Thomas H. Reiss

Recognizing Planar Objects Using Invariant Image Features

Springer-Verlag
Berlin Heidelberg New York
London Paris Tokyo
Hong Kong Barcelona
Budapest

Series Editors

Gerhard Goos
Universität Karlsruhe
Postfach 69 80
Vincenz-Priessnitz-Straße 1
W-7500 Karlsruhe, FRG

Juris Hartmanis
Cornell University
Department of Computer Science
4130 Upson Hall
Ithaca, NY 14853, USA

Author

Thomas H. Reiss
Engineering Department, Cambridge University
Trumpington Street, Cambridge CB2 1PZ, U.K.

CR Subject Classification (1991): I.3-5

ISBN 3-540-56713-5 Springer-Verlag Berlin Heidelberg New York
ISBN 0-387-56713-5 Springer-Verlag New York Berlin Heidelberg

Typesetting: Camera ready by author/editor
45/3140-543210 - Printed on acid-free paper

Summary

Given a familiar object extracted from its surroundings, we humans have little difficulty in recognizing it irrespective of its size, position and orientation in our field of view. Changes in lighting and the effects of perspective also pose no problems. How do we achieve this, and more importantly, how can we get a computer to do this? One very promising approach is to find mathematical functions of an object's image, or of an object's 3D description, that are invariant to the transformations caused by the object's motion. This book is devoted to the theory and practice of such invariant image features, so-called *image invariants*, for planar objects.

Following the introduction in chapter 1, the book discusses features that are invariant to image translations, rotations, to changes in size and in contrast, with particular attention being paid to the effect of using discrete images rather than continuous ones. The next chapter presents a tutorial introduction to the theory of algebraic invariants which lies at the heart of two important types of invariant features: moment invariants for affine transformations, and projective invariants for perspective transformations. Chapter 4 is devoted entirely to features invariant to affine transformations; the theory behind moment-based invariants, Fourier descriptors and differential techniques is presented, along with a novel technique based on correlations, and results of experiments on the stability of coarsely sampled images are discussed. Chapter 5 goes one step further and covers features invariant to perspective transformations, summarizing work on both differential and global invariants. The penultimate chapter, chapter 6, shows how invariant features can be used to recognize objects that have been partially occluded. A thorough treatment of the 'geometric hashing' method is given, followed by some novel methods of 'back-projection' which allow one to verify whether the hypothesized object really is in the image. Many authors claim that moment invariants cannot be used under partial occlusion; this is not so, and a number of schemes for their use are presented. Not only can they be used, but they have some significant advantages over other invariant features, a fact that is backed up by experiments. The final chapter contains a summary and conclusions.

Acknowledgements

I would like to thank Anil Kokoram, Michael Lai and numerous others for ensuring the smooth running of the Novell network in the lab, Mark Wright for helping me to acquire real images, Nick Kingsbury for some suggestions on how to improve the presentation of the material and Roberto Cipolla for proof reading so swiftly and accurately. I would also like to thank Peter Rayner and the United Kingdom Science and Engineering Research Council for giving me the opportunity to carry out the research contained within.

Contents

Chapter 1

Introduction

1.1 Scope

Research in computer vision is aimed at enabling computers to recognize objects without human intervention. Applications are numerous, and include automated inspection of parts in factories, detection of fires at high-risk sites and robot vision, especially for autonomous robots. For the sake of convenience, the task is usually broken up into two stages, 'low-level' vision and 'high-level' vision. Low-level vision involves extracting significant features from the image, such as the outline of an object or regions with the same texture, and often involves segmenting the image into separate 'objects'. The task of high-level vision is then to recognize these objects.

The following is concerned with high-level vision, in particular with finding properties of an image which are invariant to transformations of the image caused by moving an object so as to change its perceived position and orientation, and in some cases its brightness. The idea of invariance arises from our own ability to recognize objects irrespective of such movement — if we look at a book from a number of different orientations, we have no difficulty in recognizing it as a book each time: we can say that a book has properties which are invariant to its size, position and orientation. Finding mathematical functions of an image that are invariant to the above transformations would thus provide us with a technique for recognizing objects using computers, as well as providing us with a possible model for part of human vision. If we wish to recognize an object using a computer, and we assume the computer has stored the models (or example views) of the objects to be recognized in its memory, and the object to be recognized corresponds to one of these models, the straightforward approach searches sequentially through the computer's memory, trying out each model and seeing whether it can be positioned in such a way as to produce an image that matches that of the object to be recognized, until a good match is found. Clearly, this is computationally intensive; ideally, we would like to be able to extract the correct model directly from the information contained in the image — this is precisely what the so-called *image invariants* allow us to do.

Images are the projection of the three-dimensional world onto a two-dimensional (planar) surface, be it the retina or an array of sensors in a video (CCD) camera. Burns *et al.* [1] have proved that one cannot compute an invariant function of the image coordinates of a set of general points in three dimensions (3D) from a single view; one requires at least two views. If one restricts oneself to planar or near-planar objects however, one can obtain a large number of invariants based on a single view, as we

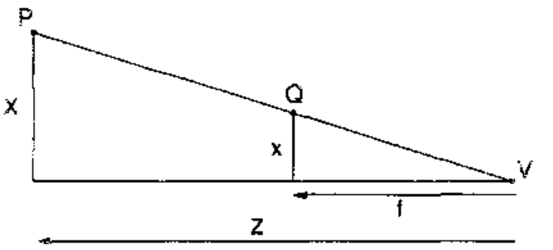

Figure 1.1: The perspective camera

Consider the X-component of a point P in 3D. V is the point of projection, the image plane is defined by $Z = f$. Q is the image point corresponding to P. $x = fX/Z$.

will see in the following chapters. Although the theory only applies to flat objects, the resulting features have been used to recognize non-planar objects, such as aircraft, successfully (see references [2, 3, 4, 5]) and will also work well for near-planar objects such as machine parts or the tools used in the experiments described in chapter 6. If one uses two views from an uncalibrated stereo camera, a number of invariants for non-planar points and curves exist — see references [6, 7, 8, 9] and in particular the excellent collection of articles edited by Mundy & Zisserman [10].

1.2 Viewing transformations

The motion of a solid object in 3D is governed by six parameters, three for translations and three for rotations. This section shows how image points are affected by these six parameters, that different views of coplanar object points are related by planar projective transformations and that, if the distance of the coplanar object points from the camera (their *depth*) does not vary much compared with their average depth, the planar projective transformation can be approximated by the affine transformation.

A very good approximation to image formation in a real camera is given by the *perspective camera* model, in which points are projected from 3D onto the image plane so that all the rays joining object and corresponding image points pass through a single point, called the point of projection. If we choose our 3D coordinates so that the origin coincides with the point of projection, the Z-axis is perpendicular to the image plane and points away from the camera, and the image plane is defined by $Z = f$ (see figure 1.1), then the image coordinates (x, y) of a 3D point (X, Y, Z) are given by

$$x = f\frac{X}{Z}; \qquad y = f\frac{Y}{Z}. \tag{1.1}$$

Let $\mathbf{p}^T = [X \;\; Y \;\; Z]$ and $\mathbf{p}'^T = [X' \;\; Y' \;\; Z']$ be a rotated and translated version, then

$$\mathbf{p}' = \mathbf{R}\mathbf{p} + \mathbf{t},$$

where $\mathbf{R}$ is a 3×3 rotation matrix and $\mathbf{t}$ is the translation vector. This can be written in full as

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}. \qquad (1.2)$$

Combining this with (1.1) allows one to write the image coordinates as a function of the original 3D coordinates and the motion parameters.

If we restrict ourselves to planar objects, Z is related to X and Y: in general, object points lie in a plane $Z = aX + bY + c$ for some constants a, b and c. Putting this into (1.2) gives

$$\begin{aligned} X' &= (r_{11} + ar_{13})X + (r_{12} + br_{13})Y + (cr_{13} + t_1) \\ &= a_{11}X + a_{12}Y + a_{13}, \end{aligned}$$

where $a_{11} = r_{11} + ar_{13}$ etc. Doing the same for Y' and Z' allows us to write (1.2) as

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \quad \text{or} \quad \mathbf{p}' = \mathbf{A}\mathbf{q}. \qquad (1.3)$$

If we set $f = 1$, the image coordinates of the transformed point are given by $x' = X'/Z'$, $y' = Y'/Z'$, from which we see that arbitrarily scaling $\mathbf{A}$ does not affect the image coordinates, which in turn allows one to set $a_{33} = 1$.

The above uses the actual 3D coordinates (X, Y, Z) and (X', Y', Z') of a point and its transformed version; to find invariant functions we must assume no knowledge of 3D coordinates, so we must find the form of the transformation linking image points in one view to those in another. In fact, they are related by a *planar projection*, which is easily proved using *homogeneous coordinates*. Let the image coordinates (x', y') be represented in homogeneous coordinates by (h'_X, h'_Y, h'_Z), with $x' = h'_X/h'_Z$ and $y' = h'_Y/h'_Z$ (we are *not* assuming that $h'_X = X'$, $h'_Y = Y'$ or $h'_Z = Z'$). Furthermore, let $\mathbf{h}'^T = [h'_X \;\; h'_Y \;\; h'_Z]$. It is clear that $\mathbf{h}' = \alpha\mathbf{p}'$ for some non-zero scalar α, from which we see that $\mathbf{h}' = \mathbf{A}_1\mathbf{q}$, $\mathbf{A}_1 = \alpha\mathbf{A}$. Let $\hat{\mathbf{h}}$ represent a second view of the planar points, then $\hat{\mathbf{h}} = \mathbf{A}_2\mathbf{q}$. If we define $\mathbf{B} = \mathbf{A}_2\mathbf{A}_1^{-1}$ we see that $\hat{\mathbf{h}} = \mathbf{B}\mathbf{h}'$:

$$\begin{bmatrix} \hat{h}_X \\ \hat{h}_Y \\ \hat{h}_Z \end{bmatrix} = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix} \begin{bmatrix} h'_X \\ h'_Y \\ h'_Z \end{bmatrix},$$

which defines a planar projection (see figure 1.2). As before, we can set $b_{33} = 1$. Since there are only six degrees of freedom, but planar projection has eight free parameters, we see that perspective forms a subset of planar projection. Since the choice of h'_Z is at our disposal, we can set $h'_Z = 1$, which means that the image coordinates $(\hat{x}, \hat{y})$ and (x, y) are related to one another by:

$$\hat{x} = \frac{b_{11}x + b_{12}y + b_{13}}{b_{31}x + b_{32}y + 1}, \qquad \hat{y} = \frac{b_{21}x + b_{22}y + b_{23}}{b_{31}x + b_{32}y + 1}. \qquad (1.4)$$

Planar projection forms a group [9] and has a number of invariants which will be discussed in chapters 3 and 5.

We can now use equation (1.3) to show that the above planar projective transformation can be approximated by an affine transformation when a planar object's

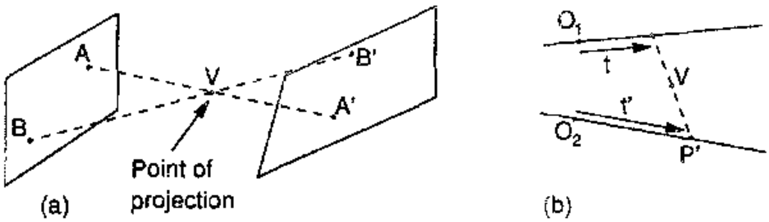

Figure 1.2: Planar and linear projection.

(a) Projection of points on one plane to points on another. (b) Projection of points on one line to points on another.

depth is small compared with its distance from the camera. Assume we have a set of coplanar points (X_i', Y_i', Z_i'), $i = 1, \ldots, N$, for which $Z_i' = Z' + \delta Z_i'$. The image coordinates $x_i' = X_i'/Z_i'$, $y_i' = Y_i'/Z_i'$ are approximately given by

$$x_i' = \frac{X_i'}{Z'}, \quad y_i' = \frac{Y_i'}{Z'} \quad \text{if} \quad Z' \gg \delta Z_i'.$$

Thompson & Mundy [11] use the rule of thumb that $(Z'_{\max} - Z'_{\min})/Z'$ should be less than 0.1 for the affine approximation to hold, with $Z'_{\max}$ being the largest value of Z_i' and $Z'_{\min}$ the smallest. If we let $s = Z'$, (1.3) becomes

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & s \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}. \tag{1.5}$$

Defining the elements c_{ij} of the 3×3 matrix C_1 by $c_{ij} = a_{ij}/s$, so that (1.5) becomes $\mathbf{p}' = \mathbf{C}_1\mathbf{q}$, letting $\mathbf{C}_2$ define a second view and $\mathbf{D} = \mathbf{C}_2\mathbf{C}_1^{-1}$ allows one to show that the image coordinates $\hat{\mathbf{h}}^T = [\hat{x} \;\; \hat{y} \;\; 1]$ of the second view are related to those of the first view $\mathbf{h}' = [x' \;\; y' \;\; 1]$ by $\hat{\mathbf{h}} = \mathbf{D}\mathbf{h}'$:

$$\begin{bmatrix} \hat{x} \\ \hat{y} \\ 1 \end{bmatrix} = \begin{bmatrix} d_{11} & d_{12} & d_{13} \\ d_{21} & d_{22} & d_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} \quad \text{— an affine transformation.}$$

This can alternatively be written as

$$\begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix} = \begin{bmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \end{bmatrix} \begin{bmatrix} x' \\ y' \end{bmatrix} + \begin{bmatrix} d_{13} \\ d_{23} \end{bmatrix}. \tag{1.6}$$

Since the affine transformation is linear, invariant features are much easier to find than in the case of projective transformations. Examples of affine transformations are shown in figure 1.4. Note that parallel lines remain parallel under such transformations.

Just as a linear transformation in 3-D is equivalent to planar (2-D) projection, so a linear transformation in 2-D is equivalent to a 1-D projection of points on one line onto another line — see figure 1.2 (b).

$$\text{1-D projection:} \qquad t' = \frac{\alpha t + \beta}{\gamma t + \delta}, \qquad \alpha, \beta, \gamma, \delta \quad \text{real.}$$

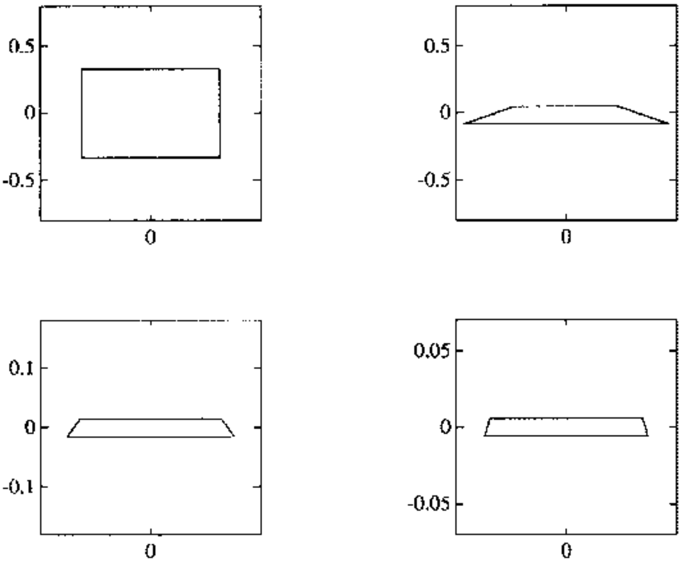

Figure 1.3: Perspective views of a rectangle.

The projective transformation tends towards an affine one as the object recedes into the distance. **Top:** *the rectangle's centre is 3 units from the point of projection; head on and at 80°.* **Bottom:** *12 and 30 units away.*

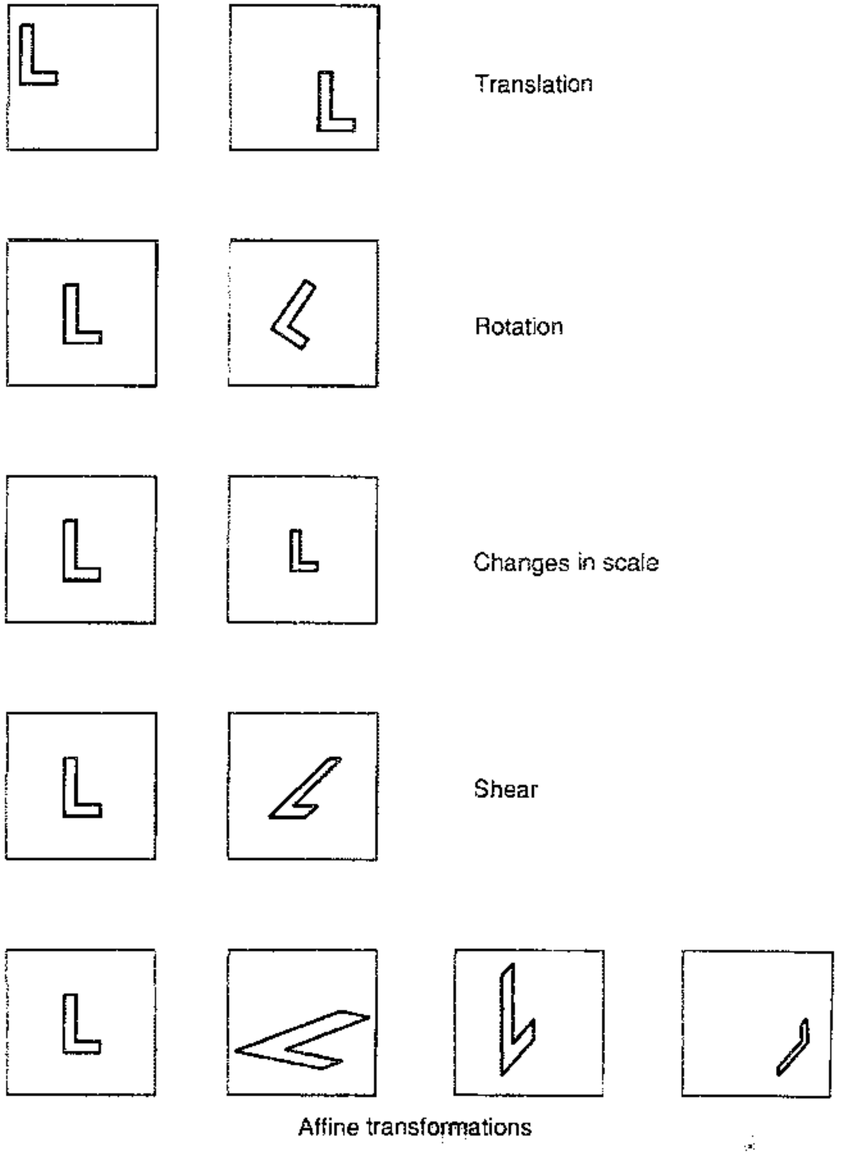

Figure 1.4: Examples of affine image transformations.

In homogeneous coordinates, with $t = x/y$ and $t' = x'/y'$, we have

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \alpha & \beta \\ \gamma & \delta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}.$$

The 1D projective transformation has three degrees of freedom, so we can arbitrarily set $\delta \equiv 1$. Any three distinct collinear points can be projected to any other such triple of points under a 1D projective transformation.

Further approximations are possible under constrained viewing conditions; for instance, if an object is viewed from a fixed camera and is constrained to lie on a flat surface parallel to the image plane near its intersection with the optical (Z) axis (the *fronto-parallel* configuration), the affine approximation reduces to rotations, translations and scale changes. This is relevant if one would like to automate inspection of objects on a conveyor belt for example. An application in which one is only interested in rotation invariants is automatic fault detection in the heads of bolts, which slide head-up along two parallel guides into a fixed position relative to the camera, save for possible rotations.

Examples of invariant features in this case are the angle between two lines, which is unchanged by image translations, rotations and changes in scale, and the ratio of distances along a line or the ratio of two areas, both of which are invariant to affine transformations. Clearly, any function which is invariant to a given set of transformations is invariant to any subset of transformations projective invariants are also invariant to affine transformations etc.

1.3 Overview

An object is said to be partially occluded when part of it is obscured from view by another object. Anyone glancing around a room will quickly see that partial occlusion is the norm rather than the exception. Our goal is to find robust techniques for recognizing partially occluded near-planar objects. To reach it, we must first obtain a thorough understanding of invariant features for unoccluded objects, which is the task of chapters 2 to 5. These chapters build up successively from invariance to simple transformations in chapter 2 to invariance to full planar projection in chapter 5: chapter 2 looks at invariance to translations, rotations, changes in scale, changes in contrast and combinations of these, first for ideal (continuous) images and then for discrete ones (as processed by computers); chapter 3 presents a tutorial introduction to the theory of algebraic invariants, which is fundamentally important when dealing with affine invariants, the subject of chapter 4, and projective invariants, the subject of chapter 5. Chapter 4 gives a definitive account of methods for obtaining invariance to affine transformations using *moments*, as well as summarizing alternative techniques and presenting a novel one based on *correlations*. Chapter 5 shows how to apply the results of chapter 3 to recognizing perspective views of objects, summarizes some of the authors results as well as discussing alternative methods. Finally, chapter 6 discusses schemes for using the invariant features of chapters 4 and 5 to recognize partially occluded objects irrespective of viewpoint, and chapter 7 provides a summary and conclusions.

To the author's knowledge, no monograph on the subject of using invariant features to recognize objects has appeared at the time of writing[1], with most results having appeared in conferences and workshops rather than journals (see in particular the collection of papers based on a workshop in 1991, reference [10]). Hence chapters 2 to 5, although presenting quite a number of new results along the way, are written to provide a comprehensive and up-to-date summary of research performed to date.

As mentioned above, chapters 2 to 5 are written to provide an understanding of invariant functions to aid in our goal of recognizing partially occluded objects. One important aspect is their robustness to image distortion. Many authors consider the effects of adding Gaussian noise to the image to test the robustness of invariants (e.g. [14, 15, 16]), but in most cases the amount of noise in the image is almost imperceptible; a more useful test of robustness is to see how the invariant features react to distortions of the object's shape, especially those caused by the discrete nature of the image when viewing distant or small objects. Theoretical results for rotation invariance are presented in chapter 2, including a novel family of invariant features, and experimental results are presented in both chapters 2 and 4; the main conclusion is that the so-called moment invariants are more robust than other invariants, contrary to beliefs voiced in the literature on the subject (see for example reference [17]).

When using image features to recognize objects, one would like them to provide good discrimination between different objects. In the case of image invariants, one is first interested in invariance to geometric transformations; having found invariant features, one must then investigate whether they provide discrimination. The experiments in chapters 2 and 4 also examine this; again, the moment invariants are shown to perform well.

If we are to recognize partially occluded objects, the information required for recognition must be available locally as well as globally, and the question naturally arises as to whether one can use the image invariants derived in chapters 2 to 5 to recognize partially occluded objects. The answer is in the affirmative, and is the subject of chapter 6. Numerous authors claim that moment invariants cannot be used under partial occlusion (see for example [18]), but this is not so: a number of schemes for their use are presented, and it is shown that they have significant advantages over other invariants, both in theory and in practice, the latter being demonstrated by experiment.

The next section ends this introductory chapter with a brief discussion of the image primitives that are used to generate invariant functions in chapters 4 to 6.

1.4 Image primitives

In order to compute invariant functions of an image, one often needs to extract certain features from the image, called primitives, which are then used in the construction of invariant functions. For instance, if one is using the angle between lines as an invariant then the lines are the primitives. As we will see in the following chapters, one can either compute invariant functions based on the intensity function of the image or based on the shape of the image boundary (in both cases one assumes that the object

[1]Wechsler's review [12] discusses some simple invariants and Kanatani's book [13] contains an extensive treatment of image invariants, but he only considers invariance to camera rotations and does not attempt to obtain invariance to general object motion relative to the camera. Furthermore, as discussed in chapter 5, some of his analysis is incorrect.

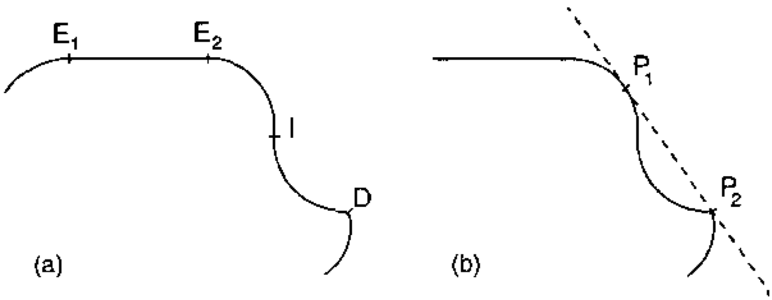

Figure 1.5: Finding reference points on a curve.

(a) E_i are end points of a line; I is a point of inflexion; D is a sharp discontinuity in curvature. Only D is a robust feature point in this case. (b) P_1 and P_2 are two points where the curve's convex hull leaves the curve itself; they are generally robust reference points, and are called bitangent points.

has been correctly separated from the background). Many methods, especially those that seek to recognize partially occluded objects, use points and lines as primitives; below we will briefly look at how these primitives can be extracted from the object's boundary.

The standard approach to finding reference points relies on differential properties of an object's boundary. Discontinuities in curvature and points of inflexion of planar curves are invariant under projection, but not always robust: the end point of a straight line is a discontinuity in curvature that is difficult to detect when the line goes into a curve of low curvature; similarly, points of inflexion can often be confused with a short line. A more robust technique uses ***bitangent points*** where two distinct points on the boundary share the same tangent (see figure 1.5) [9]. Also robust are points where a discontinuity in curvature is linked with a fairly sharp change in direction. These latter points are usually extracted by detecting large extrema of curvature — although extrema are theoretically not stable under projection, in practice they are [19, 20].

Further reference points can be obtained by noting where tangents constructed using the above points intersect with other tangents or with the boundary [9] (the latter being more robust against noise). As pointed out by Lamdan *et al.* [21], almost all near-planar objects have concavities, so this approach can be used in most cases. Forsyth *et al.* [9] present a generally robust method of extracting four reference points from a concavity that does not rely on extrema of curvature (figure 1.6(a)). The method fails when there is a discontinuity of curvature at points A or D such that the curvature has a different sign on either side of the discontinuity (point A in figure 1.6b), but in this case one can extract the tangents of the boundary on both sides of the discontinuity (figure 1.6b).

Points and lines will be used as image primitives in chapter 4 on affine invariants and in chapter 5 on projective invariants; although they can be used for the simpler transformations discussed in chapter 2, alternative global techniques will be used instead.

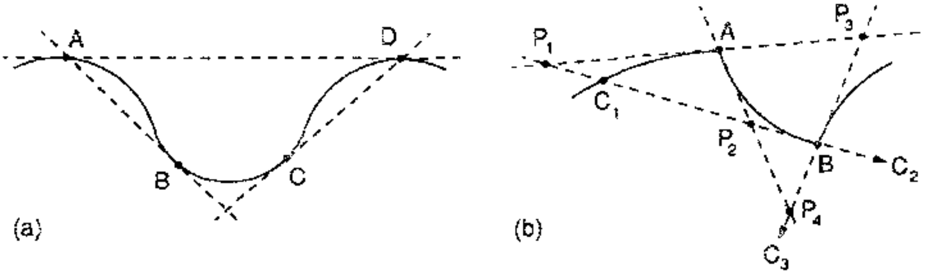

Figure 1.6: Using concavities to obtain projective invariants.

(a) A and D are points where the convex hull meets the boundary. B and C are the points where lines through A and D are tangent with the concave part of the boundary. (b) One can use the tangents at either side of a discontinuity in curvature to obtain a number of reference points. C_2 and C_3 are points where the tangents intersect with the boundary on the other side.