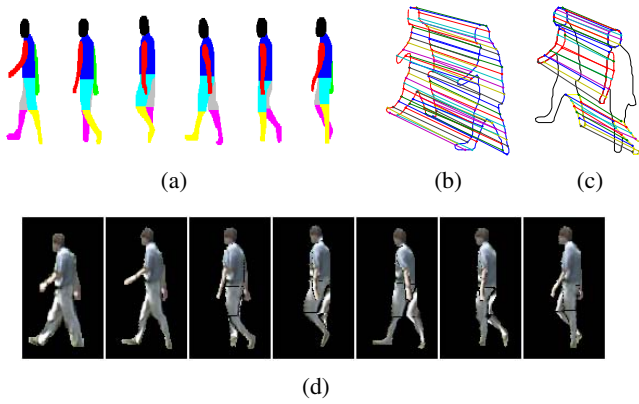


Figure 2: (a) Canonical silhouettes of six typical poses of a walking human, along with segmentation of body parts. Taken from USF database [13]. (b) Shape matching between a pedestrian’s silhouette and the canonical silhouette at a similar pose. The corresponding points are used for GeT $\mathbf{R}_{\Gamma_0\Gamma_j}$. (c) Shape matching between parts at different poses used to generate GeT $\mathbf{R}_{\gamma_j^k\gamma_i^k}$. Here we show the matching of head, left arm and left lower leg. By applying GeT to each part in a certain order, the appearance can be transformed from one pose to another. (d) First image is the sample image, followed by synthesized normalized appearance at six typical poses. Second image shows the synthetic image at the closest pose.

A GeT based on shape matching is defined as follows: suppose we have two 2D regions Γ_0, Γ_1 bounded by two contours C_0 and C_1 respectively. Denote the intensity function in region Γ_0 as A_0 . The two contours are represented by sampled points on them, i.e., $C_0 = \{\mathbf{x}_i^c | i = 1, \dots, N_0\}$, $C_1 = \{\tilde{\mathbf{x}}_i^c | i = 1, \dots, N_1\}$. Then by applying any shape matching method to the two sets of points, one-to-one mapping of their subsets are found as $\tilde{\mathbf{x}}_i^m \leftrightarrow \mathbf{x}_i^m$, for $i = 1, \dots, M$ and $M \leq \min(N_0, N_1)$. Design geometric sets for interpolated dense correspondences as

$$S(\tilde{\mathbf{x}}) = \left\{ \sum_{i=1}^M h_i(\tilde{\mathbf{x}}) \mathbf{x}_i^m \right\}, \quad (8)$$

for $\mathbf{x} \in \Gamma_0, \tilde{\mathbf{x}} \in \Gamma_1$ and $h_i(\cdot)$ satisfies $h_i(\tilde{\mathbf{x}}_j^m) = 1$ for $i = j$, $h_i(\tilde{\mathbf{x}}_j^m) = 0$ for $i \neq j$ as shown in section 2.2.2. Identity mapping are used as functionals over those sets. Then the corresponding GeT, denoted as $\mathbf{R}_{\Gamma_0\Gamma_1}$, is the GeT of the function A_0 based on shape matching between Γ_0 and Γ_1 . $\mathbf{R}_{\Gamma_0\Gamma_1}$ transforms the appearance A_0 inside contour C_0 to appearance A_1 inside contour C_1 . Although here we still use point based image warping as in Eq. (5), \mathbf{x}_i^m s are not necessarily points at places with distinctive features such as corners or points with large curvatures.

Now we show how to use this GeT to obtain a *pose-invariant* representation of a pedestrian’s appearance. Suppose two images of pedestrians are to be matched after

background subtraction. It is hard to compare them directly because of differences in poses and sizes of the silhouettes. However, if we focus on the side view of the person, a walking human usually has six typical poses as in Fig. 2(a). Although each person may have different shape and walk differently, the topology of body parts remains roughly the same for different people at the same pose. We can use this property to *normalize* appearances at the same pose. By *normalization* we mean warping the appearance to be inside a canonical shape through GeT for pixel-wise comparison. This way, shape variations of different people are handled similarly to obtaining a *normalized* appearance of a face inside a mean shape in AAM.

So given only one image of a pedestrian with an arbitrary pose, we can obtain the *normalized* appearance of pedestrians at all other poses as illustrated in Fig. 2(d), by using the GeT based on shape matching. We assume to have canonical silhouettes at six typical poses $\{\Gamma_i | i = 1, \dots, 6\}$ as shown in Fig. 2(a), along with the eight-part segmentation $\{\gamma_i^k | i = 1, \dots, 6, k = 1, \dots, 8\}$. We first *normalize* his appearance inside the canonical silhouette for the closest pose, before synthesizing his *normalized* appearance at other poses. Denote the intensity inside the pedestrian’s silhouette Γ_0 as A_0 . Here are the two steps,

(1) Use shape matching to find the most similar pose: $j = \operatorname{argmin}_{i=1, \dots, 6} \operatorname{MatchCost}(\Gamma_0, \Gamma_i)$. Use GeT $\mathbf{R}_{\Gamma_0\Gamma_j}$ to find normalized appearance A_j inside Γ_j , as illustrated in Fig. 2(b).

(2) Synthesize from pose j to pose i . For body part k , transfer the appearance a_j^k inside γ_j^k to appearance a_i^k inside γ_i^k by applying $\mathbf{R}_{\gamma_j^k\gamma_i^k}$ to a_j^k , as illustrated in Fig. 2. Transform each part in the order from the part farthest from the camera to the part closest to the camera, so that self-occlusions can be dealt with.

The final representation of the appearance does not depend on the initial pose. Results in Fig. 2 show the designed GeTs capture the structure and deformation of the parts very well. After applying GeT, appearance inside two contours can be compared directly in the transform domain. Here for shape matching, we use the inner distance based shape context method [10], which is insensitive to articulations. $h_i(\cdot)$ s are chosen to generate thin plate spline interpolations.

From this case, we show how GeT becomes an interface between shape matching and appearance modeling. Again the geometric context is embedded in the selection of geometric sets. We show GeT can deal with large deformations and articulations without feature points, while AAM is known to be limited to deal with small deformations of feature points that obey a Gaussian distribution [3]. The idea of modeling appearances based on shape matching has been illustrated in [1], but here we formulate it into a GeT that can handle articulations and self-occlusions, and apply it to model the appearances of real world objects. This

method is applied to appearance based pedestrian recognition in Section 7.1.

4. Feature curve/skeleton based GeT

Point sets haven been used in GeT based on shape matching and in AAM. But sometimes we only have correspondences of some feature curves or skeletons, instead of feature points or matched contours. In this case, we propose a feature curve based point set generation. The proposed GeT can be used as an interface between skeleton based shape matching [15] and appearance modeling. The feature curve can be generated from morphological operations, the medial axis [14], principle curves or any skeleton model available.

For example, in Fig. 3, in order to deal with bending, we can utilize the correspondence between two skeletons of the shape. The local coordinate system along the skeleton can be specified as in differential geometry. For example in Fig. 3, at every point on the skeleton, the y-axis is the tangent vector of the curve, while the x-axis is the normal vector. This coordinate system can be used to generate dense point-to-point correspondences by retaining the local coordinates of each point.

Feature curve based GeT can be defined as follows. Suppose we have two matched curves as in Fig. 3, $C_0 = \{(x(s), y(s)) | s \in [0, 1]\}$, $C_1 = \{(x(\tilde{s}), y(\tilde{s})) | s \in [0, 1]\}$ and for $s \in [0, 1]$,

$$S((\tilde{x}(s), \tilde{y}(s))) = \{(x(s), y(s))\}. \quad (9)$$

Then one simple way of generating $S((\tilde{x}, \tilde{y}))$ for any (\tilde{x}, \tilde{y}) inside the contour is as follows:

- ◊ Define the local coordinate system for every point on C and \tilde{C} that reflects correspondences locally. For example, the tangent and normal vectors of the curve at that point can be chosen as bases. But for points at the end, or joints or discontinuities, the system needs to be chosen carefully.

- ◊ For each $P_1 = (\tilde{x}, \tilde{y})$ inside the contour, find $Q_1 = \operatorname{argmin}_{Q \in C_1} |Q - P_1|$. Then find the local coordinate of P_1 at point Q_1 , denoted as (x_{loc}, y_{loc}) .

- ◊ Set $S((\tilde{x}, \tilde{y}))$ as $P_0 = (x, y)$ that has local coordinate (x_{loc}, y_{loc}) (rescale if necessary) at point Q_0 , which is the corresponding point of Q_1 on curve C_0 .

In Fig. 3, we illustrate synthetic images of human arms from a skeleton based GeT. Here the skeleton is obtained by thinning the shape. In Fig. 4, we show another example. Here we assume that the corresponding skeletons across frames are already found. Then we use the skeleton based GeT for image synthesis. The appearance of a human with articulated motion in subsequent frames can be generated from the GeT of the appearance in the first frame. The results are shown together with the ground truth. The skeleton is manually segmented into parts and then each point inside the contour is automatically assigned to the body part

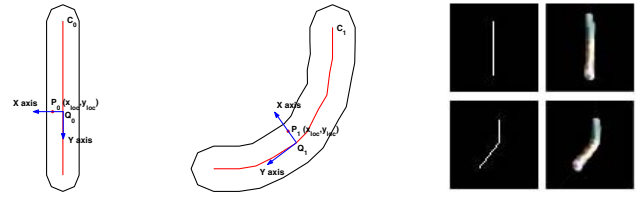


Figure 3: Illustration of mapping through local coordinate systems defined by skeletons. Images 1,2 show how to map P_1 to P_0 according to the local coordinate system at Q_1 , which is the closest point to P_1 on curve C_1 . C_0 and C_1 are matched curves and Q_0 corresponds to Q_1 . Image 3: The skeletons of arm images in Fig. 1 and synthetic appearance by using the skeleton based GeT from one arm to another.



Figure 4: Images 1 and 2: The original image and its skeleton. Images 3,4 and 5: Synthesis results, the ground truth and the skeleton. Next 3 images: Another set of synthetic imagery. The local coordinate systems at joints and end points are chosen as shown in Figure 3.

that its closest skeleton belongs to. GeT is applied in certain order to handle occlusion. We observe that the curve based GeT can handle articulated motion together with large non-linear deformations and occlusions. It is complimentary to contour-driven GeT when no canonical shapes are available or feature curves can be more reliably tracked.

5. Selection of geometric functionals

In Eq. (6), the geometric functional can be changed along the geometric set as in trace transform [12] to obtain different statistics. Some examples of the functionals are listed in section 2.2. We now show a useful geometric functional that helps to deal with occlusions.

The following geometric functional can be used to find the average intensity over set S ,

$$G_s(f(\mathbf{x})) = \frac{\int_s f(\mathbf{x}) d\mathbf{x}}{\int_s H(f(\mathbf{x})) d\mathbf{x}} \quad (10)$$

where $H(\cdot)$ is the Heaviside function and we set $H(0) = 0$. Here f corresponds to the intensity in a contour, and $f(x) > 0$ for $x \in \Omega$, and $f(x) = 0$ outside Ω . So essentially $H(f(x, y))$ gives the mask of the contour region.

Now we show why the functional in Eq. (10) makes GeT insensitive to occlusions that do not change the convex hull of the shape. In Eq. (10), if f is constant inside Ω and the set S is made of a straight line, then the transform will be constant when the line passes through the contour region and zero otherwise. Hence, the shape information is partly



Figure 5: Two illustrations of partially occluded human torso as examples of when the contour changes but the convex hull remains similar. Images 1 to 4 contain partially occluded torso, the ground truth appearance inside the convex hull containing the torso, the reconstructed appearance from the RT in image 1, and the reconstructed appearance by using filtered back projection from the average intensity times the RT of convex hull. Images 5 to 8, show another set of illustration as in 1 to 4. Note in image 6, the ground truth image has outliers because of the arm occludes the torso.

lost. We can only reconstruct the visual hull of the contour through the support of the GeT. However this becomes helpful when part of the contour is missing but the visual hull does not change much. See Fig. 5 for cases like this, when a human walks sideways with respect to the camera and the torso is partly occluded. We also show in Fig. 5 two different ways of reconstruction. One reconstruction is from the RT of $f(\mathbf{x})$ inside Ω . The exact shape and appearance can be recovered. But the appearance in the missing part is not inferred. The other one first applies GeT in Eq. (10) to finding the average intensity along each line. The binary mask of convex hull is obtained from the support of GeT. Then the RT of $f(\mathbf{x})$ inside the convex hull is estimated as the average intensity along each direction times the corresponding RT of the binary mask. Finally the reconstruction is to apply the filtered back-projection algorithm to the estimated RT. We observe that the second one has a fairly accurate reconstruction of the appearance inside the convex hull. Thus using such a GeT helps to represent the appearance with partial occlusions. It essentially gives a very good estimate of the average intensity along each line even for regions of non-uniform intensities.

6. Multiresolution analysis

The resolution problem becomes a primary concern when using model based methods, because it is not reliable to impose point-to-point correspondences. Here we propose a multiresolution geometric transform (MRGeT) that can deal with noisy observations and inexact contour extraction at a proper scale space, as well as properly combining the appearance and shape information. Specifically, we can change the $\chi(\cdot)$ function in Eq. (6) to the following kernel function: $\delta_\epsilon(x) = \frac{1}{\sqrt{2\pi\epsilon^2}} \exp(-\frac{x^2}{2\epsilon^2})$, where ϵ determines the resolution of the kernel. Note $\lim_{\epsilon \rightarrow 0} \delta_\epsilon(x) = \delta(x)$. If x is the distance of the point from the geometric set S , then by replacing $\chi(\cdot)$ with δ_ϵ in Eq. (6), we have a weighted integral in the neighborhood of set S . In the case of basic RT, δ_ϵ corresponds to a line spread function and ϵ determines the width of the spread.

The multiresolution representation can be used to combine the shape and appearance information. Consider introducing a kernel to the functional defined in Eq. (10). The key is to use different resolution parameter in the numerator and the denominator. For the basic geometric set of straight lines, we have an MRGeT used in section 7.1,

$$\mathbf{R}(\theta, p) = \frac{\int f(x, y) \delta_{\epsilon_1}(x \cos(\theta) + y \sin(\theta) - p) dx dy}{\int H(f(x, y)) \delta_{\epsilon_2}(x \cos(\theta) + y \sin(\theta) - p) dx dy}.$$

By changing ϵ_1 and ϵ_2 , we achieve different scale representations for various purposes. (i) When $\epsilon_1 \rightarrow 0$, $\epsilon_2 \rightarrow 0$, $\mathbf{R}(\theta, p)$ corresponds to the average intensity over a straight line, as discussed in section 5. (ii) When $\epsilon_1 \rightarrow 0$, $\epsilon_2 \rightarrow +\infty$, the denominator will be almost constant. So $\mathbf{R}(\theta, p)$ will be a rescaled RT of $f(x, y)$. It can be fully reconstructed using filtered backprojection. (iii) Other combinations of ϵ_1 and ϵ_2 give intermediate representations of the shape and appearance at different scales. ϵ_1 adjusts the resolution of the appearance. ϵ_2 depends on if we need to model the appearance in the actually shape or its convex hull. Using a bigger ϵ_2 allows a more accurate description of the shape, since $\mathbf{R}(\theta, p)$ is closer to a scaled RT. Using a smaller ϵ_2 makes $\mathbf{R}(\theta, p)$ closer to the average intensity along the line, thus modeling the appearance inside the convex hull. It helps to handle occlusions that do not change the convex hull of the shape. In the case when the shape is very close to its convex hull, ϵ_2 can be bigger since there is not much occlusion.

Another nice property of $\mathbf{R}(\theta, p)$ is that it still carries properties of basic RT with respect to the similarity transform. Suppose $\tilde{f}(x, y) = f(T(x, y))$, where $T(x, y) = s \begin{bmatrix} \cos(\alpha) & \sin(\alpha) \\ -\sin(\alpha) & \cos(\alpha) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$, then the transform of \tilde{f} can be easily shown as

$$\tilde{\mathbf{R}}(\theta, p) = \mathbf{R}(\theta - \alpha, t_x \cos(\theta - \alpha) + t_y \sin(\theta - \alpha) + sp). \quad (11)$$

Following Eq. (11), the registration with respect to the similarity transform can be easily obtained. If we register two contours by first aligning their centroids and then scale them according to the ratio of area, the only unknown in Eq. (11) is α , which simply corresponds to a translation in the transform domain. Thus it is very easy to match the appearances inside two contours when they are related by a 2D similarity transform.

7. Applications

In Figs. 2(d), 3 and 4, we showed how to use GeT for image synthesis. Now we show how to apply GeT to pedestrian recognition with and without part segmentation.

7.1. Object recognition under articulation

In this section, we design GeT's to incorporate geometric context into appearance modeling for objects with articu-

lated motion, bending, and local deformations. The appearance of human and body parts provides very good examples for our study. We illustrate all the methods introduced in this paper. Modeling the appearance of human is useful because sometimes it is more reliable than gait, for example, in the application of persistent tracking. Here we focus on linking the identity of humans to the representation of appearances in the transform domain.

We test three approaches with the same setting over the USF database [13] as in Fig. 6, where the body parts have been manually marked and the size of each image is around 125×72 . The first two approaches use the part information while the third one does not. *Approach I*: we design GeT for each body part and study the matching of parts as well as combined recognition of humans. We note that, although the images in the database have been manually segmented into body parts, the part-based recognition task is still not easy because of low-resolution, poor quality imagery and errors in segmentation. We apply different transforms for each body part according to their motions and possible occlusions. Because right arms and right legs are often occluded in this dataset, we discard these parts for recognition purpose. *Approach II*: as a comparison, we directly match each part using rigid templates, then combine them for human identification. *Approach III*: without using the information of manually segmented parts, we apply GeT based on shape matching to deal with articulation, as discussed in Section 3.2.

The experimental setting is as follows. There are 71 classes in this data-set. For each class, we have one image in the gallery and 28 in the probe set that are taken under similar conditions. We classify the probe image according to its distance to the gallery image either in the transform domain as in Approach I and III, or the pixel domain as in Approach II. For Approach I, each part is represented using the designed GeTs before distance is calculated. The properties of MRGeT in Eq. (11) is used to align the two parts. The choice of GeTs used in Approach I is as follows:

Head: Use MRGeT with $\epsilon_1 = 2$ and $\epsilon_2 = 4$. We set $\epsilon_2 = 4$ because the actual shape is close to its convex hull.

Torso: Since occlusion needs to be considered while the convex hull of the shape does not change much, we use MRGeT with $\epsilon_1 = 2$ and $\epsilon_2 = 2$ so that the transform is close to the average intensity along the line.



Figure 6: Sample of USF database from 3 classes. Walking pedestrians with manually segmented body parts. The first image for each class is in the gallery. Second image is in the probe set.



Figure 7: Sample results for matching body parts using GeT for Approach I. Probe images from 3 classes are illustrated, corresponding to subjects in Fig. 6. Here each class has 5 images for one part. The first image is the probe image. The second image is the correct match in the gallery using GeTs for parts. The next three images show the top 3 matches in the gallery. The ranks of the correct match for each class and each part are: from up to down, 2,1,58 for head. 3,1,1 for torso, 18,11,1 for left arm, 5,16,3 for left upper leg, 2,5,11 for left lower leg. The ranks of correct match of human by combining parts are 1,1,4 for Approach I, and 6,13,10 for Approach II.

Left Arm: We use GeT based on the level set. Using the signed distance as the level set function may not generate the structure as shown in Fig. 1, so instead the shape is first thinned to a skeleton, and we select the geometric set that contains points on the equal distance line to the skeleton. We further divide the geometric set into two parts according to whether the point is closer to the upper half of the skeleton or the lower half, so that the structure of the arm is considered. GeT takes the average intensity along those sets.

Left Upper Leg: Mainly due to 2D rigid motion, we choose $\epsilon_1 = 2$ and $\epsilon_2 = 3$. A small ϵ_2 can allow a certain degree of occlusion.

Left Lower Leg: Mainly due to 2D rigid motion, we choose $\epsilon_1 = 2$ and $\epsilon_2 = 4$.

For Approach II, each part is matched using sum of squared distance by only allowing rigid transformations. Note for torso, we use the appearance inside the convex hull to reduce the effect of occlusion. For the above two methods, the distance is normalized to a standard log-normal distribution as illustrated in Fig. 8(c). For combined recognition, we classify the probe image to the class that shows the least weighted distance. The results for matching each body part as well as their heuristically chosen weights are shown in Fig. 7 and Table 1. The part numbering is in the order as in Fig. 7. Fig. 8(a)(b)(d) shows the cumulative match curves (CMC's) for matching each body part and the combined recognition results.

For Approach III, the image in the gallery set is transformed to the normalized appearance at all six poses using the GeT described in Section 3.2. The image in the probe

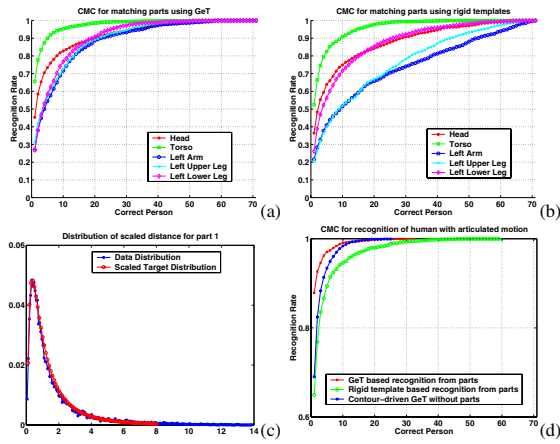


Figure 8: (a)(b) Cumulative matching curves of matching parts for Approach I and II. (c) Illustration of normalizing the distance for part 1(head). We observe that the distance has similar distribution as a log-normal distribution with shape and scale parameters σ, m . The data is normalized to have $m = \sigma = 1$ and shown along with the scaled target distribution. (d) Combined recognition rate of human appearance for all three cases.

Table 1: Top One Recognition Rate (%)

Part No.	1	2	3	4	5	All
GeT(with parts)	45.3	65.6	27.0	31.1	26.8	87.9
Templates	36.4	52.4	21.6	20.4	26.2	64.9
GeT(no parts)	-	-	-	-	-	69.0
weights	0.2	0.4	0.13	0.13	0.13	

set is transformed to the normalized appearance at its closest pose and the corresponding *mirror* pose. By a *mirror* pose, we mean two similar silhouettes with different topology of parts, such as pose 1 and 4 in Fig. 2(a). This helps to obtain more robust matching results. Then we match the transformed image with the normalized gallery image at the same pose and choose the closest match. This way we accomplish human identification without part segmentation.

As we can see, Approach I outperforms Approach II for each part and both I and III does better than II in combined recognition of human. Comparing Approach I and II in part recognition, we observe that GeT obtains 10% higher rates for parts 2 and 4. It is mainly due to the advantage of GeT for handling occlusion. Overall, the matching of parts is a difficult task, as we can see in Fig. 7. Part 3 usually contains very few pixels and is very blurred, thus contour-driven GeT only gets 6% higher than rigid templates. For part 5, GeT is only slightly higher, because part 5 displays mostly 2D rigid motion with no occlusions. For overall recognition, despite the non-rigid motions that the probe images have with respect to the gallery images, GeT’s top one recognition with part information is as high as 87.9%, while contour-driven GeT without part segmentation gives 69.0%. The superior performance Approach III over II shows that the designed

GeT handles the articulation better even though Approach II uses the part information.

8. Conclusion

In summary, a very general definition of geometric transform is given that unifies Radon transform, trace transform and image warping. We show how to design the two key elements of GeT, namely, geometric sets and functionals, to incorporate geometric context into appearance modeling. We also propose a multi-resolution representation by using kernel functions. Essentially, all the components of the basic RT in Eq. (1) are generalized. Future work includes modeling appearance over time and applying this approach to segmentation and tracking.

Acknowledgments

Partially funded by the ARDA/VACE program (contract 2004H80200000). We thank Haibin Ling for his code of shape matching.

References

- [1] S. Belongie *et al.* Shape matching and object recognition using shape contexts. *PAMI*, 24(4):509–522, 2002.
- [2] V. Blanz and T. Vetter. Face recognition based on fitting a 3d morphable model. *PAMI*, 25(9), 2003.
- [3] T. Cootes and C. Taylor. Statistical models of appearance for medical image analysis and computer vision. In *Proc. SPIE Medical Imaging*, 2001.
- [4] L. Ehrenpreis. *The Universality of the Radon Transform*. Clarendon Press, Oxford, 2003.
- [5] D. Forsyth and J. Ponce. *Computer Vision, A Modern Approach*. Prentice Hall, 2003.
- [6] L. Gorelick *et al.* Shape representation and classification using the poisson equation. In *CVPR*, June 2004.
- [7] A. Jain. *Fundamentals of digital image processing*. Prentice-Hall, Inc., 1989.
- [8] A. C. Kak and M. Slaney. *Principles of Computerized Tomographic Imaging*. Soc. of Industrial and Appl. Math., 2001.
- [9] M. Lades *et al.* Distortion invariant object recognition in the dynamic link architecture. *IEEE Trans. on Computers*, 42:300–311, 1993.
- [10] H. Ling and D. Jacobs. Using the inner-distance for classification of articulated shapes. In *CVPR*, June 2005.
- [11] S. Osher and N. Paragios. *Geometric Level Set Methods*. Springer-Verlag, 2003.
- [12] M. Petrou and A. Kadyrov. Affine invariant features from the trace transform. *PAMI*, 26(1):30–44, 2004.
- [13] P. Phillips *et al.* The gait identification challenge problem: data sets and baseline algorithm. In *ICPR*, 2002.
- [14] T. Sebastian, P. Klein, and B. Kimia. Recognition of shapes by editing their shock graphs. *PAMI*, 26(5):550–571, 2004.
- [15] A. Tamrakar and B. Kimia. Medial visual fragments as an intermediate image representation for segmentation and perceptual grouping. In *POCV*, 2004.
- [16] R. Veltkamp and M. Hagedoorn. State-of-the-art in shape matching. Technical Report UU-CS-1999-27, 1999.
- [17] G. Wolberg. *Digital Image Warping*. IEEE Computer Society Press, 1994.