Computer Vision CITS4240

School of Computer Science & Software Engineering
The University of Western Australia

Binary Images

We will first consider how to model the effect of noise in images, and show that noise can be reduced by taking the average of several images. Clearly, such integration is precisely what biological systems do. We will then go on to study the simplest of images, namely binary images, and determine those measurements that can be used to distinguish one object from another in a binary image. Most of these notes are adapted from Horn's book on "Robot Vision" [2].

Measurements and Noise

No imaging system is perfect. Noise is introduced into the imaging process via the use of real lenses and cameras that differ in operation from the pinhole camera model. Moreover, lighting and atmospheric conditions can also affect the resulting image. In addition, digital images suffer deviations in image values introduced by sampling. Thus, measurements are affected by fluctuations in the signal being measured, and these fluctuations are described according to some probability distribution, $p(x)$.

Since $p(x)$ is a probability distribution, it always satisfies

$$p(x) \geq 0, \text{for all } x$$

and

$$\int_{-\infty}^{\infty} p(x)dx = 1.$$

The *mean* or first moment of the distribution μ is given by

$$\mu = \frac{\int_{-\infty}^{\infty} xp(x)dx}{\int_{-\infty}^{\infty} p(x)dx},$$

but

$$\int_{-\infty}^{\infty} p(x)dx = 1,$$

implying

$$\mu = \int_{-\infty}^{\infty} x p(x) dx.$$

The spread of the distribution is given by the second moment or *variance*:

$$\text{variance} = (\text{std.deviation})^2 = \sigma^2 = \int_{-\infty}^{\infty} (x-\mu)^2 p(x) dx.$$

The *cumulative probability distribution*

$$P(x) = \int_{-\infty}^{x} p(t) dt$$

tells us the probability that the measurement will be less than or equal to x. Thus, the probability density distribution is just the derivative of the cumulative probability distribution, that is, $p(x) = P'(x)$.

One way to improve accuracy is to average several measurements, assuming that the 'noise' in them will be independent and tend to cancel out. To see how this is indeed the case, we consider the following analysis.

Let $x = x_1 + x_2$, the sum of two independent variables with probability distributions $p_1(x_1)$ and $p_2(x_2)$. Then what can we say about $p(x)$, the probability distribution of x?

We first look at the probability of getting a value between x and $x+\delta x$. If the sum x_1+x_2 is between x and $x+\delta x$, and we are given a value of x_2, then x_1 must be between $x - x_2$ and $x + \delta x - x_2$. The probability of this occurring is $p_1(x - x_2)\delta x$.

But x_2 can also take on a range of values. The probability that x_2 lies between a particular x_2 and $x_2 + \delta x_2$ is

$$p_2(x_2)\delta x_2.$$

To get the probability that the sum lies between x and $x + \delta x$ we have to integrate the product over all x_2. Thus

$$p(x)\delta x = \int_{-\infty}^{\infty} p_1(x - x_2)\delta x p_2(x_2) dx_2,$$

or

$$p(x) = \int_{-\infty}^{\infty} p_1(x - t) p_2(t) dt,$$

where t is a dummy variable of integration.

By a similar argument we can show symmetrically that

$$p(x) = \int_{-\infty}^{\infty} p_2(x - t) p_1(t) dt.$$

This is called the *convolution* of p_1 and p_2 and is denoted by $p_1 \otimes p_2$.

We can use this result to prove that the mean of the sum of several random variables is equal to the sum of the means, and the variance of the sum is the sum of the variances.

Taking multiple measurements of a variable

If we calculate the average of N independent measurements each having mean μ and standard deviation σ, then

$$\overline{x} = \frac{1}{N} \sum x_i.$$

So the average $\overline{x}$ will have mean $\mu = \frac{1}{N} N\mu$ and standard deviation $\frac{\sigma}{\sqrt{N}}$.

This latter result is because the variance of the sum of the distributions is just $N\sigma^2$, and so the standard deviation of the sum is $\sqrt{N}\sigma$. Thus the distribution $\overline{x}$ has standard deviation $\frac{\sigma}{\sqrt{N}}$ and hence we obtain a more accurate measurement by taking the average of N independent measurements.

The usual probability distribution we use to model noise in an image is the normal or Gaussian distribution with mean μ and standard deviation σ given by

$$p(x) = \frac{1}{\sqrt{2\pi\sigma}}.e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2}.$$

Of course, digital images are *sampled* versions of continuous or analog images, and the sampling is done both spatially and in the luminance values (where it is known as quantization).

Binary Images

Binary images are images that have been quantized to two values, usually denoted 0 and 1, but often with pixel values 0 and 255, representing black and white.

Binary images are used in many applications since they are the simplest to process, but they are such an impoverished representation of the image information that their use is not always possible. However, they are useful where all the information you need can be provided by the silhouette of the object and when you can obtain the silhouette of that object easily.

Some sample application domains include

- identifying objects on a conveyor — for example, sorting chocolates!
- identifying orientations of objects, and
- interpreting text.

Sometimes the output of other image processing techniques is represented in the form of a binary image, for example, the output of edge detection can be a binary image (edge points and non-edge points). Binary image processing techniques can be useful for subsequent processing of these output images.

Binary images are typically obtained by thresholding a grey level image. Pixels with a grey level above the threshold are set to 1 (equivalently 255), whilst the rest are set to 0. This produces a white object on a black background (or vice versa, depending on the relative grey values of the object and the background). Of course, the 'negative' of a binary image is also a binary image, simply one in which the pixel values have been reversed.

However, choosing a threshold can be difficult, and is even considered by some [1] to be a 'black art'. Most approaches make use of the histogram of the number of times each grey level occurs in the image. If you are fortunate the histogram will be bimodal and choosing a threshold manually will be easy. It may even be possible to construct an automatic procedure to determine it. Ideally, if we had a black object on a white background the histogram should appear as in figure 1.

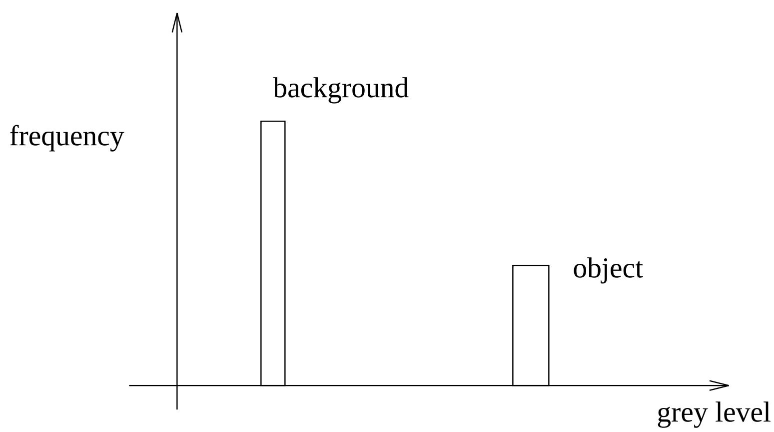

Figure 1: The ideal histogram of a light object on a darker plain background.

But we have measurement noise. The histogram we end up seeing is the result of convolving the 'ideal' histogram with the probability distribution of the noise (see figure 2).

If the grey levels of the object and the background are fairly close the influence of noise may result in the object only appearing as a 'shoulder' in the histogram (figure 3).

In this case the histogram will no longer be bimodal. There will be no clear way of choosing the threshold. Repeated observations and averaging may help, but the spread of the histogram may be due to lighting variations or colour variations in the background and object.

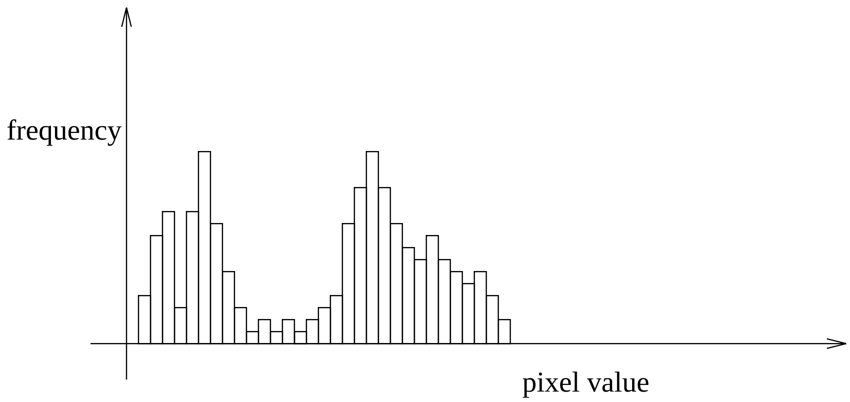

Figure 2: The histogram of an image showing the frequency of occurrence of each grey scale value.

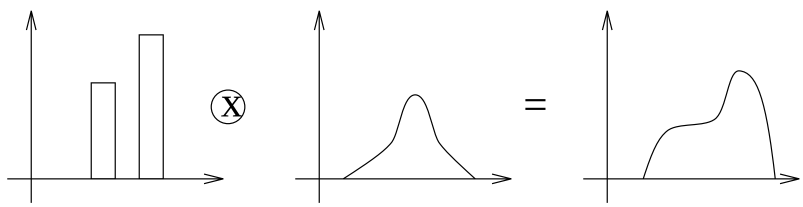

Figure 3: When background and object are close in grey levels, thresholding is difficult to determine automatically.

Analysis of Binary Images

We wish to determine various attributes of the objects in the scene with the aim of using these to identify the objects and to determine their position and orientation. We define the characteristic function of an object in an image to be (see figure 4)

$$b(x, y) \begin{cases} = 1 & \text{for points on the object} \\ = 0 & \text{for background points.} \end{cases}$$

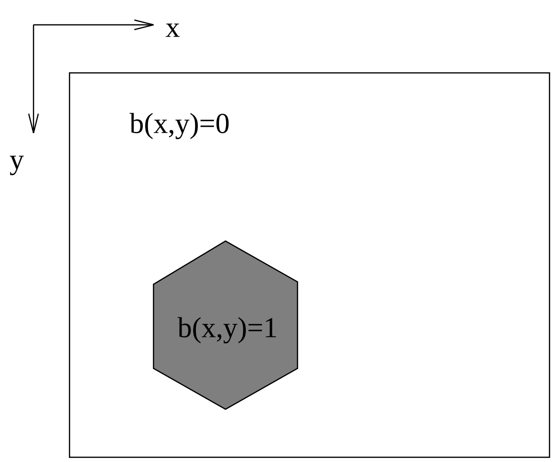

Figure 4: A binary image.

Now consider the image to be continuous (that is, to have infinite resolution).

The *area* is given by the 0^{th} moment of the object:

$$A = \int \int b(x, y) dx dy.$$

The *centre of mass*, denoted by $(\overline{x}, \overline{y})$, is given by the 1^{st} moments of the object:

$$\overline{x} = \frac{\int \int x b(x, y) dx dy}{\int \int b(x, y) dx dy},$$

and

$$\overline{y} = \frac{\int \int y b(x, y) dx dy}{\int \int b(x, y) dx dy}.$$

Generally we use the axis of minimum inertia of the object to identify the orientation. This is the axis of least 2^{nd} moment (see 5).

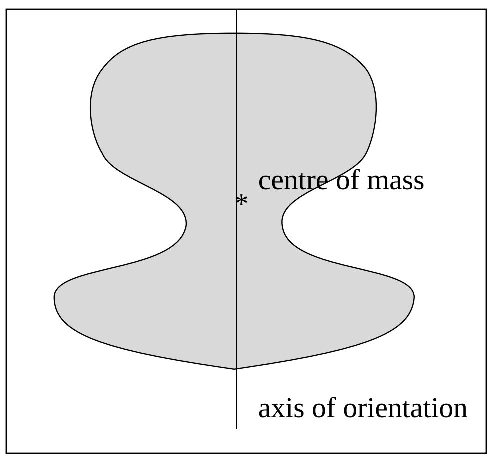

Figure 5: The 0^{th} moment is the area of the object; the 1^{st} moment gives the centre of mass; and the 2^{nd} moments give the axes of orientation.

There are a few ways to parametrize a line in an image plane (2D space). For instance, we could use

$$y = mx + b,$$

where m denotes the slope and m denotes the y-intercept of the line. The disadvantage of this parametrization is m becomes ∞ for all vertical lines. Another way to parametrize a line is using

$$\alpha x + \beta y + \gamma = 0,$$

where the parameters α, β, and γ that describe the line are defined only up to a scale (i.e., if we multiply them by the same arbitrary constant, they still describe the same line). Thus, one of the three parameters is in fact redundant. A better way to parametrize a line is to use two parameters ρ and θ as shown in figure 6. The parameter ρ specifies the distance of the line from the origin of the coordinate system and θ denotes the angle between the line and the x-axis. Note that, using this parametrization, vertical lines have $\theta = \pi/2$ and lines passing the origin have $\rho = 0$.

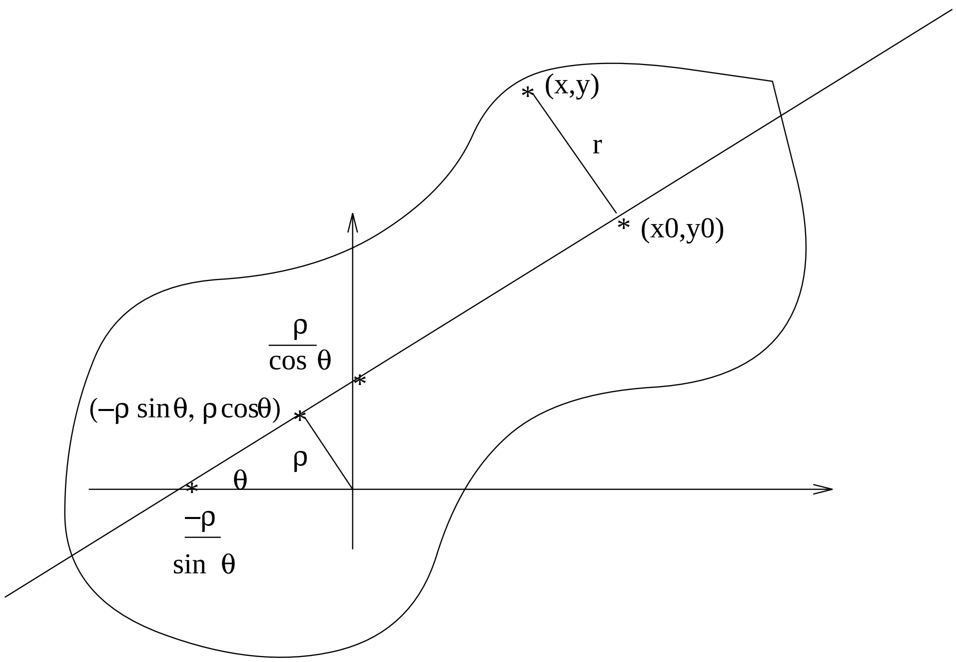

Figure 6: Parametrizing a line by its angle, θ, with the x-axis and its distance, ρ, from the origin.

We will see later that, using this parametrization, the equation of the line has the form

$$x \sin\theta - y \cos\theta + \rho = 0. \tag{1}$$

Our aim is to find the line for which the integral

$$I = \int\int r^2 b(x, y) dx dy$$

is a minimum, where r is the perpendicular distance from (x, y) to the line we want to find (see figure 6).

Thus we have to solve for the θ and ρ that will give the axis of least second moment.

Given this equation we can write parametric equations for points on the line as follows:

$$x_0 = -\rho \sin\theta + t \cos\theta,$$

and

$$y_0 = \rho \cos\theta + t \sin\theta,$$

where t is the distance along the line from the closest point to the origin.

Given a point (x, y) we need to find the closest point on the line so that we can calculate the distance r, which is given by (see figure 6)

$$r^2 = (x - x_0)^2 + (y - y_0)^2.$$

We substitute for x_0 and y_0 to get

$$r^2 = x^2 + y^2 + \rho^2 + 2\rho(x \sin\theta - y \cos\theta) - 2t(x \cos\theta + y \sin\theta) + t^2.$$

What value of t minimizes this expression?

Differentiating with respect to t and equating the result to 0 gives

$$t = x \cos\theta + y \sin\theta.$$

So we substitute this back into a parametric equations for x_0 and y_0. This gives

$$\begin{aligned} x - x_0 &= x - (-\rho \sin\theta + (x \cos\theta + y \sin\theta) \cos\theta) \\ &= x + \rho \sin\theta - x \cos^2\theta - y \sin\theta \cos\theta \\ &= x(1 - \cos^2\theta) + \rho \sin\theta - y \sin\theta \cos\theta \\ &= x \sin^2\theta + \rho \sin\theta - y \sin\theta \cos\theta \\ &= \sin\theta(x \sin\theta - y \cos\theta + \rho), \end{aligned}$$

and likewise for $y - y_0$.

Thus r^2 can be written as

$$r^2 = (x \sin\theta - y \cos\theta + \rho)^2.$$

This line is the locus of points for which $r = 0$. We see that by setting r to 0, we get the line equation (1) given earlier. Hence, by parametrizing the line in this manner we can obtain the distance from the line directly.

Solving for ρ and θ

We now want to find the ρ and θ that describe the line that minimizes distances of points in the object to that line, that is, that minimizes

$$I = \int\int (x \sin\theta - y \cos\theta + \rho)^2 b(x, y) dx dy. \tag{2}$$

We differentiate (2) with respect to ρ and set the resulting expression to 0, giving

$$\int\int 2(x \sin\theta - y \cos\theta + \rho) b(x, y) dx dy = 0,$$

which is just

$$\int\int x \sin\theta b(x, y) dx dy - \int\int y \cos\theta b(x, y) dx dy + \int\int \rho b(x, y) dx dy = 0.$$

Multiplying and dividing by $A = \int\int b(x,y)dxdy$ gives

$$A\left(\frac{\int\int x\sin\theta b(x,y)dxdy}{\int\int b(x,y)dxdy} - \frac{\int\int y\cos\theta b(x,y)dxdy}{\int\int b(x,y)dxdy} + \frac{\int\int \rho b(x,y)dxdy}{\int\int b(x,y)dxdy}\right) = 0,$$

and this is just

$$A(\overline{x}\sin\theta - \overline{y}\cos\theta + \rho) = 0,$$

where $\overline{x}$, $\overline{y}$ is the centre of mass. That is, the axis of minimum $2nd$ moment passes through the centre of mass.

Note that we can eliminate the term ρ by a simple change of coordinates so that the centre of mass of the object becomes the origin of the coordinate system. Let

$$\begin{aligned} x' &= x - \overline{x}, \\ y' &= y - \overline{y}. \end{aligned}$$

This simplifies the equation of our line to

$$x\sin\theta - y\cos\theta + \rho = x'\sin\theta - y'\cos\theta,$$

and the term I in (2) that we attempt to minimize becomes

$$I = \int\int (x'\sin\theta - y'\cos\theta)^2 b(x',y')dx'dy'$$

$$\Rightarrow I = a\sin^2\theta - b\sin\theta\cos\theta + c\cos^2\theta,$$

where

$$\begin{aligned} a &= \int\int x'^2 b(x',y')dx'dy', \\ b &= 2\int\int x'y'b(x',y')dx'dy', \\ c &= \int\int y'^2 b(x',y')dx'dy'. \end{aligned} \tag{3}$$

The constants a, b, and c are called the *second moments.*

If we use the substitutions $b\sin\theta\cos\theta = \frac{b}{2}\sin 2\theta$ and $\cos 2\theta = 2\cos^2\theta - 1$, then

$$\begin{aligned} I &= a(1-\cos^2\theta) + c\cos^2\theta - \frac{b}{2}\sin 2\theta \\ &= a + (c-a)\cos^2\theta - \frac{b}{2}\sin 2\theta \\ &= a + \frac{(c-a)}{2}\cos 2\theta + \frac{(c-a)}{2} - \frac{b}{2}\sin 2\theta. \end{aligned}$$

Thus

$$I = \frac{1}{2}(c+a) - \frac{1}{2}(a-c)\cos 2\theta - \frac{1}{2}b\sin 2\theta.$$

Recall that we are after ρ and θ that describe the line and that minimizes the distances of points in the object to that line. We have differentiated the expression I with respect to ρ earlier on. Now we differentiate this new expression of I with respect to θ and set the result to 0, obtaining

$$(a - c)\sin 2\theta - b\cos 2\theta = 0,$$

which implies

$$\tan 2\theta = \frac{b}{a - c},$$

unless $b = 0$ and $a = c$.

Thus

$$\frac{\sin^2 2\theta}{\cos^2 2\theta} = \frac{b^2}{(a - c)^2},$$

which implies that

$$\frac{\sin^2 2\theta}{1 - \sin^2 2\theta} = \frac{b^2}{(a - c)^2}.$$

This is just a quadratic in $\sin^2(2\theta)$, so we see that it has solution

$$\sin 2\theta = \frac{\pm b}{\sqrt{b^2 + (a - c)^2}},$$

and

$$\cos 2\theta = \frac{\pm(a - c)}{\sqrt{b^2 + (a - c)^2}}.$$

When we choose the positive solution for θ, we obtain the orientation of the major principal axis and the expression I is minimized, giving the value I_{min}; when we choose the negative solution for θ, we obtain the orientation of the minor principal axis and the expression I is maximized, giving the value I_{max} (see figure 7). So

$$\begin{aligned} \theta_1 &= \frac{1}{2}\operatorname{atan2}\left(\frac{b}{\sqrt{b^2 + (a - c)^2}}, \frac{a - c}{\sqrt{b^2 + (a - c)^2}}\right) \\ \theta_2 &= \frac{1}{2}\operatorname{atan2}\left(-\frac{b}{\sqrt{b^2 + (a - c)^2}}, -\frac{a - c}{\sqrt{b^2 + (a - c)^2}}\right). \end{aligned}$$

In the case that $b = 0$ and $a = c$ we see that I is unaffected by the direction of axis of orientation, that is, our object is rotationally symmetric.

The ratio $\frac{I_{min}}{I_{max}}$ gives us some idea of how rounded the object is. This ratio will be 0 for a line and 1 for a circle.

Computing the principal axes by analyzing the covariance matrix

We note that the expressions for a, b, and c in (3) are, respectively, the variance of the binary object for the x-direction, the correlation between the x and y-directions of the binary object, and the variance of the binary object for the y-direction. Putting a, b, and c together into matrix form, we obtain the *covariance matrix* C of the binary object as given below:

$$C = \begin{bmatrix} a & b/2 \\ b/2 & c \end{bmatrix} = \begin{bmatrix} \int\int x'^2 b(x', y')dx'dy' & \int\int x'y'b(x', y')dx'dy' \\ \int\int x'y'b(x', y')dx'dy' & \int\int y'^2 b(x', y')dx'dy' \end{bmatrix} \tag{4}$$

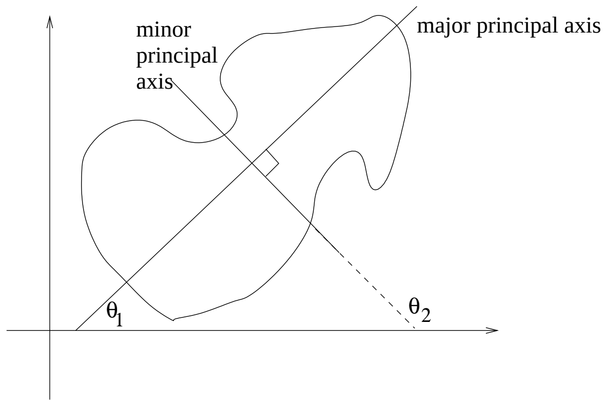

Figure 7: The major and minor principal axes of a sample binary object. The positive solution for θ gives θ_1 and the negative solution for θ gives θ_2.

The covariance matrix of any binary object can be easily computed using the Matlab function `cov`.

The computation of the principal axes of a binary object (or a cluster of points) is known as the *Principal Component Analysis* (or PCA for short). This operation can be easily done by extracting the eigenvalues and eigenvectors of the covariance matrix C — a procedure known as *eigen-decomposition* and the Matlab function `eig` will do the job. In the 2×2 covariance matrix given in (4) above, there are two eigenvalues, s_1 and s_2, and the corresponding eigenvectors $\mathbf{v}_1 = (u_1, v_1)^\top$ and $\mathbf{v}_2 = (u_2, v_2)^\top$.

Since $\mathbf{v}_1$ and $\mathbf{v}_2$ are orthonormal vectors (i.e., $\|\mathbf{v}_1\| = \|\mathbf{v}_2\| = 1$ and $\mathbf{v}_1 \cdot \mathbf{v}_2 = 0$), the 2×2 matrix

$$V = \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 \end{bmatrix} = \begin{bmatrix} u_1 & u_2 \\ v_1 & v_2 \end{bmatrix} \tag{5}$$

is a rotation matrix that gives the rotation required to bring the coordinate system to align with the two principal axes.

Suppose that $s_1 > s_2$. Then, under the new coordinate system (after rotation by the matrix V), s_1 and s_2 tell us, respectively, the variances of the binary object along the major and minor principal axes. In fact, s_1 is identical to $I_{\max}$ and s_2 is identical to $I_{\min}$ as computed before!!! The two vectors $\mathbf{v}_1$ and $\mathbf{v}_2$ give the orientations of the major and minor principal axes of the binary object.

Again, if $s_1 = s_2$ then it means that the object is rotationally symmetric. If $s_1 > s_2 = 0$ then it means that the binary object collapses to a straight line.

Since the covariance matrix is symmetric and positive semi-definite[1], its eigenvalues and eigenvectors are identical to its singular values and singular vectors. Rather than com-

[1] A matrix is said to be

- *positive definite* if all its eigenvalues are positive;
- *positive semi-definite* if all its eigenvalues are positive or zero;
- *negative definite* if all its eigenvalues are negative;
- *negative semi-definite* if all its eigenvalues are negative or zero;
- *indefinite* otherwise.

puting the eigenvalues and eigenvectors of C, we can compute the singular values and singular vectors instead, using the operation known as *singular value decomposition* (or SVD in short). The Matlab function `svd` can be used to obtain the singular values and singular vectors. The SVD decomposes the matrix C into the product of three matrices as given below:

$$C = USV^\top, \tag{6}$$

where U and V are both orthonormal matrices and S is a diagonal matrix containing the singular values, which are often sorted in decreasing order. Because C is a covariance matrix, we have $U = V$ and the V matrix computed from (6) is the same as the V matrix obtained from the eigen-decomposition in (5). The columns of V contain the singular vectors required. The angles θ_1 and θ_2 for the major and minor principal axes are then

$$\begin{aligned} \theta_1 &= \text{atan2}(v_1, u_1) \\ \theta_2 &= \text{atan2}(v_2, u_2). \end{aligned}$$

Discrete Binary Images

In a discrete binary image objects are represented in terms of discrete pixels. Calculations of areas, centroids, moments and so on are straightforward — we use summations instead of the integral equations we used in the continuous case.

Suppose there are several objects in the image. We must separate them before calculating the various moments. We need to establish what makes an object connected or separate from other objects.

Two points in a binary image are *connected* if a path can be found between them along which the characteristic function remains constant. To decide this we need to know what the topology is, that is, under what conditions do we say that two pixels are neighbours.

There are various possibilities for deciding upon the appropriate topology.

1. Four connectedness. In this case only edge adjacent pixels are neighbours.

2. Eight connectedness. In this case edge and corner adjacent cells are considered neighbours.

Neither choice is very good, since neither allow the Jordan Curve Theorem to be satisfied.

The Jordan Curve Theorem says *A simple closed curve separates the plane into two simply connected components* (namely, the inside and the outside).

Consider the following simple example:

If we use 4 connectedness we have 4 separate objects. These objects are not connected so all the background elements should be connected as one object — but the centre cell (a background one) is not connected to the others, giving us a contradiction. There are (at least) two background regions without a closed curve.

0	1	0
1	0	1
0	1	0

Figure 8: A discrete curve that contradicts the Jordan Curve Theorem.

If we use 8 connectedness we have an object that creates a closed curve — but the background element in the middle is also connected to the outside!

The ideal solution is to use a hexagonal grid, which gives us 6 connectedness. Unfortunately, images don't come in this form.

Another solution is to consider object pixels as 8 connected and background points as 4 connected (or vice versa). However, this asymmetry may be considered undesirable. Alternatively one can skew the rectangular grid.

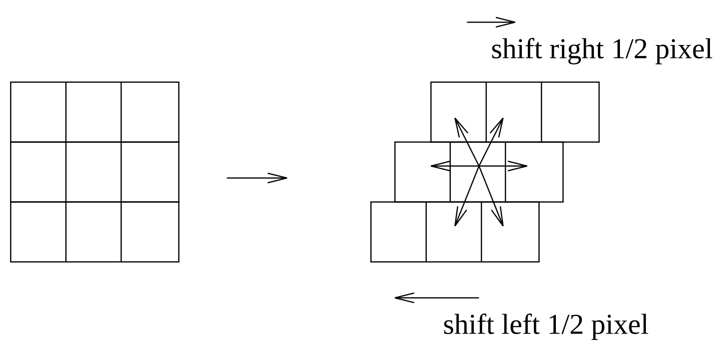

Figure 9: How to obtain 6-connectedness by shifting a rectangular grid.

This establishes 6 connectedness. In the undistorted grid this corresponds to an arrangement where the centre pixel is connected to its four immediate neighbours (as in 4-connectedness) plus two of its diagonal neighbours. Thus, one must chose between a left-skewed 6-connectedness, or a right-skewed 6-connectedness.

References

[1] Olivier Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint.* M.I.T. Press, 1993.

[2] Berthold Klaus Paul Horn. *Robot Vision.* MIT Press, 1986.