ELSEVIER

Pattern Recognition Letters 16 (1995) 987–996

Pattern Recognition Letters

Symmetry detection using gradient information

Changming Sun

CSIRO Division of Mathematics and Statistics, Institute of Information Science and Engineering, Locked Bag 17, North Ryde NSW 2113, Australia

Received 24 January 1995; revised 3 April 1995

Abstract

Symmetry detection is important in the area of computer vision. A simple and fast symmetry detection algorithm has been developed in this paper. The algorithm employs only the original image and the gradient information. The direction of the symmetry axis is obtained from the gradient orientation histogram; and the position of this symmetry axis is decided either by the center of gravity or by the profile of the image projection along the direction of the symmetry axis. This method works directly on the grey-scale image and does not require any prior segmentation of the input image. Both simulated and real images have been tested and the results are very convincing.

Keywords: Image symmetry detection; Orientation histogram; Image center of mass; Image projection

1. Introduction

Symmetry is a prolific phenomenon in the world. Many objects around us are strongly constrained. For instance, not only cultural artifacts but also many natural objects are bilateral symmetric (or with mirror symmetry). There are a lot of symmetries in human anatomy, and the symmetry of the ribs is one example. Ideas of symmetry arose among the ancient Greek philosophers and mathematicians in connection with their study of the *harmony* of the world. The ancient sculptors, artists, and architects created their masterpieces in accordance with the canons of harmony. The method of symmetry has become a powerful and effective instrument of theoretical research in modern science (Shubnikov and Koptsik, 1974). Many physical laws are formulated as some kind of *variational principle*: a physically possible state is the state which minimizes (or maximizes) some quantity. A state is said to be *homogeneous* if it has translation as its symmetry, and *isotropic* if it has rotation as its symmetry (Kanatani, 1990). The goal of an image understanding system is to identify and locate a specified object in the scene. In such cases, the system must have some knowledge of the shape of the desired object. Symmetries are good candidates for describing shape. It is a powerful concept that facilitates object detection and recognition in many situations. These representations can be used in robotics for recognition, inspection, grasping, and reasoning. Symmetry in an image allows it to be described economically. For example, if one half of an object is the mirror image of the other half, then one half need not be described. Symmetry may be defined in terms of three linear transformations in n-dimensional Euclidean space E^n: reflection, rotation and translation. Formally, a subset S of E^n is *symmetric* with respect to a linear transformation T

SSDI 0167-8655(95)00049-6

if $T(S) = S$. We shall only concentrate on reflectional symmetry in this paper. A reflectional symmetry has a reflection line, for which the left half space is a mirror image of the right half. The ribcage in a single CT slice, for instance, is symmetrical with respect to a vertical line that passes through the spine. The ribs in a chest X-ray image also appear to be symmetrical with respect to the mediastinum. Two example images are shown in Fig. 1.

Most of the work carried out on symmetry detection has been based on edge or contour or point sets information. Burton et al. (1984) considered a simple indexing scheme to implement the exponential pyramid data structure for particular symmetries. Wolter et al. (1985) described exact algorithms for detecting all rotational and involutional symmetries in point sets, polygons and polyhedra. Atallah (1985) and Davis (1977) have used evaluation techniques for symmetry detection on images composed of line segments, circles and points. Highnam (1986) presented an asymptotically optical algorithm to locate all the axes of mirror symmetry and optimal algorithms for finding rotational symmetries of a planar point set. Marola (1989a) presented an algorithm for finding the number and position of the symmetry axes of a symmetric or almost symmetric planar image. This method required the evaluation of some rational functions. He also presented a recognition procedure based on the measurements of the degree of symmetry of planar intensity images by superposition or by convolution (Marola, 1989b). Zabrodsky et al. (1992b) defined a Continuous Symmetry measure to quantify the symmetry of objects. They also presented a multiresolution scheme (Zabrodsky et al., 1992a) that hierarchically detects symmetric and almost symmetric patterns. Yuen (1990) and Yuen and Chan (1994) use the Hough transform technique for detecting skewed and rotational symmetry on a set of points. Jiang and Bunke (1992) presented an algorithm for determining rotational symmetries of only polyhedral objects. Parry-Barwick and Bowyer (1993) developed methods that can detect both hierarchical and partial symmetry of two-dimensional set-theoretic models with components constructed with a few straight edges or polynomials. This method has the disadvantage of being computational intensive. Zielke et al. (1992; 1993) only look at vertical or near-vertical symmetry axes in the image for car-following. Masuda et al. (1993) described a method of extracting rotational and reflectional symmetry by performing correlation with the rotated and reflected images. But the method has high computational cost and memory requirements. Bowns and Morgan (1993) presented a method of extracting facial features using natural orientation information. But this method is not suitable for general-shaped objects when the peak orientation is not related to the orientation of the symmetry axis.

In this article, we investigate the use of gradient information for symmetry detection in a grey-scale

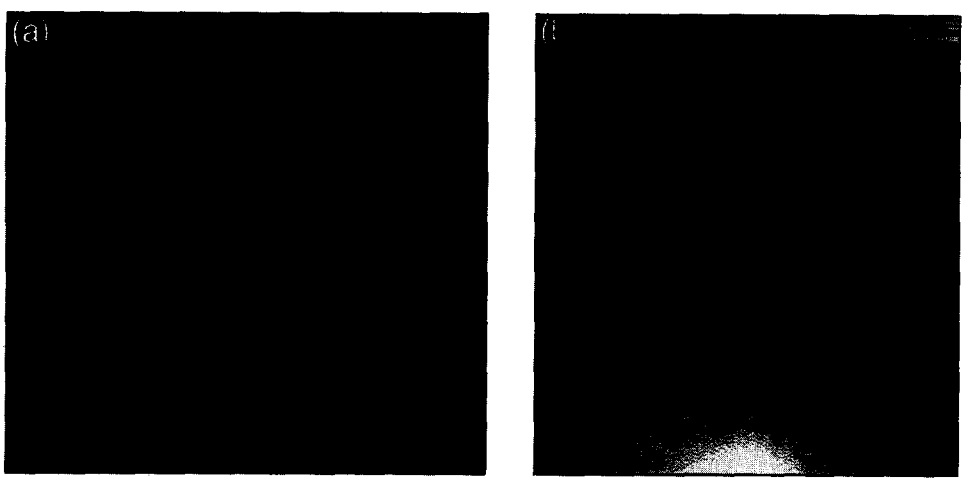

Fig. 1. Symmetrical ribs in (a) CT, and (b) X-ray images.

image by analyzing the shape of the orientation histogram. Section 2 briefly describes the gradient of an image. Section 3 and Section 4 give the algorithm for finding the orientation and position of the symmetry axis. Section 5 shows the results of the algorithm on both simulated and real images. Section 6 concludes.

2. Gradient of an image

The image formation includes imaging geometry, surface photometry, and surface contours. The image itself is often ambiguous in its representation of scene information, and the interpretation of such an image requires a combination of analytical tools and knowledge or assumptions about the scene (objects) and the image production (Bracho and Sanderson, 1985). The observed brightness of objects corresponds to the image irradiance. For our analysis, we will only consider Lambertian surfaces. Specular reflection or shadowing effect will not be treated here. For an object surface described by the equation,

$$z=f(x, y), \tag{1}$$

the surface gradient vector $[p, q]^{\mathrm{T}}$ is defined by

$$p=\frac{\partial f(x, y)}{\partial x}, \qquad q=\frac{\partial f(x, y)}{\partial y}. \tag{2}$$

The magnitude of this vector is

$$m=\sqrt{p^2+q^2}, \tag{3}$$

and the orientation of this gradient vector is

$$\phi=\arctan\frac{q}{p}. \tag{4}$$

The domain of ϕ is $[0, 2\pi)$. It is expected that the statistical information of the gradient orientation of an image can be used for symmetry detection. Details of the method will be described in Sections 3–5. This method can be misled by the presence of periodic texture within the object (if the texture pattern does not have the same symmetry axis as the object), as these repetitive patterns may give high peaks in the orientation histogram compared with those from the object boundary. But as long as the predominent gradient orientations are symmetric, the algorithm will be able to find the symmetry axis.

3. Orientation of the symmetry axis

Our algorithm is based on the gradient orientation distribution of the intensity image. From the previous section, the histogram of this gradient orientation image can therefore be obtained. The range of the orientation is from 0 to 360 degrees. The user is able to specify the number of bins for the histogram. In order to have better angular resolution, the bin number can be chosen to be some larger number than 360. In our case the number of bins is 1440 which corresponds to 0.25 degree resolution. It can be observed that for a symmetrical object in the image, its orientation histogram is symmetrical along a vertical line. It is also clear that this histogram function is periodic with period 360 (or 2π). That is,

$$h(\theta)=h(\theta\pm 2n\pi), \quad n=0, 1, 2, \ldots \tag{5}$$

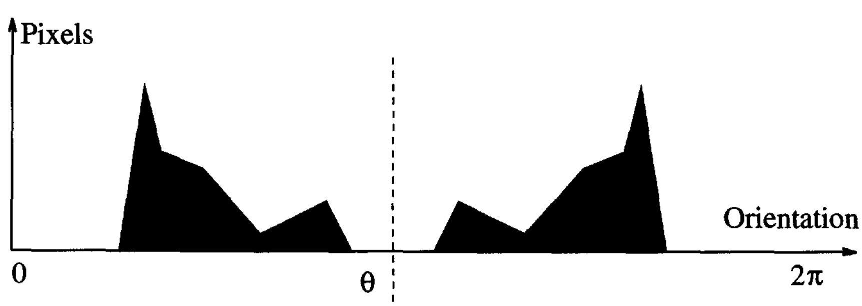

Fig. 2. Typical shape of the gradient orientation histogram for a reflectional symmetry object.

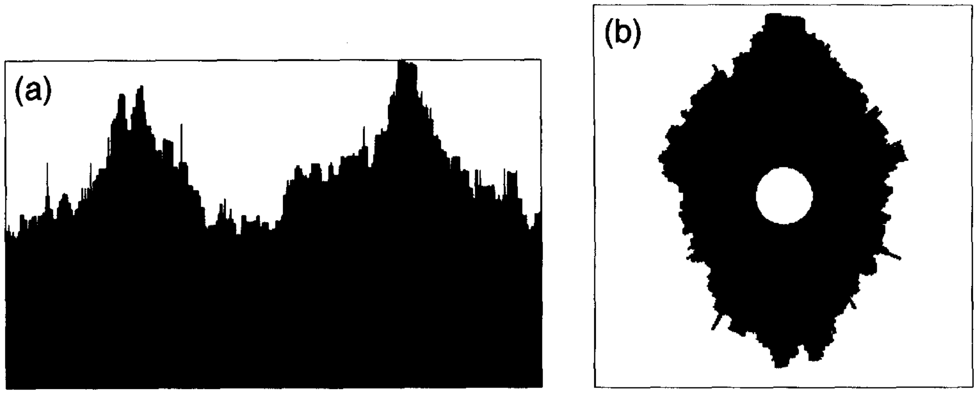

Fig. 3. Two ways to show a general histogram. (a) The usual histogram. (b) Circular histogram.

where $h(\theta)$ is the gradient orientation histogram of the image, and θ is in $[0, 2\pi)$. If we cut a window at position x of the histogram with length π from the left and π from the right, and calculate the following function:

$$c(x) = \sum_{\theta=0}^{\pi} h(x+\theta)h(x-\theta) \tag{6}$$

then the orientation of the symmetry axis is the value x which gives the maximum value for $c(x)$. Function $c(x)$ can be treated as a "correlation" function or a symmetry measure. It is extendable to the case of more than one symmetry axis. Currently only one symmetry axis is chosen simply by selecting the maximum correlation value. If there is more than one symmetry axis, the correlation function should have more than one peak. So by setting an appropriate threshold, multiple peak positions can be obtained which correspond to the multiple symmetry axes.

Fig. 2 shows the typical shape of the gradient orientation histogram for a reflectional symmetry object. The shape should be mirror symmetrical or nearly mirror symmetrical because of digitization error. Fig. 3 gives two ways of representing the

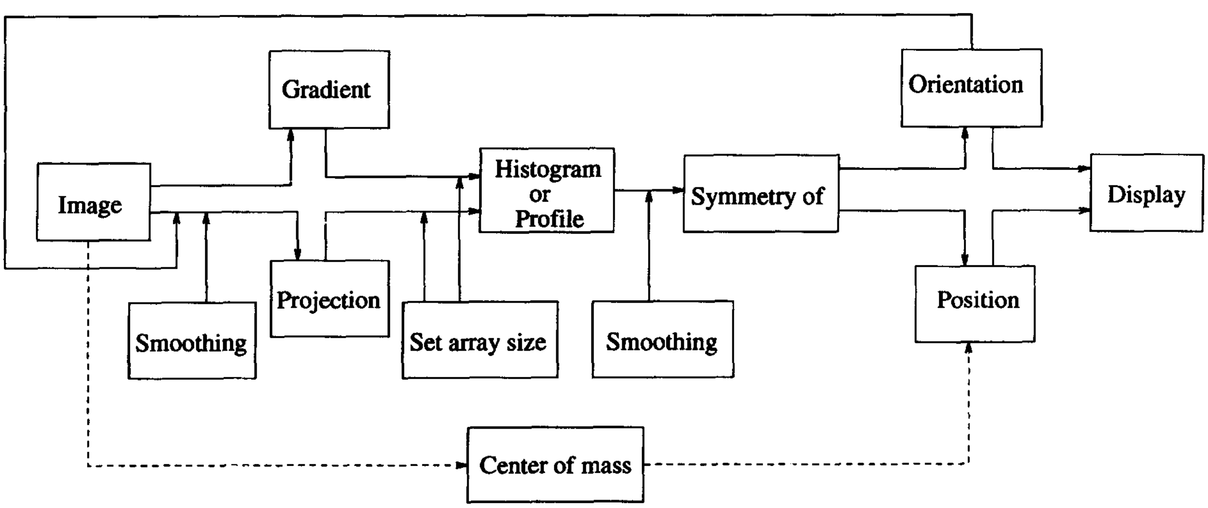

Fig. 4. Processing blocks of the algorithm.

histogram. Fig. 3(a) is the usual histogram with the horizontal axis showing the orientation (0–360°) and the vertical axis showing the number of pixels at this orientation, while Fig. 3(b) is a circular representation of (a) with the orientation angle starting from the center right (0°) and increasing clockwise, and the radial length showing the number of pixels at this orientation. This circular representation makes it easier for us to perceive the 2π periodicity.

4. Position of the symmetry axis

After the direction of the symmetry axis has been obtained, it is also necessary to determine the position of this line. This is performed by projecting the original image onto a line that is perpendicular to the symmetry axis. By analyzing the profile of this projection (in the way as in the previous section for finding the orientation of the symmetry axis), the position of the symmetry axis can be determined. Or even simpler, the center of gravity of the image (or the object of interest) can be used for determining the position of the symmetry axis.

The sequence of operations for obtaining the axis direction and position are:

(i) Performing gradient operation on the image;

(ii) Obtaining orientation histogram;

(iii) Searching for the correlation peak in this histogram using Eq. (6) to obtain the symmetry axis orientation;

(iv) Obtaining the position of this symmetry axis by either (a) using the center of mass of the object; or (b) projecting the image in the direction obtained from the previous step, and searching for the correlation peak on the projection profile to obtain the symmetry axis position;

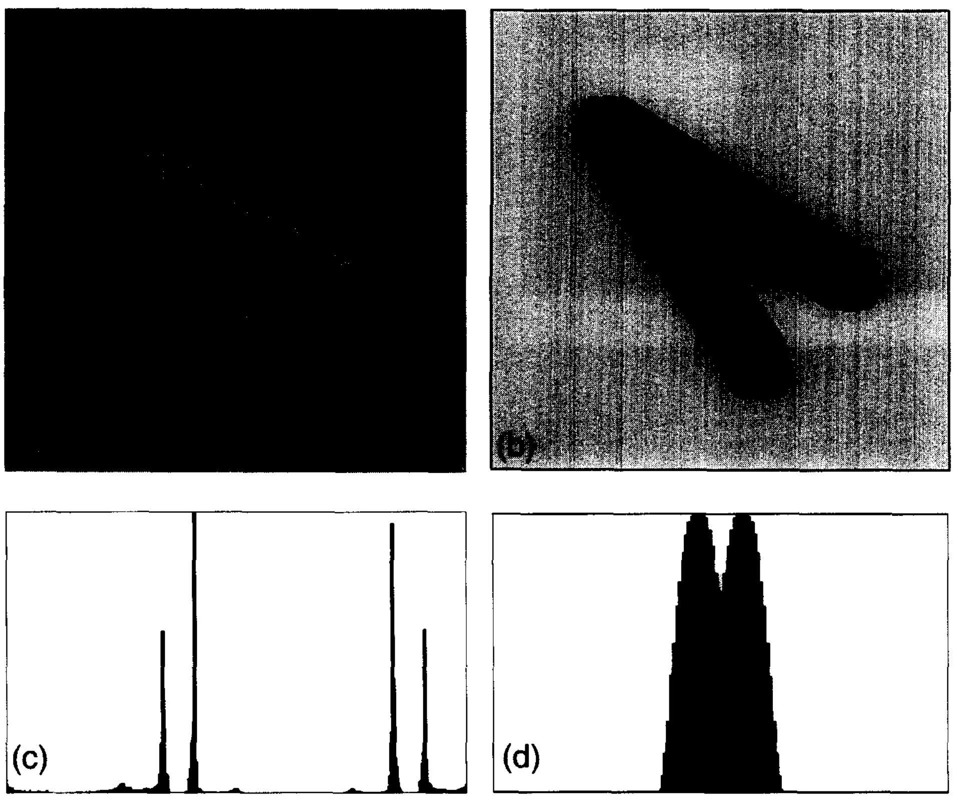

Fig. 5. Symmetry detection result. (a) Original image and the symmetry axis obtained. (b) Gradient orientation image (normalized from 0–360 to 0–255). (c) Histogram of the gradient orientation image (b). (d) Projection profile of the original image (a).

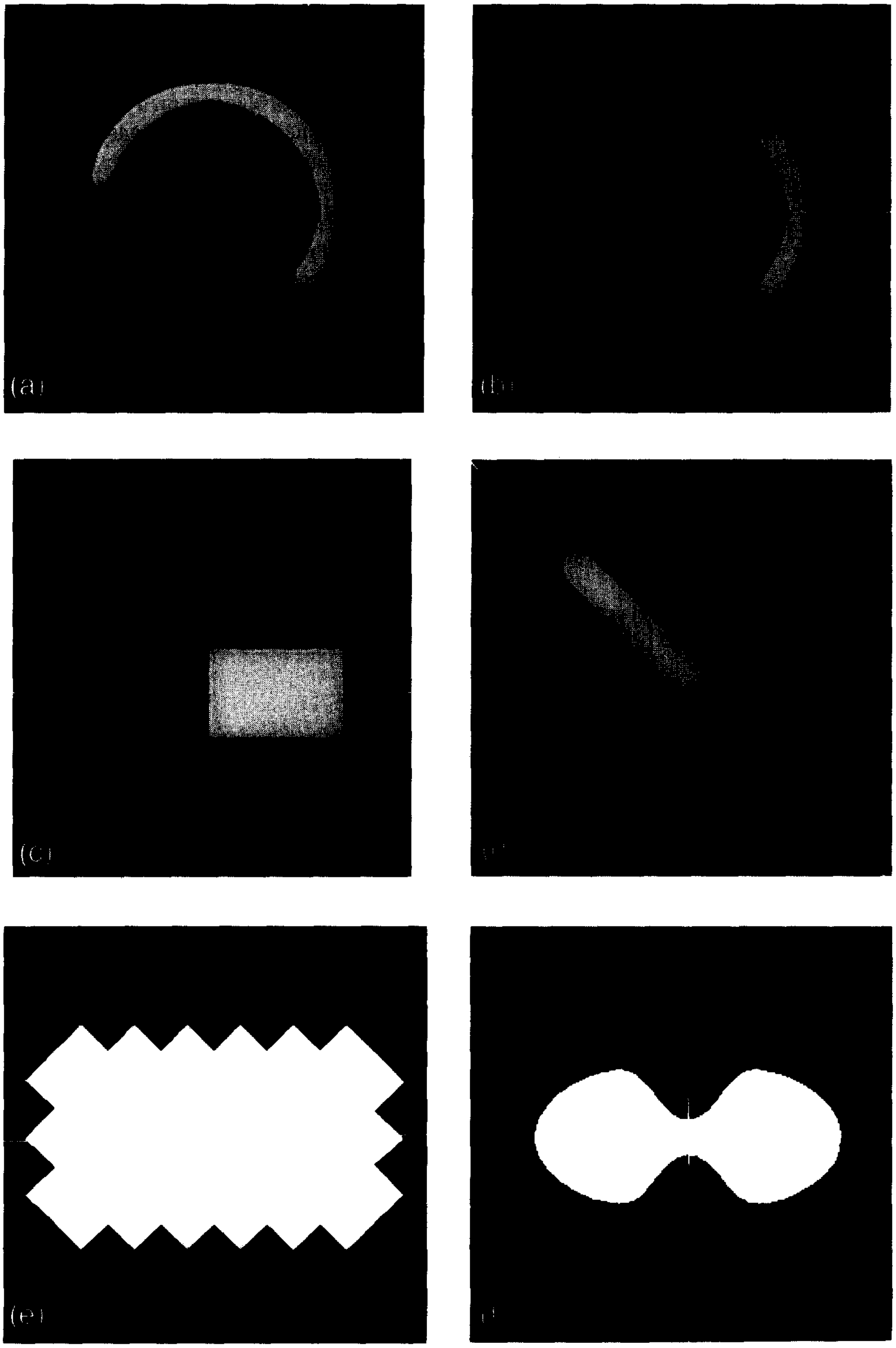

Fig. 6. Symmetry detection results for simulated images. (a) An arc. (b) A large portion of a ring. (c) A rectangular object. (d) A rod. (e) A rectangle with zigzagged boundary. (f) A fat propeller blade.

(v) Drawing the symmetry axis based on the orientation and position obtained.

The processing elements of the algorithm can be shown in the following diagram in Fig. 4. The dotted line shows one of the choices for obtaining the position of the symmetry axis using the center of mass.

5. Experimental results

The results for the above-described algorithm on simulated and real images are given in this section. The gradient image was obtained by using a Sobel operator. From this gradient image the orientation of the gradient vector for every image point was calculated and the orientation histogram was accumulated. No initial smoothing was applied to the original image before the gradient operation, as the process of obtaining the histogram has the effect of cancelling the noise contribution. The histogram is circularly smoothed, that is, the smoothing window is wrapped around at the ends since the angular data is circularly continuous. The correlation peak is found from the smoothed data.

Fig. 5(a) shows the original image and the symmetry axis obtained; Fig. 5(b) shows the gradient orientation image normalized from 0–360 to 0–255 for display purpose; Fig. 5(c) gives the histogram of the gradient orientation image (b); and Fig. 5(d) shows the projection profile of the original image in the direction of the symmetry axis. Fig. 6 shows the results of the symmetry detection algorithm on simulated images. Fig. 7 illustrates the steps for obtaining the symmetry axis for a real image. Fig. 8 gives the results of the algorithm on real images. The typical CPU running time for a 256×256 image is 0.15 secs on a DECstation5000. The CPU time of the algorithm using the center of mass or using the projection method to obtain the symmetry position are nearly the same; but the user times are different.

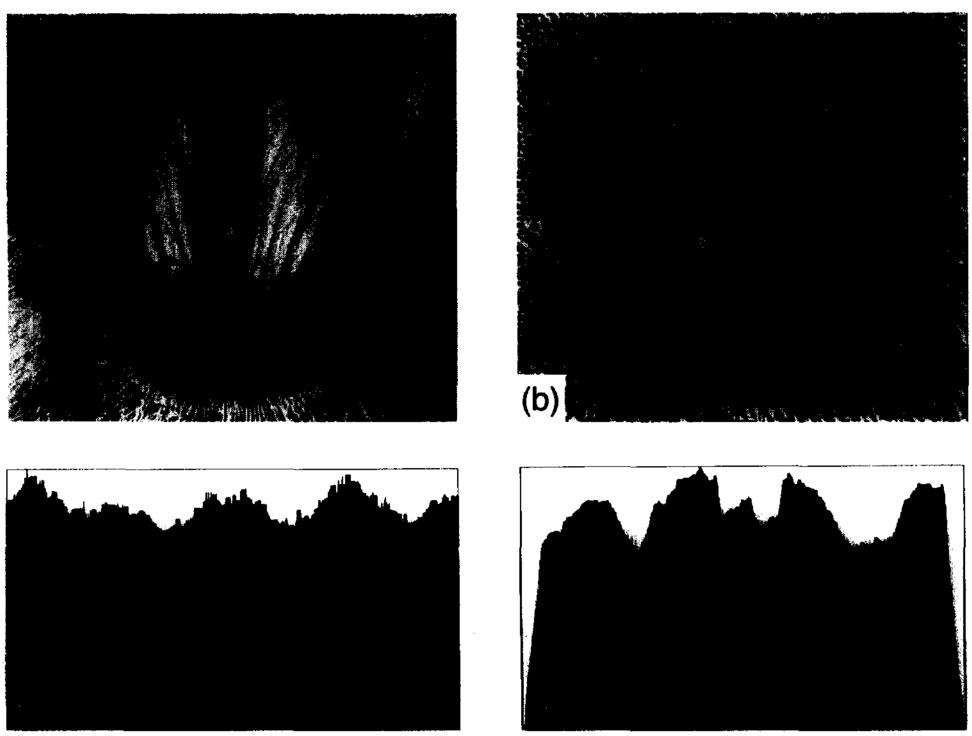

Fig. 7. Symmetry detection process for a real image. (a) Original image and the symmetry axis obtained. (b) Gradient orientation image (normalized from 0–360 to 0–255). (c) Histogram of the gradient orientation image (b). (d) Projection profile of the original image (a).

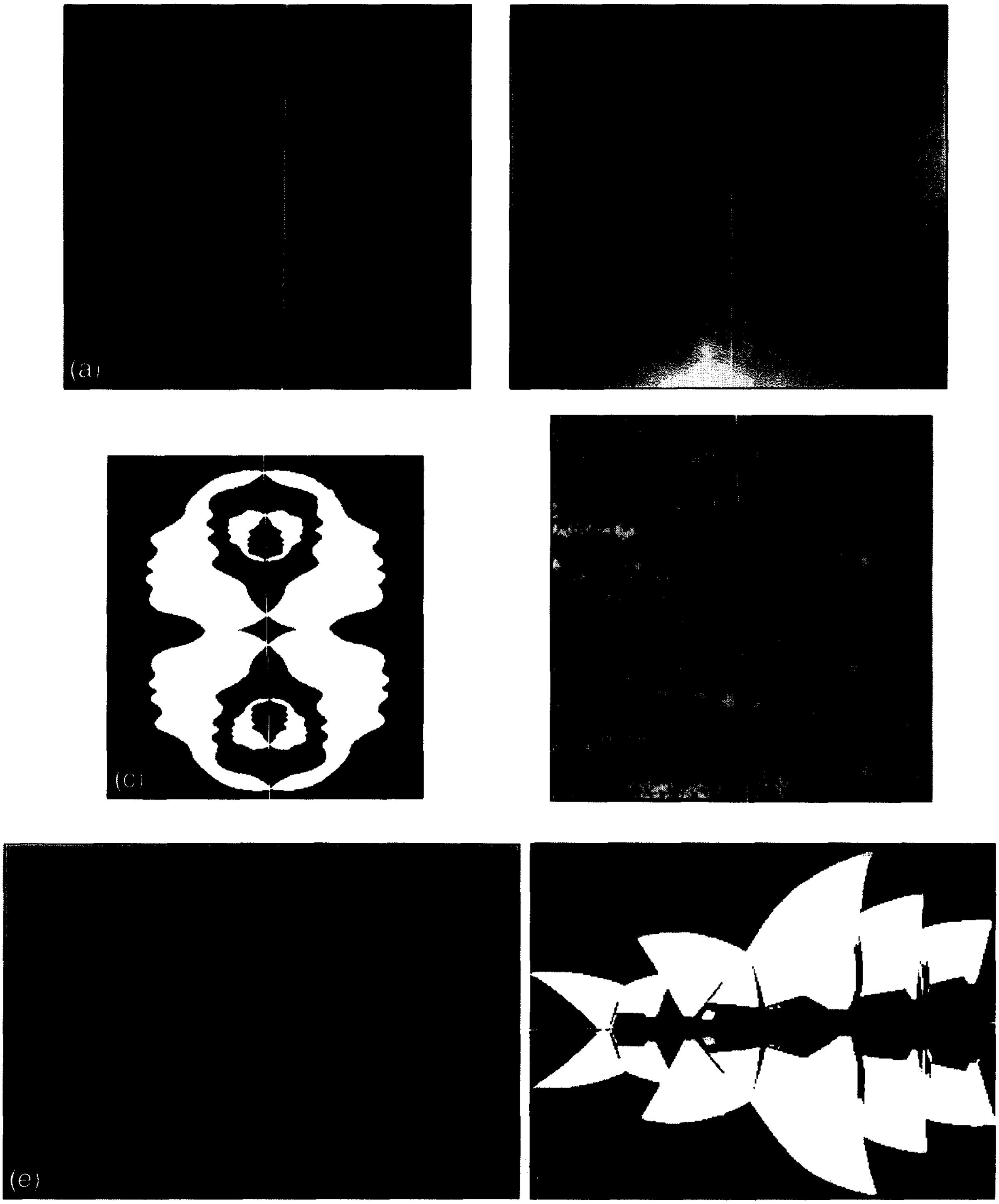

Fig. 8. Symmetry detection results for real images. (a) A slice of CT image. (b) A normal X-ray image. (c) Human head profiles. (d) A tile pattern. (e) A bridge. (f) Sydney Opera House. (Figs. (c), (e), and (f) are reproduced with kind permission from (Wade, 1990, p. 144, fig. 2.14, fig. 2.70.)

The user time of the algorithm using the projection method is about 3.8 secs, while the center of mass method takes about 1.8 secs. The code has not been optimized for speed of processing.

Because of the digitization effect, the boundary of an object consists mostly of zigzagged short line segments, and very often these line segments are either horizontal or vertical with diagonal segments connecting the horizontal or vertical short lines. Therefore, in most of the gradient orientation histograms, peaks often appear at 45° and multiples of 45° (such as 90°, 135°, 180°, 225°, 270°, and 315°). This effect can be reduced by increasing the kernel size in the gradient operation to some extent. But it is hard to eliminate the effect completely just by increasing the kernel size. A median filter has been used to remove these peaks.

6. Conclusions

A simple and fast symmetry detection algorithm has been developed which employs only the original grey-scale image and the gradient information. The results show that the statistics of the gradient orientation is enough to obtain the symmetry information of an object in an image. The CPU time for a 256×256 image takes only about 0.15 second on a DECstation 5000. The information about the symmetry axis can be of use for many applications (such as robot operation, further image segmentation, etc). In this article, only the reflectional symmetry has been analyzed (rotational symmetry detection will be reported elsewhere). Translational and skewed symmetry have not been dealt with. Further work in this line might be rewarding.

Work has only been performed on a single object in an image. The symmetry axes for separated multiple objects can be obtained after the image segmentation of these objects. For every single object, apply the algorithm described in this paper to obtain the symmetry axis. At the moment, only one symmetry axis has been obtained for an object. Other possible symmetry axes can also be identified by searching for more peaks of the correlation function. The number of peaks depends upon the threshold value.

The magnitude of the gradient has not been used. It is expected that the magnitude information can be helpful for the symmetry detection. From the nature of the algorithm, only "low-level" processing is involved, and therefore, it is possible to implement the algorithm in parallel.

Acknowledgement

This work was supported by the CSIRO Institute of Information Science and Engineering. The author is grateful to Leanne M. Bischof for her suggestions and for reading the manuscript. The author would like to thank Professor N. Wade for his kind permission to use three illustrations from his book: *Visual Allusions: Pictures of Perception*, 1990. The author is indebted to the anonymous referees for their valuable comments.

References

Atallah, M.J. (1985). On symmetry detection. *IEEE Trans. Comput.* 34 (7), 663–666.

Bowns, L. and M.J. Morgan, (1993). Facial features and axis of symmetry extracted using natural orientation information. *Biol. Cybernet.* 70, 137–144.

Bracho, R. and A.C. Sanderson (1985). Segmentation of images based on intensity gradient information. *Proc. Computer Vision and Pattern Recognition*, June 19–23, 1985, San Francisco, CA, USA, 341–347.

Burton, F.W., J.G. Kollias and N.A. Alexandridis (1984). An implementation of the exponential pyramid data structure with application to determination of symmetries in pictures. *Computer Vision, Graphics, and Image Processing* 25, 218–225.

Davis, L.S. (1977). Understanding shape: II. Symmetry. *IEEE Trans. Syst. Mach. Cybernet.*, March, 204–212.

Highnam, P.T. (1986). Optimal algorithms for finding the symmetries of a planar point set. *Inform. Process. Lett.* 22, 219–222.

Jiang, X.Y. and H. Bunke (1992). A simple and efficient algorithm for determining the symmetries of polyhedra. *Computer Vision, Graphics, and Image Processing: Graphical Models and Image Processing* 54 (1), 91–95.

Kanatani, K. (1990). *Group-Theoretical Methods in Image Understanding*. Springer, Berlin.

Marola, G. (1989a). On the detection of the axes of symmetry of symmetric and almost symmetric planar images. *IEEE Trans. Pattern Anal. Mach. Intell.* 11 (1), 104–108.

Marola, G. (1989b). Using symmetry for detecting and locating objects in a picture. *Computer Vision, Graphics, and Image Processing* 46, 179–195.

Masuda, T., K. Yamamoto and H. Yamada (1993). Detection of partial symmetry using correlation with rotated-reflected images. *Pattern Recognition* 26 (8), 1245–1253.

Parry-Barwick, S. and A. Bowyer (1993). Symmetry analysis and geometric modelling. In: K.K. Fung and A. Ginige, Eds, *Digital Image Computing: Techniques and Applications, Vol. I.* Australian Pattern Recognition Society, 39–46.

Shubnikov, A.V. and V.A. Koptsik (1974). *Symmetry in Science and Art*. Plenum Press, New York.

Wade, N. (1990). *Visual Allusions: Pictures of Perception.* Lawrence Erlbaum Associates, Hove and London.

Wolter, J.D., T.C. Woo and R.A. Volz (1985). Optimal algorithms for symmetry detection in two and three dimensions. *The Visual Computer* 1, 37–48.

Yuen, K.S.Y. and W.W. Chan, (1994). Two methods for detecting symmetries. *Pattern Recognition Lett.* 15, 279–286.

Yuen, S.Y.K. (1990). Shape from contour using symmetries. *Proc. European Conf. on Computer Vision*, Lecture Notes in Computer Science 427. Springer, Berlin, 437–453.

Zabrodsky, H., S. Peleg and D. Avnir (1992a). Hierarchical symmetry. *Proc. Internat. Conf. on Pattern Recognition, Vol. III*, Aug. 30–Sept. 3, 1992, The Hague, Netherlands, 9–11.

Zabrodsky, H., S. Peleg and D. Avnir (1992b). A measure of symmetry based on shape similarity. *Proc. Computer Vision and Pattern Recognition*, 703–706.

Zielke, T., M. Brauckmann and W. von Seelen (1992). CARTRACK: Computer vision-based car-following. *Workshop on Applications of Computer Vision*, 156–163.

Zielke, T., M. Brauckmann and W. von Seelen (1993). Intensity and edge-based symmetry detection with an application to car-following. *Computer Vision, Graphics, and Image Processing: Image Understanding* 58 (2), 177–190.