Available online at www.sciencedirect.com
SCIENCE DIRECT®
Image and Vision Computing 22 (2004) 1229–1239
image and vision computing
www.elsevier.com/locate/imavis

Discrete contours in multiple views: approximation and recognition

M. Pawan Kumar, Saurabh Goyal, Sujit Kuthirummal, C.V. Jawahar*, P.J. Narayanan

Centre for Visual Information Technology, International Institute of Information Technology, Gachibowli, Hyderabad 500 019, India

Received 20 February 2003; received in revised form 13 January 2004; accepted 17 March 2004

Abstract

Recognition of discrete planar contours under similarity transformations has received a lot of attention but little work has been reported on recognizing them under more general transformations. Planar object boundaries undergo projective or affine transformations across multiple views. We present two methods to recognize discrete curves in this paper. The first method computes a piecewise parametric approximation of the discrete curve that is projectively invariant. A polygon approximation scheme and a piecewise conic approximation scheme are presented here. The second method computes an invariant sequence directly from the sequence of discrete points on the curve in a Fourier transform space. The sequence is shown to be identical up to a scale factor in all affine related views of the curve. We present the theory and demonstrate its applications to several problems including numeral recognition, aircraft recognition, and homography computation.

Keywords: Planar shape recognition; Polygonal approximation; Fourier transform; Invariant; Projective geometry; Piecewise conic approximation

1. Introduction

Analysis of multiple views of the same scene is an area of active research in computer vision today. The study of the structure of projections of points and lines in two views received much attention in the 1980s and early 1990s [1–3]. Similar study on the projections of points and lines in three and more views followed since then [3–5]. These multi-view studies have concentrated on how geometric primitives like points, lines and planes are related across views. Specifically, the algebraic constraints satisfied by the projections of such primitives in different views have been the focus of intense studies. Discrete contours formed by boundaries of objects are of great interest for shape recognition. The contour consists of an ordered sequence of points. While projection of each point satisfies multi-view constraints separately, their sequence can have additional constraints that can help in recognizing them under multiple views.

We study the issues in recognizing discrete planar contours, consisting of a sequence of points, in multiple views under different projection models in this paper. We limit the recognition problem to planar objects represented using their discrete boundaries. Planar shape recognition has been studied well. Recognition by alignment [6], polygonal approximation [7], based on geometrically invariant features [8], linear combination of models [9], etc. have appeared in the literature. Descriptors computed in the Fourier domain [10] have also been used for recognition. These algorithms work well for similarity transformations between views. The image-to-image transformation is more general in practical situations of interest. When a planar object is imaged from multiple viewpoints using a perspective camera, its images are generally related by a projective homography. In many practical situations, the homography is close to be an affine. For both affine and projective cases, conventional algorithms, designed for similarity transforms, will not be enough.

The problem of shape recognition in the context of multiple views can be posed thus. Given the image of an object in one or more views, can we recognize them to be images of the same object when the viewing parameters of the cameras are unknown.

Projective and affine invariants for points, lines and parametric and algebraic curves, such as conics, have been discovered [11]. Discrete contours, not amenable to simple algebraic or parametric representations, pose new problems and have not been studied much earlier. Differential invariants that require higher order derivatives of contours have been used [11,12]. Subsequently integrals involving

* Corresponding author.
E-mail address: jawahar@iiit.net (C.V. Jawahar).

0262-8856/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.imavis.2004.03.022

differentials of image curves have been used successfully in recognition systems [13]. Another relevant effort identifies a set of affine invariant features and uses them for recognition [14] in a classical pattern recognition framework.

In this paper, we present two approaches for the recognition of discrete planar contours in multiple views. The first approach is inspired by polygonal approximation techniques [7]. We propose a polygonal approximation algorithm for an invariant characterisation of a contour under projective transformation. We extend it to piecewise parametric approximation of the discrete contour. The specific parametric structures used are lines, resulting in a polygonal approximation, and conics. We show how the multi-view invariants of the underlying parametric structure can be exploited to devise a projective invariant piecewise parametric approximation. Thus, the approximation computed from multiple views of the contour will be isomorphic to each other. This necessary condition satisfied by matching contours can be used for recognizing its shape in multiple views without explicit correspondence between points.

The second approach uses new contour invariants computed from the sequence of points on the contour. This approach combines multi-view constraints on corresponding points and the Fourier domain constraints of the sequence of points. We achieve their fusion in a Fourier domain representation of the point sequence. We show that the novel Fourier vector representation of the contour points yields a single-view invariant sequence κ that is identical up to a scale factor for all affine-related projections of the contour. This can be the basis for recognizing the shape in multiple views. This is a direct method for recognition that does not involve the selection of starting points or approximation of the contour using parametric structures. We establish the contour invariants for affine homographies, but show that they are satisfied for most practical situations when the image-to-image homography is not strictly affine. We present a general contour invariant in this paper. In a previous work, we have shown that recognition constraints based on the magnitude and phases of the Fourier vector representation also exist [15].

Section 2 presents the basic problem formulation and establishes the notation used throughout this paper. Section 3 discusses the general theory behind projective invariant piecewise parametric representation of discrete contours. The Fourier domain contour invariant is presented in Section 4. We establish the affine invariant and present the multi-view recognition condition as a rank constraint on a measurement matrix. Section 5 shows that the problem of recognizing polygonal shapes in multiple views can be solved using the Fourier domain invariant on a sequence of lines in the dual space. Section 6 shows the results of applying both approaches for contour recognition on a number of applications involving the recognition of numerals and aircraft silhouettes. We also show the extension of the recognition scheme to compute point-wise correspondence between matching contours and then to compute the homography between a pair of views. Section 7 presents a few concluding remarks.

2. Problem formulation

Let **O** be a set of N points on the boundary of a planar object imaged in multiple views. Let $(u^l[i], v^l[i], w^l[i])$ be the homogeneous coordinates of points on the image of the closed boundary in view l. This boundary is represented by a sequence of vectors as shown below

$$\mathbf{x}^l[i] = \begin{bmatrix} u^l[i] \\ v^l[i] \\ w^l[i] \end{bmatrix}$$

When a planar object is imaged from multiple viewpoints, points on it undergo a transformation that can be mathematically described as a general projective transformation, which is a linear operation (3×3 matrix) in homogeneous coordinates

$$\mathbf{x}^l[i] = \mathbf{M}^l \mathbf{x}^0[i] \tag{1}$$

In some special cases, additional constraints on $\mathbf{M}^l$ can result in affine or similarity transformation.

Two views are related by a similarity transformation only when there is a rotation, and/or a translation, and/or a uniform scaling between them. Most reported recognition algorithms are primarily designed to handle similarity transformations. However, affine transformations deform the shape of an object beyond rigid motions though they do preserve parallelism and points at infinity. An affine homography between two images is represented by a $\mathbf{M}^l$ matrix with its last row as [0 0 1]. Affine transformations form a subset of the general projective group.

Projective transformations, in general, do not preserve parallelism of lines and can map points at infinity to finite points and vice versa. It has been shown that a planar object viewed from multiple viewing positions results in a projective image-to-image homography [3]. However, for cases like narrow field of view or imaging an object from a distance, an affine transformation is considered to be a good approximation of the projective transformation.

As we go higher up in the hierarchy of transformations (similarity $\subset$ affine $\subset$ projective) the constructs that are preserved (remain invariant) become fewer, making the task of recognition more challenging. We address the problem of planar object recognition using appropriate invariants.

2.1. Invariants

Let p be a parameter vector subject to the linear transformation $\mathbf{M}^l$. A scalar invariant $I(p)$ of p is preserved under the transformation if $I(p^l) = I(p^0)$ [11]. Here, $I(p^l)$ is the function of the parameters after the linear transformation. If the transformation $\mathbf{M}^l$ is affine, the invariants are called affine invariants. For example, ratio of areas and ratio of lengths on parallel lines are invariant to affine transformations. Several cross-ratios are invariant under projective transformation. Cross-ratio of four collinear points is defined as the ratio of ratios of distances between points. Below we describe the cross-ratios that we employ, namely cross-ratio of areas and concurrent lines.

Cross-ratio of areas of five points. The cross-ratio of the areas of five points $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4$, and $\mathbf{x}_5$ is defined by

$$\mathrm{cr}(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5) = \frac{\Delta_{\mathbf{x}_1\mathbf{x}_2\mathbf{x}_5}\Delta_{\mathbf{x}_3\mathbf{x}_4\mathbf{x}_5}}{\Delta_{\mathbf{x}_1\mathbf{x}_3\mathbf{x}_5}\Delta_{\mathbf{x}_2\mathbf{x}_4\mathbf{x}_5}}, \quad (2)$$

where $\Delta_{\mathbf{x}_i\mathbf{x}_j\mathbf{x}_k}$ is the area of the triangle formed by points $\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k$. This is invariant to general linear or projective transformations [11].

Cross-ratio of four concurrent lines. The cross-ratio of four concurrent lines l_1, l_2, l_3, l_4 is defined as

$$\mathrm{cr}(l_1, l_2, l_3, l_4) = \frac{\sin\,\theta_{13}\,\sin\,\theta_{24}}{\sin\,\theta_{23}\,\sin\,\theta_{14}}, \quad (3)$$

where θ_{ij} represents the angle formed by the lines l_i and l_j. This is invariant to a general projective transformation [11].

We employ these invariances for piecewise parametric approximation of discrete contours.

3. Projectively invariant piecewise parametric representation

Parametric representation is a popular method for providing a compact representation of planar contours. A piecewise parametric representation of a contour partitions the points on the contour into sets, and represents each set with the help of a few parameters. One example of such a representation is polygonal approximation. This has been extensively used as an intermediate step in various applications such as planar shape recognition, volume rendering and multi-resolution modeling [16–18].

Polygonal approximation may be achieved by minimizing the error in approximation. A general optimization of the objective function may be computationally expensive and prone to get stuck in local minima. Most popular polygonal approximation algorithms look for an optimal solution using a greedy algorithm [7,19]. Some of these algorithms are designed for fast approximation. They, in general, exploit the connectedness and ordering of the points in the set. There are also algorithms that do not exploit the connectedness of points [20]. They group points on the boundary into linear clusters. Some other algorithms emphasize the optimality and efficiency of the polygonal approximation [21]. Conic sections are also often used as the parametric representation for each partition [22].

Let $\mathbf{x}^l[i]$, $i = 1, 2, \ldots, n$ be a set of points on the boundary of a planar object to be approximated using parametric sections in the lth view. These sequences in multiple views are related by $\mathbf{x}^l[i] = \mathbf{M}^l\mathbf{x}^0[i]$. In individual views, these boundary sequences can be approximated into k mutually exclusive and collectively exhaustive subsets $\phi_1^l, \ldots, \phi_k^l$ such that each of the subsets can be approximated using a few parameters. This can be achieved by minimizing an objective function of the form

$$J^l = \sum_{i=1}^{|\phi_j^l|} d(\mathbf{x}^l[i], Q_j^l), \qquad \mathbf{x}^l[i] \in \phi_j^l \quad (4)$$

where Q_j^l is the parametric structure which approximates the points in ϕ_j^l and $d(\cdot)$ is a measure of separation of $\mathbf{x}^l[i]$ from Q_j^l. The separation $d(\cdot)$ can be the distance along the normal or a measure of how the parametric structure Q_j^l distorts the real point $\mathbf{x}^l[i]$. For the parametric representation to be invariant to projective transformations, there should be a one-to-one mapping from ϕ_i^l to ϕ_i^0.

We now show that as long as we can identify a projective invariant which gives a measure of separation for the desired parametric structure, we can find an algorithm to represent the curve using that parametric structure. We discuss two specific cases of this approach in the form of polygonal approximation and piecewise conic section approximation.

3.1. Proposed approach

The existence of a projectively invariant piecewise parametric method depends on the availability of a projective invariant which can give a measure of separation of a point from the desired parametric structure. Let $I(p)$ be a projective invariant for a set of points p and let $I(\mathbf{x}[i])$ denote its value when all points of p except $\mathbf{x}[i]$ are kept fixed. If the locus of $\mathbf{x}[i]$ for which $I(\mathbf{x}[i]) = k$ is a parametric structure Q, we can use $|I(\mathbf{x}[i]) - k|$ as a measure of separation. We first select the fixed points and compute the invariants corresponding to those points. We then move along the contour until the approximation error is sufficiently large. Thus, we can identify regions in the curve which can be approximated using a parametric structure.

3.2. Projective invariant polygon approximation

In this section, we describe an algorithm that can result in isomorphic polygonal approximation for projectively transformed versions of the same image. Some preliminary results of this have been presented in Ref. [16].

Consider points $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5$ and $\mathbf{x}_6$ (see Fig. 1). Using Eq. (2), we get

$$\mathrm{cr}_{12345} = \frac{\Delta_{541}\Delta_{321}}{\Delta_{531}\Delta_{421}}, \qquad \mathrm{cr}_{12346} = \frac{\Delta_{641}\Delta_{321}}{\Delta_{631}\Delta_{421}}.$$

We see that the area $\Delta_{541} = \frac{1}{2}\mathrm{len}_{14}\mathrm{dist}_5^{14}$, where len_{14} is the length of the line segment defined by points 1 and 4

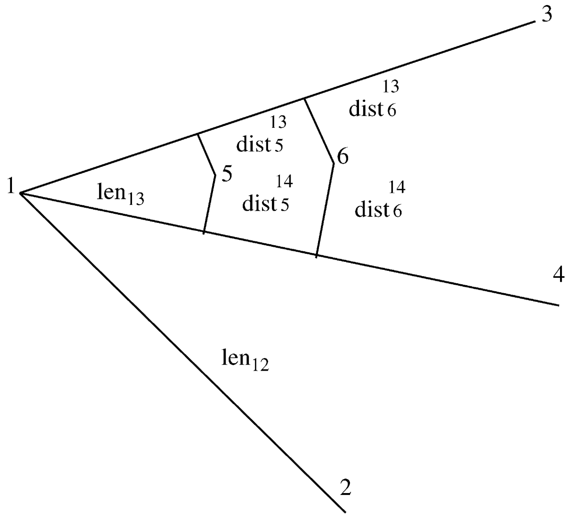

Fig. 1. Points 1, 2, 3, 4, 5 and 6 along with the distances.

and dist_5^{14} is the perpendicular distance of point 5 from the line defined by points 1 and 4. Now we consider the ratio of cross-ratios [11,16] as

$$\lambda = \frac{\mathrm{cr}_{12345}}{\mathrm{cr}_{12346}} = \frac{\Delta_{541}\Delta_{631}}{\Delta_{531}\Delta_{641}} = \frac{\mathrm{len}_{14}\mathrm{dist}_5^{14}\mathrm{len}_{13}\mathrm{dist}_6^{13}}{\mathrm{len}_{13}\mathrm{dist}_5^{13}\mathrm{len}_{14}\mathrm{dist}_6^{14}} = \frac{\mathrm{dist}_5^{14}/\mathrm{dist}_5^{13}}{\mathrm{dist}_6^{14}/\mathrm{dist}_6^{13}}. \quad (5)$$

We first establish three useful properties before proving the existence of projective invariant polygon approximation.

Property 1. *An invariant polygonal approximation algorithm can exist only for a transformation where collinearity is preserved. To verify this, we assume that we have a polygonal approximation algorithm invariant to a transformation which does not preserve collinearity. Under such a transformation, not every line gets transformed as a line. From this it is clear that there cannot exist a polygonal approximation algorithm which is invariant to such a transformation. Since projective transformation preserves collinearity, there can exist a polygonal approximation algorithm invariant to this transformation.*

Property 2. *The ratio of cross-ratios of areas gives us a measure of deviation $d(\cdot)$ for polygonal approximation. This is true as Eq.* (5) *shows that λ can be interpreted as the ratio of the ratio of perpendicular distances of the points $\mathbf{x}_5$ and $\mathbf{x}_6$ from the two lines $line_{13}$ and $line_{14}$. This value equals 1 for the line $line_{15}$. When the point $\mathbf{x}_6$ moves away from this line, λ moves away from 1. Thus, λ gives a measure of the collinearity of the points $\mathbf{x}_1, \mathbf{x}_5$ and $\mathbf{x}_6$. The measure $d(\cdot) = |\lambda - 1|$ can then be used as the measure of deviation as given in Eq. (4).*

Property 3. *The ratio of cross-ratios of area (λ) is invariant to projective transformations since the cross-ratios of areas are projectively invariant.*

Therefore, there exists an algorithm for polygonal approximation which is invariant to projective transformation as it is clear that there is a function $d(\cdot)$ which is invariant to projective transformation. This would result in a one-to-one mapping between the subsets $(\phi_1, \phi_2, \ldots, \phi_k)$.

We develop the algorithm in the following manner.

(1) Choose point 1 as the starting point and point 5 as the point next to point 1 on the curve. Points 2, 3, and 4 may or may not lie on the curve.
(2) Choose the point adjacent to 5 on the curve as point 6. Measure $d(6) = |\lambda(6) - 1|$.
(3) If $d(6) < t$ (a suitable tolerance threshold), move point 6 forward. Go to step 2.
(4) Otherwise, approximate the curve by the line segment joining points 1 and 6. Set point 6 as the new point 1. Go to step 1.

The algorithm described above was implemented to work with real images. Points 3, 4, and 5 have to be chosen carefully in order to overcome errors that are present in real images. The most prominent error is due to the discretization of the curve because of which the cross-ratios calculated may differ in different views [11]. In order to decrease the effect of discretization, points 3, 4 and 5 were chosen far apart from each other so that the areas to be calculated for λ were large. This prevents a small change in the locations of points 3, 4 or 5 to affect the cross-ratios noticeably. Point 5 need not be the same in multiple views of the same object as long as it is close to point 1. This is because the contours are smooth, so we can assume that points within a small distance from point 1 are almost collinear and hence give the same value for λ.

The reference points should not be collinear in which case the cross-ratio of areas will be undefined. Another problem that arises is obtaining points 3 and 4 which correspond in different views. An affine or euclidean invariant heuristic to obtain these points can be used which leads to slight differences in the positions of points 3 and 4. In most real images, if the threshold is kept low, differences in the positions of points 3 and 4, change the locations of boundary points by only 1–2 pixels. The choice of the threshold should depend on the curvature of the section that we are trying to approximate. For example, for a linear section the threshold should be low to retrieve the same linear section as the approximation.

Example. The results of the proposed algorithm on two planar boundaries are shown in Fig. 2. The results were obtained on images of size 300×300. We considered only the boundary pixels for approximation. The original image

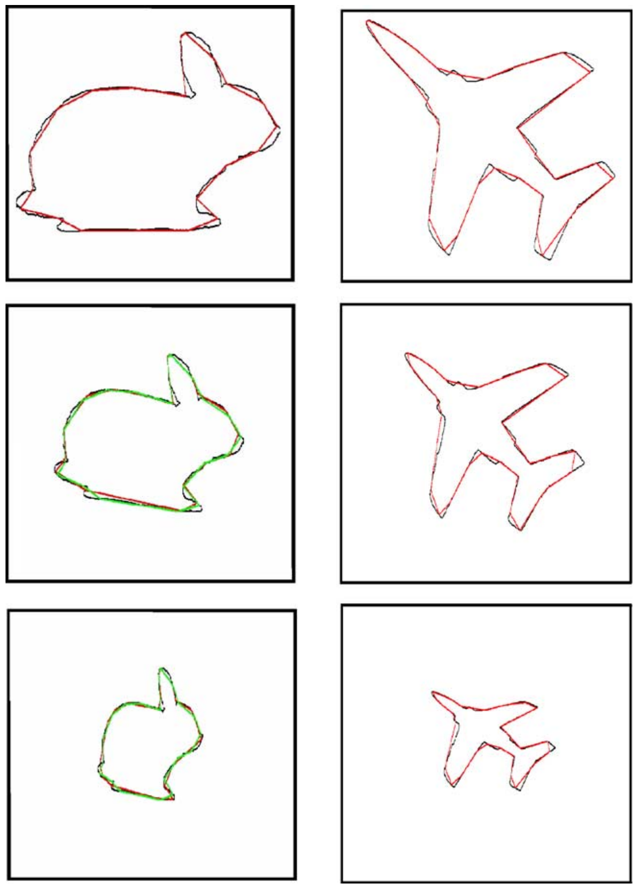

Fig. 2. Polygon approximation of two original curves and after applying two projective transformations. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article).

is shown on top with two projectively transformed versions below it. The boundary pixels are drawn in black colour. The red lines in the figure show the polygonal approximation of the boundary using our algorithm. The green lines show the projectively transformed polygon which approximates the original images. The green and red polygons in the figures are identical in all cases except one or two breakpoints which got shifted by 1–2 pixels due to discretization. We obtained good results on other planar boundaries as well.

3.3. Projectively invariant piecewise conic approximation

The general method for parametric approximation described in Section 3.2 can be applied for finding piecewise conic approximations to a closed curve which is invariant to projective transformations. Such a method is possible because a projective invariant exists which can give the measure of deviation from a conic section. This invariant is the cross-ratio of lines which is defined for four concurrent lines as

$$\text{cr Lines}(\mathbf{l}_1, \mathbf{l}_2, \mathbf{l}_3, \mathbf{l}_4) = \frac{\sin\,\theta_{13}\,\sin\,\theta_{24}}{\sin\,\theta_{23}\,\sin\,\theta_{14}} \tag{6}$$

where θ_{ij} is the angle between lines $\mathbf{l}_i$ and $\mathbf{l}_j$. This is invariant to general linear or projective transformations [11]. This can alternatively be defined for five points in general position as

$$\text{cr}(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5) = \text{cr Lines}(\mathbf{l}_{15}, \mathbf{l}_{25}, \mathbf{l}_{35}, \mathbf{l}_{45}) \tag{7}$$

where $\mathbf{l}_{ij}$ is the line joining points $\mathbf{x}_i$ and $\mathbf{x}_j$.

According to Chasles' theorem [23], if we keep $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3$ and $\mathbf{x}_4$ fixed, the locus of all points $\mathbf{x}$ such that $\text{cr}(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x})$ is fixed, is a conic curve. If we define $\lambda = \text{cr}(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5)$, then the value $|\text{cr}(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}) - \lambda|$ gives the measure of separation of point $\mathbf{x}$ from the conic defined by points $\mathbf{x}_1 - \mathbf{x}_5$.

The above results are used to develop an algorithm for projectively invariant piecewise conic approximation. The algorithm is similar to the one described for polygonal approximation except that the measure of deviation used is the cross-ratio of lines. We denote the starting point by $\mathbf{x}_1$. Points $\mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4$ and $\mathbf{x}_5$ are chosen near $\mathbf{x}_1$ on the curve. We then move clockwise from point $\mathbf{x}_5$ and compute the deviation $d(\mathbf{x})$ as the difference between the cross-ratio of lines of points $(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5)$ and $(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x})$ for every boundary point $\mathbf{x}$. As long as $d(\mathbf{x})$ is less than a suitable tolerance threshold, the visited points can be approximated with a conic section. When $d(\mathbf{x})$ becomes larger than the threshold, we put a breakpoint there and continue with the next point on the boundary as $\mathbf{x}_1$.

The algorithm will partition the curve points into sets which can be approximated using a conic section. A projectively invariant conic fitting algorithm should then be used to estimate the conic section which fits those points. This, however, does not pose a big problem as the points to be fitted are not highly scattered and therefore algorithms which are affine or similarity invariant lead to only very small errors in fitting [22].

Example. Fig. 3 shows the breakpoints obtained by the above mentioned algorithm on two views of a contour. The boundary is shown in black and the breakpoints are shown by the red dots. The contours show the outer boundary points of five intersecting circles. In both the views the conic approximation obtained is identical and the breakpoints are very close to the intersections of the circles.

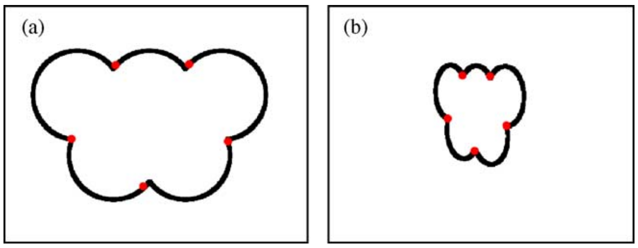

Fig. 3. Breakpoints found by the conic approximation algorithm on two views of a contour. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article).

4. Fourier domain affine invariant for discrete contours

In this section, we analyze the properties of a collection of points, such as a planar object's contour, in a transform domain. Collections of points such as a boundary have more information than isolated points. The sequencing inherent in such a collection makes a transform domain approach, such as the Fourier one, a good tool to study their properties. The linear image-to-image relationships combined with the properties of the contour in the Fourier domain enable rich constraints that essentially characterize the contour independent of the viewpoint. We come up with a view-independent characterization of the planar shape boundary using a measure computed in the Fourier domain. Some preliminary results were presented in Refs. [15,24].

4.1. Fourier domain representation of the contour

Let the Fourier domain representation of the sequence $\mathbf{x}^l[i], 0 \le i < N$ (introduced in Section 2) be $\mathbf{X}^l[k], 0 \le k < N$ such that

$$\mathbf{X}^l[k] = \begin{bmatrix} U^l[k] \\ V^l[k] \\ W^l[k] \end{bmatrix}$$

where $U^l[k]$, $V^l[k]$, and $W^l[k]$ are, respectively, the Fourier transforms of the individual sequences $u^l[i]$, $v^l[i]$, and $w^l[i]$. We call this the vector Fourier representation of the contour. Note that the sequences $\mathbf{X}^l[k]$ are periodic and conjugate symmetric.

Theorem 1. *The Fourier transform and the collineation commute with the above representation. That is, if points are transformed between views 0 and l using Eq. (1), the same homography will transform corresponding frequency terms in the Fourier domain also. In other words*

$$\mathbf{X}^l[k] = \mathbf{M}^l\mathbf{X}^0[k], \qquad 0 \le k < N. \tag{8}$$

Proof. Let $\mathbf{M}^l = m^l_{ij}, 1 \le i,j \le 3$. Expanding Eq. (1) for the u term

$$u^l[i] = m^l_{11}u^0[i] + m^l_{12}v^0[i] + m^l_{13}w^0[i]$$

Taking the Fourier transform of the above equation and using the linearity property of Fourier transforms, we get

$$U^l[k] = m^l_{11}U^0[k] + m^l_{12}V^0[k] + m^l_{13}W^0[k]$$

Similarly for $V^l[k]$ and $W^l[k]$ (note that $w^l[i]$ need not be 1). It is now easy to see that

$$\mathbf{X}^l[k] = \mathbf{M}^l\mathbf{X}^0[k]$$

giving us the desired result. □

Given a set of M views, the recognition problem can be formulated as the identification of a view-independent function $f(\cdot)$ such that $f(\mathbf{x}^0, \mathbf{x}^1, \ldots, \mathbf{x}^M) = 0$. This recognition constraint can be linear or nonlinear in image coordinates. The algebraic relation given by $f(\cdot)$ can then be used to settle the question whether the M observed views were of the same object.

4.2. Rank constraint for recognition

If the image-to-image homography is affine, the transformation matrix has $m^l_{31} = m^l_{32} = 0$ and $m^l_{33} = 1$. The transformation can be expressed in terms of inhomogeneous coordinates as

$$\mathbf{x}^l[i] = \mathbf{A}^l\mathbf{x}^0[i] + \mathbf{b}^l \tag{9}$$

where $\mathbf{x}^l[i]$ is the inhomogeneous representation of the ith point on the contour in view l, $\mathbf{A}^l$ is the upper 2×2 minor of $\mathbf{M}^l$ and $\mathbf{b}^l$ is the upper two elements of the last column of $\mathbf{M}^l$.

The above expression is valid for the scenarios when correspondence between points across views is known. However, in practice, correspondence is rarely available. In case correspondence information is not available, Eq. (9) assumes the form

$$\mathbf{x}^l[i] = \mathbf{A}^l\mathbf{x}^0[i + \lambda_l] + \mathbf{b}^l$$

where cyclic shifting of the sequence $\mathbf{x}^0$ by λ_l would align the corresponding points of $\mathbf{x}^0$ and $\mathbf{x}^l$. The frequency domain representation can be given by

$$\mathbf{X}^l[k] = \mathbf{A}^l\mathbf{X}^0[k]\exp\left(\frac{j2\pi\lambda_l k}{N}\right), \qquad 0 < k < N \tag{10}$$

if the $\mathbf{b}^l$ term is eliminated by omitting the $k = 0$ term in the Fourier domain.

Let us define a measure called the cross-conjugate product on the Fourier representations of two views as

$$\psi(0,l)[k] = (\mathbf{X}^0[k])^{*\mathrm{T}}\mathbf{X}^l[k],$$
$$0 < k < N = (\mathbf{X}^0[k])^{*\mathrm{T}}\mathbf{A}^l\mathbf{X}^0[k]\exp\left(\frac{j2\pi\lambda_l k}{N}\right). \tag{11}$$

The matrix $\mathbf{A}^l$ can be expressed as a sum of a symmetric matrix and a skew symmetric matrix as $\mathbf{A}^l = \mathbf{A}^l_{\mathrm{s}} + \mathbf{A}^l_{\mathrm{sk}}$ where $\mathbf{A}^l_{\mathrm{s}} = \frac{1}{2}(\mathbf{A}^l + (\mathbf{A}^l)^{\mathrm{T}})$ and $\mathbf{A}^l_{\mathrm{sk}} = \frac{1}{2}(\mathbf{A}^l - (\mathbf{A}^l)^{\mathrm{T}})$. The skew symmetric matrix reduces to

$$c\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix},$$

where $c = m^l_{12} - m^l_{21}$ is the difference of the off-diagonal elements of $\mathbf{A}^l$. We now have

$$\psi(0,l)[k] = \mathbf{X}^0[k]^{*\mathrm{T}}(\mathbf{A}^l_{\mathrm{s}} + \mathbf{A}^l_{\mathrm{sk}})\mathbf{X}^0[k]\exp\left(\frac{j2\pi\lambda_l k}{N}\right)$$

The term $\mathbf{X}^0[k]^{*\mathrm{T}}\mathbf{A}^l_{\mathrm{s}}\mathbf{X}^0[k]$ is purely real while the term $\mathbf{X}^0[k]^{*\mathrm{T}}\mathbf{A}^l_{\mathrm{sk}}\mathbf{X}^0[k]$ is purely imaginary. We observe that the effect of the transformation matrix $\mathbf{A}^l$ on the second term is restricted to a scaling by a factor c. We can define a new measure κ, ignoring scale, for the sequence $\mathbf{X}^l$ in view l as

$$\kappa(l)[k] = \mathbf{X}^l[k]^{*\mathrm{T}} \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \mathbf{X}^l[k]. \tag{12}$$

It can be shown (see Appendix A) that

$$\kappa(l)[k] = |\mathbf{A}^l|\kappa(0)[k], \qquad 0 < k < N \tag{13}$$

where $|\mathbf{A}^l|$ is the determinant of $\mathbf{A}$. Eq. (13) gives a necessary condition for the sequences $\mathbf{X}^0$ and $\mathbf{X}^l$ to be two affine views of the same planar shape, or in other words, the values of the measure $\kappa(\cdot)$ in the two views should be scaled versions of each other. The $\kappa(\cdot)$ values for a contour are, thus, signatures of that contour. This extends to multiple views also. Consider the $M(N-1)$ matrix formed by the coefficients of the $\kappa(\cdot)$ measures for M different views

$$\Theta = \begin{bmatrix} \kappa(0)[1] & \cdots & \kappa(0)[N-1] \\ \kappa(1)[1] & \cdots & \kappa(1)[N-1] \\ \cdots & \cdots & \cdots \\ \kappa(M-1)[1] & \cdots & \kappa(M-1)[N-1] \end{bmatrix}$$

The necessary condition for matching of the planar shape in M views then reduces to

$$\mathrm{rank}(\Theta) = 1. \tag{14}$$

The view independent function that we want is $f(\cdot) = 0 \equiv \mathrm{rank}(\Theta) - 1 = 0$. It should be noted that this recognition constraint does not require correspondence across views and is valid for any number of views.

Since, the κ measures in the various views are only scaled versions of each other, if we normalize the κ measure terms in each view with respect to a fixed one then

$$\gamma(l)[k] = \kappa(l)[k]/\kappa(l)[p], \quad p \text{ is fixed}$$

$$= (|\mathbf{A}^l|\kappa(0)[k])/(|\mathbf{A}^l|\kappa(0)[p]) = \kappa(0)[k]/\kappa(0)[p]$$

$$\gamma(l)[k] = \gamma(0)[k] \tag{15}$$

The terms of the normalized κ measure—the γ measure are independent of the view. Hence, γ is an affine view invariant of a contour, whose computation does not need correspondence information across views.

Example. We consider four views of a dinosaur shown in Fig. 4. These views are related by affine transformations. Euclidean measures are not preserved under these transformations. However, we can still recognize them using the rank constraint. Discretization noise does introduce errors into the framework that make the rank constraint an approximation, but the constraint is robust and still works appreciably. To verify whether a matrix has rank r, we consider the ratio of rth to $(r+1)$th singular values of the matrix. This ratio is high if the matrix has rank r. To determine whether the rank of Θ is 1, we need to compare the ratio of the highest two singular values. The ratios of two highest singular values of Θ constructed from various two-view combinations of views shown in Fig. 4 are arranged in Table 1.

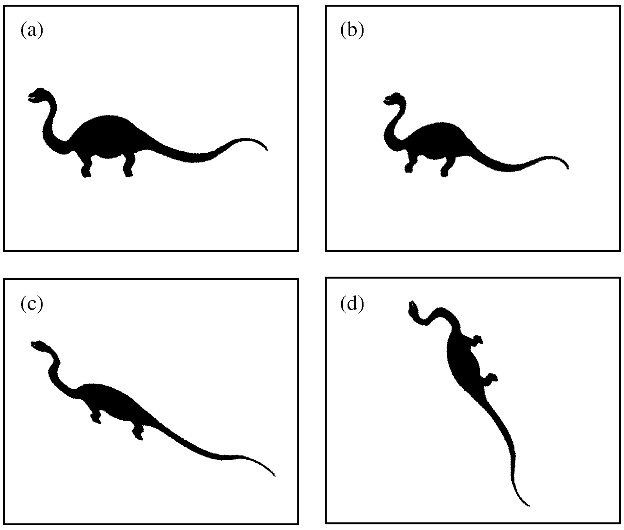

Fig. 4. Four affine transformed views of a dinosaur.

5. Polygons in multiple views

The recognition mechanism developed in Section 4 can be used for recognizing polygons under affine image-to-image homographies. In this section, we formulate a technique for recognizing polygons in multiple views, specifically when the optical axes of the cameras used for imaging are coincident. We represent the boundary as a sequence of lines instead of points. This is applicable when the objects are polygons or the representation is an approximation.

When a planar object is being imaged from multiple view points so that attention is focused on it, the location of the view points can be characterized using the azimuthal angle α in the horizontal plane, the elevation angle β from the vertical axis, the twist angle τ about its own axis, and the distance d from the origin (Fig. 5(a)). The image-to-image homography H_{12} relating two views given by $(\alpha^1, \beta^1, \tau^1, d^1)$ and $(\alpha^2, \beta^2, \tau^2, d^2)$ due to the object plane can be expressed up

Table 1
Ratio of the highest singular value to the second highest singular value of Θ constructed from various two-view combinations of views shown in Fig. 4

	Dinosaur 1	Dinosaur 2	Dinosaur 3	Dinosaur 4
Dinosaur 1	–	43,176.5	23,988.5	35,453.9
Dinosaur 2	43,176.5	–	25,733.7	35,352.6
Dinosaur 3	23,988.5	25,733.7	–	17,548
Dinosaur 4	35,453.9	35,352.6	17,548	–

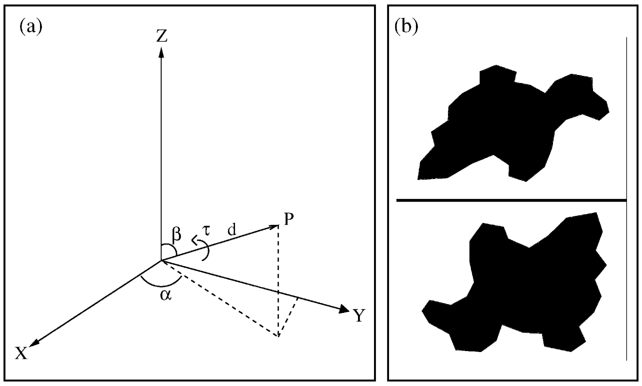

Fig. 5. (a) Characterizing a 3D point in terms of azimuthal, elevation and twist angles and the distance from the origin. (b) Two views of a polygon related by a homography whose transpose has the form of an affine homography.

to scale as

$$\mathbf{H} = \begin{bmatrix} d^1\begin{vmatrix} r^2_{11} & r^1_{21} \\ r^2_{12} & r^1_{22} \end{vmatrix} & -d^1\begin{vmatrix} r^2_{11} & r^1_{11} \\ r^2_{12} & r^1_{12} \end{vmatrix} & 0 \\ d^1\begin{vmatrix} r^2_{21} & r^1_{21} \\ r^2_{22} & r^1_{22} \end{vmatrix} & -d^1\begin{vmatrix} r^2_{21} & r^1_{11} \\ r^2_{22} & r^1_{12} \end{vmatrix} & 0 \\ A & B & C \end{bmatrix}$$

where

$$A = \begin{vmatrix} r^1_{21} & (d^1 r^2_{31} - d^2 r^1_{31}) \\ r^1_{22} & (d^1 r^2_{32} - d^2 r^1_{32}) \end{vmatrix}, \qquad B = \begin{vmatrix} r^1_{11} & (d^1 r^2_{31} - d^2 r^1_{31}) \\ r^1_{12} & (d^1 r^2_{32} - d^2 r^1_{32}) \end{vmatrix},$$

$$C = d^2\begin{vmatrix} r^1_{11} & r^1_{12} \\ r^1_{21} & r^1_{22} \end{vmatrix}$$

and r^1_{ij} and r^2_{ij} are the entries of the 3×3 rotation matrices that relate the vertical view to the view given by $(\alpha^i, \beta^i, \tau^i)$, respectively, for the views 1 and 2. This homography matrix has the special form of the transpose of an affine homography.

If the homography $\mathbf{H}$ relates points in two views, then $\mathbf{H}^{-\mathrm{T}}$ relates corresponding lines in two views. Therefore, if the homography relating corresponding points in two views has the form of transpose of affine, corresponding lines in the two views are related by affine image-to-image homographies. So, if we represent the boundary in a line space—making use of the edges of the polygons we can use the same approach as outlined in Section 4.2 for achieving recognition.

Example. Fig. 5(b) shows two views of a polygon, whose vertices are related by a homography whose transpose has the form of an affine homography. Hence, corresponding edges are related by affine homographies. All lines were normalized, so that the third component of each line was unity. The Θ matrix was computed from the line representation of the polygon. Its rank was essentially 1 as its highest singular values was 11,061.5 times bigger than the next highest singular value.

6. Applications and extensions

6.1. Numeral recognition

We demonstrate the applicability of polygonal approximation and the Fourier domain invariant to recognition of numerals. Numeral recognition has been conventionally addressed among the document image processing community. Popular OCRs are designed to address this problem only under similarity transformations. There are many other situations in image and video processing where the numerals are to be recognized under transformations more general than similarity. For instance, the alpha numerals on a number plate imaged from different viewpoints undergo projective transformations as can be seen in Fig. 6(a). We tried approximating the numerals in multiple views. Fig. 6(b) shows the result of approximating the numeral 8 in the two views shown in Fig. 6(a).

The numerals in the multiple views are related by projective homographies. Though the Fourier domain invariant has been derived for affine image-to-image homographies, we empirically found it to be applicable to most real life situations. We can solve the generic numeral recognition problem using both the approaches—polygon approximation and the Fourier domain invariant. We constructed a dataset of 10 numerals with 35 images, some of which are shown in Fig. 7.

Both approaches provide excellent results on numerals. The Fourier domain invariant had an accuracy of 96.81%, while polygonal approximation resulted in 94.70%. The numerals 6 and 9 are related by a rotation and since the algorithms are rotationally invariant, once the numerals were identified as 6 or 9, we had to take the horizontal projection of the numeral to distinguish them.

The Fourier domain invariant had difficulty in distinguishing between 1–7 and 0–4 pairs, while the polygonal approximation based methods were able to correctly identify them. The Fourier domain invariant

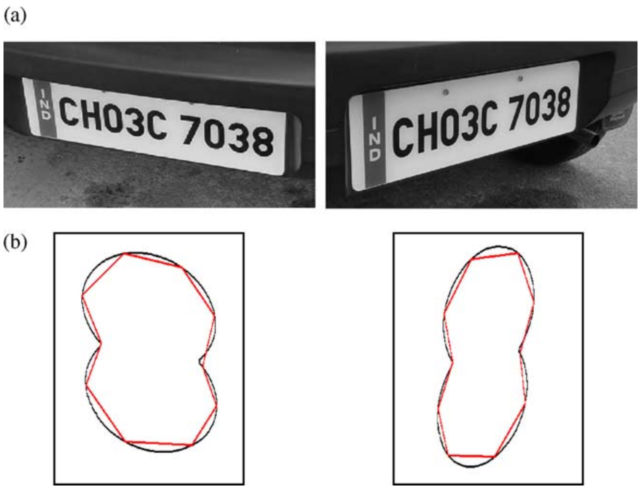

Fig. 6. Two views of a number plate.

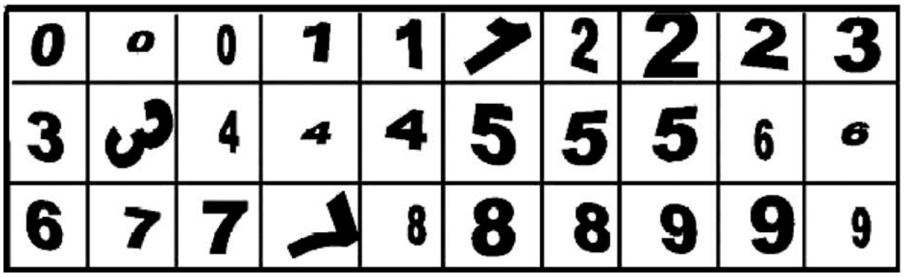

Fig. 7. Some inputs used in numeral recognition.

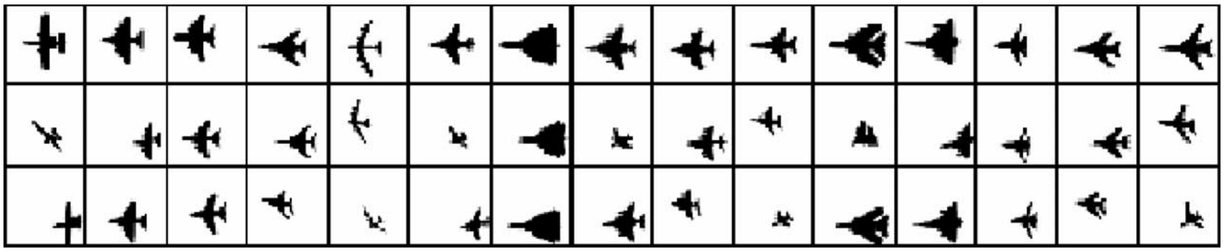

Fig. 8. Some inputs used in aircraft recognition.

works well when the boundary has more detail (presence of high frequencies). Thus, the problem in distinguishing 1 and 7 is to be expected.

6.2. Aircraft recognition

Most aircrafts can be recognized from their boundaries. Hence, shape based approaches for their recognition have received a lot of attention [25]. We considered 15 categories of aircrafts for the experimental verification of the proposed algorithms. Using projective transformations, these planar contours were transferred to 10 views each. Some of the sample inputs are shown in Fig. 8.

The polygonal approximation based recognition approach gave a recognition accuracy of 92.66%, while the Fourier domain invariant gave a recognition accuracy of 97.33%. The aircraft contour is complex and not amenable to easy polygonal approximation and so the performance of the Fourier domain invariant (which prefers complex contours) would be expected to be better.

For the Fourier domain invariant recognition mechanism, we consider the ratio of the highest two singular values of the measurement matrix Θ constructed from the views to be matched. When views of the same contour were matched, this ratio was more than 100. The ratio dropped to below 10 when attempting to match views of dissimilar contours. The Fourier domain invariant can be extended to compute correspondence across views. Once correspondence is established, we can then compute the homography between the views.

6.3. Computing correspondence across views

We can use a modified version of the invariant described in Section 4.2 to determine the shift λ_l which when applied to the boundary representation in view 0 would align all corresponding points in affine views l and 0 (the reference view).

The κ measured derived above correlates each vector Fourier coefficient with itself. The modified measure $\kappa_p^l(l)$ correlates all frequency terms with a fixed coefficient p. We define $\kappa_p^l(l)$ as

$$\kappa_p^l(l)[k] = (\mathbf{X}^l[k])^{*\mathrm{T}}\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}\mathbf{X}^l[p], \qquad p \neq 0.$$

Expanding using $\mathbf{X}^l[k] = \mathbf{A}^l\mathbf{X}^0[k]$ as before, we get

$$\kappa_p^l(l)[k] = |\mathbf{A}^l|\kappa_p^l(0)[k]\exp\left(-\frac{j2\pi\lambda_l(k-p)}{N}\right) \tag{16}$$

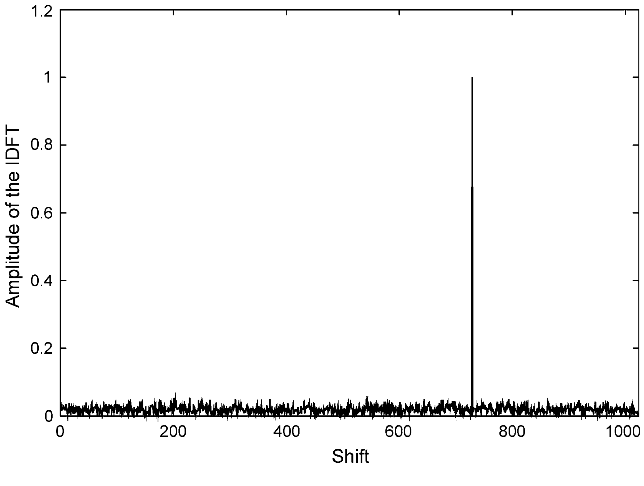

Fig. 9. Graph showing the amplitude of the IDFT of $\kappa_1^l(l)/\kappa_1^l(0)$ against the shift for views in Fig. 10(a) and (b). The shift aligning corresponding points in these views is given by the location of the peak, which is 729.

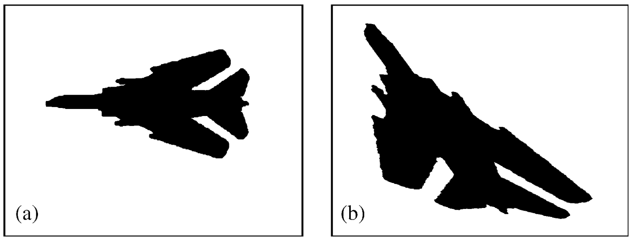

Fig. 10. Two views of an aircraft related by affine image-to-image homographies.

and

$$\frac{\kappa_p^l(l)[k]}{\kappa_p^l(0)[k]} = |\mathbf{A}^l|\exp\left(-\frac{j2\pi\lambda_l(k-p)}{N}\right) \tag{17}$$

Eq. 17 states that the quotient series $\kappa_p^l(l)/\kappa_p^l(0)$ would be a complex sinusoid, whose inverse Fourier transform would show a distinct peak at the shift that would align corresponding points. Fig. 9 shows the magnitude spectrum of the IDFT of $\kappa_1^l(l)/\kappa_1^l(0)$ against the shift for the views shown in Fig. 10(a) and (b). The IDFT magnitude spectrum peaks at 729, which is the correct shift value.

6.4. Computing image-to-image homographies

Given two views of a planar shape, we can use κ^l to compute the shift that would align corresponding points in two views. In practice, though, the sampling of the contour does not extract projections of the same points and so we cannot get corresponding points. A spatial domain approach to computing the homography by solving the set of equations derived from corresponding points would hence be prone to errors. We can solve the problem of homography computation more robustly in the Fourier domain by solving the linear equations obtained on taking the Fourier transform of Eq. (9). The homography relating views (a) and (b) of Fig. 11 is projective. On computing an affine homography between the two contours and projecting view (a) into view (b), we see that the reprojected view is a very good approximation of the target view. The contours are shown in Fig. 12.

(a) (b)

Fig. 11. Two views of a road sign.

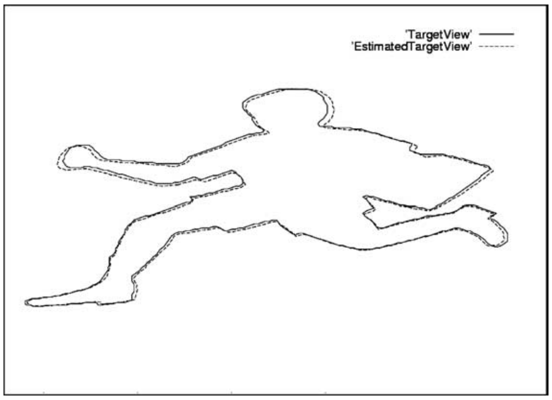

Fig. 12. Computation of homography. The homography between views in Fig. 11(a) and (b) was computed and used to project (a) into (b). The target view and the projected view are shown above.

7. Conclusions and future work

We presented two approaches to recognize discrete planar contours in this paper. The first involves computing a piecewise parametric representation of the contour. Specifically, we used polygonal approximation and piecewise conic approximation. The approximation is projective invariant. Thus, the piecewise parametric representations generated from multiple views are isomorphic to each other. The second approach involved computing an invariant directly from the contour from a Fourier domain representation. The affine invariant we derived has excellent recognition properties and work well on reasonable projective transformed contours also. We demonstrated the application of these approaches on real-life problems such as numeral recognition and aircraft recognition.

The two approaches are, in a sense, complementary in nature. The Fourier domain approach requires sufficient frequency components in the contour and works well on irregular contours. It tends to perform poorly when the contour has many smooth sections. The piecewise parametric representation, on the other hand, works exceedingly well in these cases. They have difficulty in approximating the contour by parametric sections if the contour is highly irregular. Together, they can provide excellent recognition of discrete planar contours in multiple views. We are presently working on an algorithm that combines these two approaches to get robust recognition of all types of contours.

Acknowledgements

This work was partially supported by the Advanced Data Processing Research Institute (ADRIN), Department of Space, Government of India.

Appendix A

Let

$$\mathbf{R}_\perp = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}.$$

The function κ defined in Section 4.2 can be written as

$$\begin{aligned}
\kappa(l)[k] &= \mathbf{X}^l[k]^{*\mathrm{T}}\mathbf{R}_\perp\mathbf{X}^l[k].\kappa(l)[k] \\
&= \left(\mathbf{A}^l\mathbf{X}^0[k]\exp\left(\frac{j2\pi\lambda_l k}{N}\right)\right)^{*\mathrm{T}} \\
&\quad \mathbf{R}_\perp\left(\mathbf{A}^l\mathbf{X}^0[k]\exp\left(\frac{j2\pi\lambda_l k}{N}\right)\right) \\
&= \mathbf{X}^0[k]^{*\mathrm{T}}(\mathbf{A}^l)^{\mathrm{T}}\exp\left(\frac{-j2\pi\lambda_l k}{N}\right)\mathbf{R}_\perp\mathbf{A}^l\mathbf{X}^0[k] \\
&\quad \exp\left(\frac{j2\pi\lambda_l k}{N}\right) \\
&= \mathbf{X}^0[k]^{*\mathrm{T}}(\mathbf{A}^l)^{\mathrm{T}}\mathbf{R}_\perp\mathbf{A}^l\mathbf{X}^0[k] = |\mathbf{A}^l|\mathbf{X}^0[k]^{*\mathrm{T}}\mathbf{R}_\perp\mathbf{X}^0[k] \\
&= |\mathbf{A}^l|\kappa(0)[k]
\end{aligned}$$

which is the result in Eq. (13)

References

[1] H.C. Longuet-Higgins, A computer algorithm for reconstructing a scene from two projections, Nature 293 (1981) 133–135.

[2] O. Faugeras, Q. Luong, The Geometry of Multiple Images, MIT Press, Cambridge, MA, 2001.

[3] R. Hartley, A. Zisserman, Multiple View Geometry, Cambridge University Press, Cambridge, 2000.

[4] R. Hartley, Lines and points in three views: an integrated approach, Proceedings of ARPA Image Understanding Workshop II (1994) 1009–1016.

[5] A. Shashua, Algebraic functions for recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 16 (1995) 778–790.

[6] D.P. Huttenlocher, S. Ullman, Object recognition using alignment, Proceedings of International Conference on Computer Vision (1987) 102–111.

[7] T. Pavlidis, Structural Pattern Recognition, Springer, New York, 1977.

[8] A.K. Jain, Fundamentals of Digital Image Processing, Prentice-Hall, Englewood Cliffs, NJ, 1989.

[9] S. Ullman, R. Basri, Recognition by linear combination of models, IEEE Transactions on Pattern Analysis and Machine Intelligence 13 (1991) 992–1006.

[10] C. Zahn, R. Roskies, Fourier descriptors for plane closed curves, IEEE Transactions on Computers C-21 (1972) 269–281.

[11] J.L. Mundy, A. Zisserman, Geometric Invariance in Computer Vision, MIT Press, Cambridge, MA, 1992.

[12] L.J. Van Gool, T. Moons, E. Pauwels, A. Oosterlinck, Geometric invariance in computer vision, Ch. Semi-Differential Invariants, MIT Press, Cambridge, MA, 1992.

[13] E. Pauwels, T. Moons, L. VanGool, P. Kempenaers, A. Oosterlinck, Recognition of planar shapes under affine distortion, International Journal of Computer Vision 14 (1995) 49–65.

[14] K. Arbter, W. Snyder, H. Burkhardt, G. Hirzinger, Application of affine-invariant Fourier descriptors to recognition of 3D objects, IEEE Transactions on Pattern Analysis and Machine Intelligence 12 (1990) 640–647.

[15] S. Kuthirummal, C.V. Jawahar, P.J. Narayanan, Multiview constraints for recognition of planar curves in Fourier domain, Indian Conference on Computer Vision, Graphics and Image Processing (2002) 323–328.

[16] M. Pawan Kumar, S. Goyal, C.V. Jawahar, P.J. Narayanan, Polygonal approximation of closed curves across multiple views, Indian Conference on Computer Vision, Graphics and Image Processing (2002) 317–322.

[17] M. DeHaemer Jr., M.J. Zyda, Simplification of objects rendered by polygonal approximations, Computers and Graphics 15 (2) (1991) 175–184.

[18] P. Shirley, A.A. Tuchman, Polygonal approximation to direct scalar volume rendering, in: Proceedings of San Diego Workshop on Volume Visualization, Computer Graphics 24 (5) (1990) 63–70.

[19] M.K. Leung, Y. Yang, Dynamic strip algorithm in curve fitting, in: Computer Vision,Graphics and Image Processing 23 (1990) 69–79.

[20] I. Anderson, J. Bezdek, Curvature and tangential deflection of discrete arcs, T-PAMI 6 (1984) 27–40.

[21] M. Salotti, An efficient algorithm for the optimal polygonal approximation of digitized curves, Pattern Recognition Letters 22 (2) (2001) 215–221.

[22] F. Bookstein, Fitting conic sections to scattered data, in: Computer Graphics and Image Processing 9 (1979) 56–71.

[23] R. Mohr, Projective Geometry and Computer Vision, World Scientific, Singapore, 1993.

[24] S. Kuthirummal, C.V. Jawahar, P.J. Narayanan, Planar shape recognition across multiple views, International Conference on Pattern Recognition (2002) 456–459.

[25] T.P. Wallace, P.A. Wintz, An efficient three-dimensional aircraft recognition algorithm using normalized Fourier descriptors, Computer Graphics and Image Processing 13 (2) (1980) 99–126.