Straight lines and circles in the log-polar image

David Young

School of Cognitive and Computing Sciences
University of Sussex
Brighton, BN1 9QH, UK
`email: davidy@cogs.susx.ac.uk`

Abstract

Foveal or spatially-variant image representations are important
components of active vision systems. Log-polar sampling is a
particularly powerful example as a result of the simplicity with
which expansion and rotation can be handled. These properties
are exploited here for the detection of general straight lines, line
segments, and circles through the foveation point. An efficient
and practical method based on convolution is described, and in-
vestigated in the context of a simple foveation strategy.

1   Introduction

The potential value of non-uniformly sampled or spatially variant images is
greatly increased when vision is active. Foveal sampling, where sample points
are densest in the centre, allows computational resources to be concentrated
on regions of particular interest, whilst maintaining a wide field of view, but
it requires eye or camera movements to allow such regions to be selected. Al-
though in animal eyes non-uniform sampling is the rule rather than the excep-
tion, this form of image representation has been exploited relatively little in
computer vision. This is partly because of the prevailing camera technology,
which is geared to image transmission and processing and so employs uni-

1

form sampling, and partly because without effective active cameras non-uniform sampling sacrifices too much potential information. Now that active vision systems are becoming more common, non-uniform sampling is likely to increase in importance.

Despite its difficulties, non-uniform sampling, and in particular log-polar sampling, has received a certain amount of attention. Funt [3] demonstrated some of the fundamental advantages of an active foveated system for representing solid motion in 2-D, whilst Weiman and Chaikin [11] laid some mathematical groundwork. Wilson [12] emphasised the approximate log-polar mapping of the optic array onto the visual cortex in primates. A number of researchers, notably Tistarelli and Sandini [7, 8] have used the scheme in the context of motion detection; Tunley and Young [9] investigated the advantages of log-polar representations in estimating first-order optic flow. Also using log-polar sampling, Lim, West and Venkatesh [4] have developed mechanisms for precise foveation of features, Peters and Bishay [5] have described foveation on vanishing points, and Bederson, Wallace and Schwartz [1] have described an active vision system incorporating log-polar sampling.

The present paper builds on the theoretical work of Weiman and Chaikin [11] to explore the representation and detection of straight lines and circles in log-polar sampled images. An efficient new algorithm for finding these structures is described and its performance on real images investigated. The algorithm is intended to be applied in the context of a system like that of Brunnström, Eklundh and Uhlin [2], where a representation of a scene is built up using directed foveations.

2   The log-polar sampled image

In log-polar sampling, pixels are indexed by ring number R and wedge number W, related to ordinary x, y image coordinates by the mapping

$$r = [(x - x_c)^2 + (y - y_c)^2]^{1/2}, \qquad \theta = \tan^{-1}\left(\frac{y - y_c}{x - x_c}\right)$$

$$R = \frac{(n_r - 1)\log(r/r_{min})}{\log(r_{max}/r_{min})}, \qquad W = \frac{n_w \theta}{2\pi} \tag{1}$$

where (r, θ) are polar coordinates, (x_c, y_c) is the position of the centre of the log-polar sampling pattern, n_r and n_w are the numbers of rings and wedges respectively, and r_{min} and r_{max} are the radii of the smallest and largest rings of samples. We also define $\rho = \log r$.

A log-polar sampled image is one whose samples are centred on points mapping to integral R and W, $R \in \{0, \ldots, n_r - 1\}$, $W \in \{0, \ldots, n_w - 1\}$. The separation between sample points is proportional to distance from the sam-

pling centre, as shown in Fig. 1a. This arrangement appears to be approximated by the ganglion cells of the primate retina and the visual cortex [6]. In this representation, image expansions and rotations about (x_c, y_c) become shifts in R and W, but image translation has a more complex effect.

In order to keep a pixel's nearest neighbours in orthogonal directions at approximately equal distances from it, the following constraint is needed

$$r_{min} = r_{max}e^{-2\pi(n_r - 1)/n_w} \tag{2}$$

Log-polar sampled images are often displayed on orthogonal (R, W) axes, as in Fig. 1b, but this is misleading since it leads them to be regarded as "distorted" representations. In fact, the distortion only arises when they are displayed on the page or screen: as a mapping from coordinate values to position on a plane, the log-polar representation is no more distorted than the conventional one. When displayed with the correct mapping to position, as in Fig. 1c, the significant observable feature is the loss of resolution towards the periphery, as the samples become further apart.

These images should ideally be generated using special-purpose cameras, such as those described in [8]. However, a reasonable approximation for research is obtained by resampling a conventionally digitised image, and this method is used in the present work.

3   The straight line in the log-polar image

3.1   The log-polar straight line and its Fourier transform

Any straight line, not passing through (x_c, y_c), can be mapped into any other straight line by a rotation (to make the lines parallel) followed by a uniform expansion with (x_c, y_c) fixed. This property can be exploited to allow easy detection of lines in log-polar images. The idea was introduced by Weiman and Chaikin [11], and an implementation briefly discussed by Young [13].

Essentially, the log-polar image of a straight line is taken as a template and convolved with the log-polar image under analysis. Peaks in the output correspond to the rotations and expansions that map the template onto matching structures in the image, and so directly give the parameters of detected lines.

The equation of the straight line $x = 1$ in log-polar coordinates is $\rho = -\log\cos\theta$ and a graph of this equation is shown in Fig. 1d. If we convolve the reflected log-polar image of this special straight line with the image of a general line given by $\rho = \rho_l - \log\cos(\theta - \theta_l)$ then the peak of the convolution output will be at (ρ_l, θ_l).

Since the template is the same size as the image, it is efficient to perform the convolution by multiplication in the Fourier domain. It is possible to find a closed-form expression for the Fourier Transform of the straight line in log-
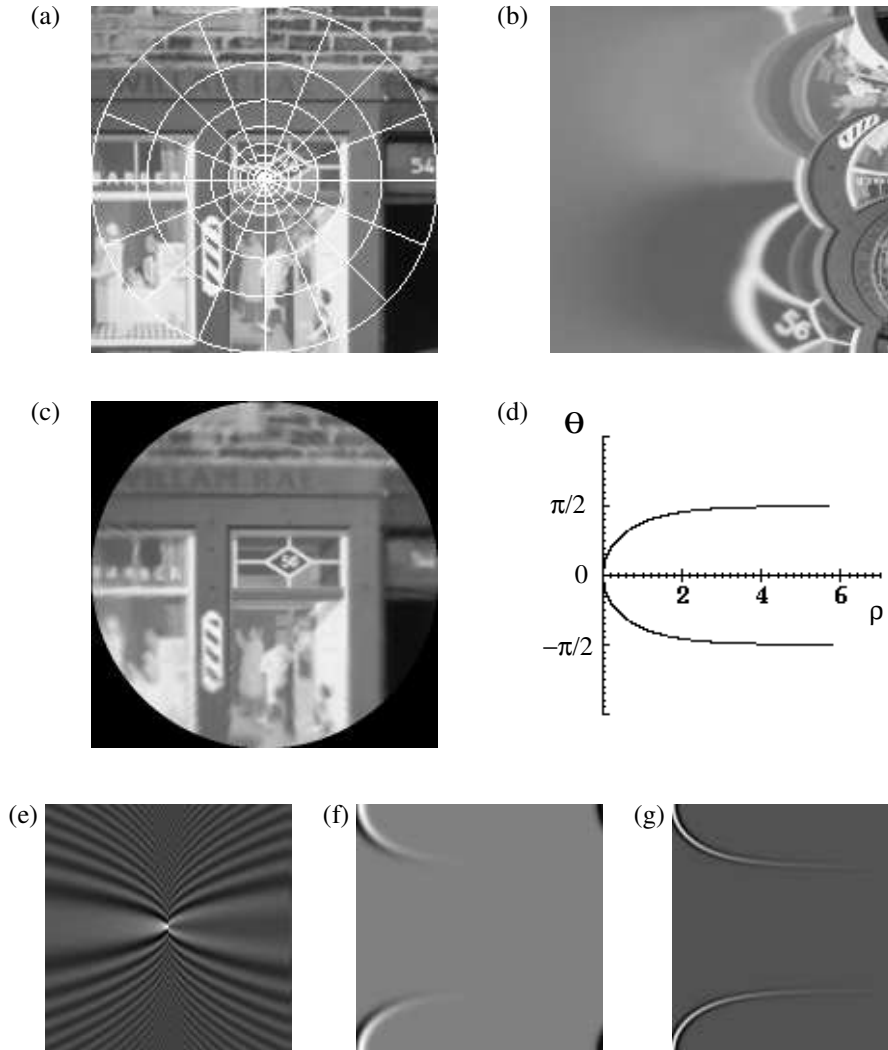
Figure 1: (a) A log-polar grid with 16 rings and 16 wedges superimposed on a 180×180 pixel image. Each sample in a 16×16 log-polar image would be derived from the grey levels in one segment of this grid. **(b)** The same image sampled on a log-polar grid with 180 rings and 180 wedges, displayed on orthogonal axes, R horizontal and W vertical, the origin at the bottom left and top left corners (since W wraps round). Moving up a column of (b) corresponds to moving anticlockwise round a ring in (a), starting at 3 o'clock. **(c)** The log-polar image of (b) displayed with veridical mapping onto the plane. (Bilinear interpolation was used to display the image.) **(d)** The graph of $\rho = \log r = -\log \cos \theta$ (i.e. $(x = 1)$). **(e)** The sum of the real and imaginary parts of the discrete Fourier transform of a straight line mask; origin at the centre, k_ρ horizontal and k_θ vertical. **(f)** A straight line mask with 128 rings and 128 wedges, coordinate system as in (b), generated in the Fourier domain and transformed numerically. The mask was differentiated with respect to R, and smoothed with a circular Gaussian mask with (spatial) $\sigma = 2$, both operations carried out in the frequency domain. **(g)** As (e), but the mask was convolved with a difference of Gaussians with inner $\sigma = 1$ and outer $\sigma = 1.2$ in the frequency domain.

4

polar space. Although in practice it might be adequate to synthesise the straight line in a log-polar array and apply the discrete Fourier transform, computing its transform directly avoids noise caused by starting from a discrete representation of the line. The formula for the transform also opens up the possibility of further analysis of the properties of the process in the frequency domain, though this is not exploited here.

To find the transform, we take a path integral along the line in log-polar space; if S is the standard line $\rho = -\log\cos\theta$ with element ds in (ρ, θ) space, the integral is

$$F(k_\rho, k_\theta) = \int_S e^{-ik_\rho\rho} e^{-ik_\theta\theta} w(\rho, \theta) \mathrm{d}s \qquad (3)$$

where $w(\rho,\theta)$ is a weighting factor to allow convergence. This must be smooth and tend to zero for large ρ. A suitable choice is

$$w(\rho, \theta) = (\cos\theta)^{1-\alpha}, \qquad 0 < \alpha < 1 \qquad (4)$$

where a larger α makes the template more localised round the minimum of ρ. In all the examples in this paper, $\alpha = 0.2$. Since $\mathrm{d}\theta/\mathrm{d}s = \cos\theta$ the integral becomes

$$F(k_\rho, k_\theta) = \int_{-\pi/2}^{\pi/2} (\cos\theta)^{ik_\rho - \alpha} e^{-ik_\theta\theta} \mathrm{d}\theta \qquad (5)$$

Rearranging and using standard tables, this evaluates to

$$F(k_\rho, k_\theta) = \frac{2^{\alpha - ik_\rho}\Gamma(1 - \alpha + ik_\rho)}{\Gamma(1 - (\alpha - k_\theta - ik_\rho)/2)\Gamma(1 - (\alpha + k_\theta - ik_\rho)/2)} \qquad (6)$$

where Γ is the complex gamma function. The line $x = 1$ is at an arbitrary position in the log-polar grid; to make a useful mask we choose as the template $x = r_{\min}$. The discrete Fourier transform of this is obtained by evaluating $F(k_\rho, k_\theta)$ at unit intervals of k_R and k_W from 0 to $n_r - 1$ and $n_w - 1$ respectively, with $k_\rho = 2\pi(n_r - 1)k_R/(n_r\log(r_{\max}/r_{\min}))$ and $k_\theta = k_W$.

3.2 Implementation of straight line detection

Transformed straight line templates were generated using Eq. 6. Because of symmetry, the sum of the real and imaginary parts is sufficient; an example of a computed template in the frequency domain is shown in Fig. 1e. Templates were multiplied by the frequency domain representations of a variety

of other operators, allowing Gaussian smoothing, differentiation with respect to *R* or *W*, and difference of Gaussians convolution to be combined with line detection in a single step. This allows matching to various kinds of boundaries. Examples of the masks generated are shown in Figs 1f and 1g.

Any linear operation can be thus combined with line detection, but the images used in this study were first subjected to a non-linear process to reduce sensitivity to the grey-level range. This involved subtracting a local average of the grey level, obtained by Gaussian smoothing, from each pixel in the original conventional images, and then applying the logistic function $1/(1 + \exp(-Kx))$ to each pixel of the result.

This compression was carried out on the conventional image input prior to resampling in order to speed up the tests, but it could easily and effectively be done on the log-polar images. Figs 2a and 2b show the effect of this preprocessing. For the results described here, σ for the smoothing was 10, the initial maximum grey level was 255 and *K* was 10/255.

Preprocessed images were resampled to a log-polar grid on the basis of Eq. 1. Bilinear interpolation was used in the inner region where the log-polar pixels are closer together than the original pixels, and simple averaging over a disc was used in the outer region where the log-polar pixels are bigger than the originals. Although this approach is quite crude, it is adequate for this investigation, and it is fast.

The log-polar image was transformed using the Fast Fourier Transform (FFT). The results were multiplied point-by-point with the template transform and the (complex) product transformed back to the spatial domain, to give the (real) convolution output, $C(R,W)$. Defining a peak as a pixel whose value exceeds some threshold and is not less than any of its eight nearest neighbours, peaks in *C* and in -*C* were found and ranked by absolute value. Peak positions were refined by taking the centre of gravity of a 3×3 region in the convolution output centred on the peak pixel. The strongest peaks were taken as representing the most salient straight lines in the sampled region. An example is shown in Fig. 2c.

3.3 Line segment detection

Line detection only provides the parameters of infinite lines, not the end points of line segments. To find end points it is necessary to reinspect the original image, tracing the line to see which parts of it contributed significantly to the convolution output. This must take account of the operations such as smoothing and differentiation that were incorporated into the convolution.

A straightforward way to do this is to multiply the image pixel by pixel with the spatial form of the convolution mask, having first reflected and translated the latter to make it line up with the detected line. The resulting array contains the values that would have been summed to produce the convolution
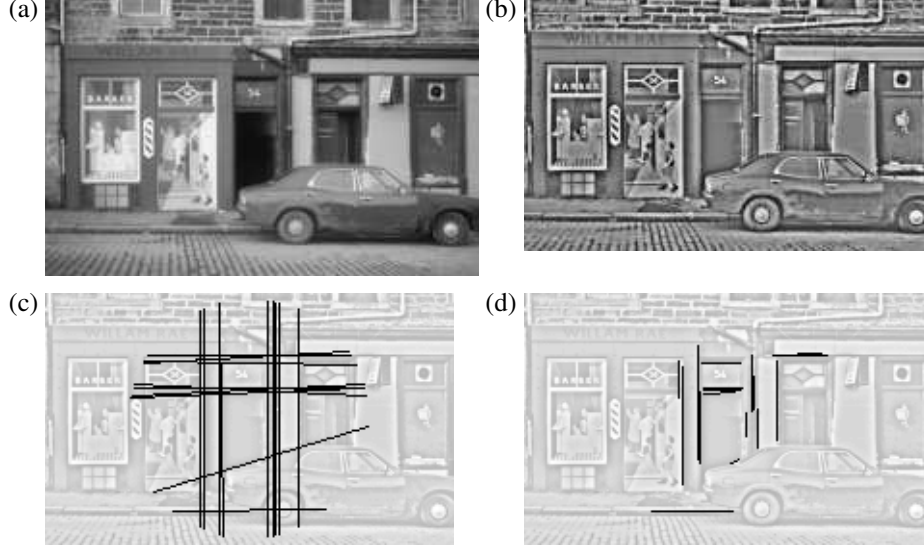
Figure 2: (a) The original 903×577 pixel image. **(b)** The image after non-linear preprocessing. **(c)** The straight lines found using log-polar sampling with r_{max}=250, n_r=128, n_w=256, r_{min}≈11, the mask smoothed with σ=1 and differentiated with respect to R. The threshold for peak detection was 5 times the standard deviation of the convolution output. **(d)** The lines truncated to regions of high evidence. The 1-D evidence array was smoothed with σ=2 and truncation occurred where the evidence fell to 0.5 times its peak value.

peak, if the convolution had been carried out in the spatial domain. The size of these values indicates how much each pixel of the image contributed to the peak.

We then simply project this array onto the W axis, by summing over all R from 0 to $n_r - 1$ for each W. Starting from the maximum in the resulting one-dimensional array, we search outwards in each direction, wrapping round if necessary, to find a value less than some constant times the peak value. This gives the limits in θ of the line segment which contributed most to the detected line. For a line with parameters ρ_l and θ_l, these limits can be converted to conventional coordinates using

$$x = e^{\rho_l} \cos\theta / \cos(\theta - \theta_l) + x_c, \qquad y = e^{\rho_l} \sin\theta / \cos(\theta - \theta_l) + y_c \quad (7)$$

This simple and fast procedure assumes that the line does not pass exactly through the sampling centre (x_c, y_c). In fact, this case is rare because the high density of samples close to (x_c, y_c) means that a small offset between the line and the centre is represented by a significant distance in R.

Truncating the lines shown in Fig. 2c results in those shown in Fig. 2d.

4 The circle in the log-polar image

4.1 The log-polar circle and its Fourier transform

It is reasonable to ask whether the simplicity of straight line detection in log-polar images can be extended to other curves. Since a circle through the sampling centre can be mapped onto any other such circle by a rotation and an expansion, such circles can also be detected by a convolution in log-polar space. Circles passing through the origin are, of course, unlikely to occur by chance. However, if the sampling centre is deliberately placed on a smooth boundary, possibly using the output of log-polar straight line detection, then the circle detected should approximate the osculating circle and the orientation and curvature of the boundary can be estimated.

In fact, the relationship to the straight line method goes deeper than this, and circles through (x_c, y_c) can be found for almost no additional computational effort alongside straight lines. As Weiman and Chaikin [11] pointed out, circles through (x_c, y_c) are mirrored straight lines in log-polar coordinates: the equation of the circle centred on $x = x_c + 1/2$, $y = y_c$ is $\rho = \log \cos \theta$. Thus the templates shown in Figs 1f and 1g will match circles if they are simply left-right reflected. A peak in the convolution output at (ρ_c, θ_c) gives the point on the circle diametrically opposite to the sampling centre, and hence the circle's centre and radius. Figs 3a and 3b demonstrate this.

It follows from the symmetry that the Fourier transform of the circle mask is just the complex conjugate of that of the straight line mask. It is possible to obtain the convolution results for straight lines and circles with a single transform. If L_R and L_I are the real and imaginary parts of the discrete transform of the log-polar image $L(k_R, k_W)$, and F_R and F_I are the real and imaginary parts of the straight line template transform $F(k_R, k_W)$, then

$$M(L_R - L_I) + iM(L_R + L_I) = FL + iF{*}L \tag{8}$$

where $M = F_R + F_I$ (and is real). We therefore form the expression on the left in the Fourier domain, and on transforming this back to the spatial domain, the real part of the result is the convolution with the line mask F and the imaginary part is the convolution with the circle mask F^*. This combined technique is only useful if the same preprocessing is suitable for both line and circle detection.
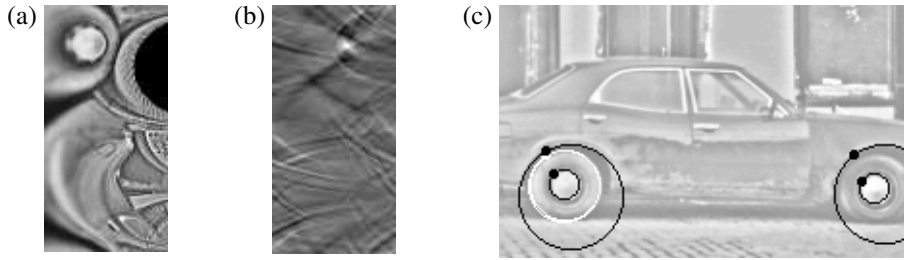
Figure 3: (a) A log-polar image displayed as in Fig. 1b with (x_c, y_c) on the upper edge of the car's rear wheel. The sampling parameters are as for Fig 2. The car's shadow and the wheel show the reflection symmetry between lines and circles. **(b)** The result of convolving the image of (a) with the circle mask. The mask was smoothed with σ=1, differentiated with respect to W and multiplied by $\sin\theta$. **(c)** The circle corresponding to the maximum of (b) (in white), along with four other circles (black) generated the same way at different (x_c, y_c) (black dots), each chosen to be close to a curved boundary. Wheel arches are found even though they are not exact or complete circles.

4.2 Implementation of circle detection

The implementation of circle detection is almost identical to the line detection method described above. One addition to the repertory of preprocessing options turns out to be desirable: modulation of the mask by $\sin\theta$ (which is also carried out in the frequency domain). This is helpful because, for a curve passing through the centre, differentiation with respect to R is less useful than differentiation with respect to W, but the sign of the gradient with respect to W will change on passing through the centre. Modulation of the template by angle overcomes this difficulty.

Unfortunately, circle detection in resampled log-polar images is impeded by the poor representation in the inner rings, where the log-polar pixels are smaller than the pixels of the original image. This means that boundaries close to the centre are affected by the pixel structure of the original image, and hence are hard to match well. Whilst not generally a problem for straight line detection, this does affect circle detection, where the template is aimed at picking up evidence close to the centre in order to find boundary orientation and curvature.

This difficulty can be overcome to some extent by reducing the number of rings in the log-polar image to avoid oversampling the centre, and an example of circle detection in a real image is shown in Fig. 3c. However, the real solution would be to use log-polar hardware that could sample the physical image everywhere at the appropriate resolution.

5 'Eye movements': a resampling strategy

To get some sense of the potential value of log-polar line detection, it is important to simulate the way it might be exploited in an active vision system. To this end, a simple recentring strategy was used to move the log-polar pattern around in conventional images, in rough simulation of saccadic eye movements in the optic array. The strategy adopted is designed to demonstrate the possibilities of the approach rather than to be optimum in any respect.

By analogy with eye movements, a foveation is taken to mean extraction of information for a single sampling centre (x_c, y_c), and a saccade to mean a movement of the sampling centre. Various kinds of saccade were programmed, including:

   (i) a small random step from a Gaussian distribution round the current centre;

  (ii) a random movement to anywhere in the original image with a probability depending on the density of previous sampling;

 (iii) a step to the nearest point on a recently detected line;

 (iv) a step to the intersection of two recently detected lines.

After each foveation the most prominent lines were recorded and one of the steps above was chosen, using a set of fixed probabilities. (This is similar to the Iterated Function System method of generating a fractal set, in which one of a set of affine transformations is chosen at each step using fixed probabilities.)

A first step in building up a description of image structure from the lines picked up by the system is to combine segments from different foveations where these are collinear and overlapping. In the present system, line segments were combined when they were collinear to within the quantisation error of the log-polar representation, and when projected onto a line with the average parameters, they overlapped. This simple approach, sufficient for graphical demonstrations, needs to be developed further and put on a sound statistical basis.

Two examples are shown: in Fig 4a small saccades towards line intersections build up detailed local structure, whilst in Fig 4b large saccades away from areas already covered give a wide coverage of the input image.

6 Discussion

The work reported here has put some flesh on the bones of Weiman and Chaikin's theoretical ideas [11]. Efficient detection of straight lines and circles in log-polar images has been implemented. In the resampled images used
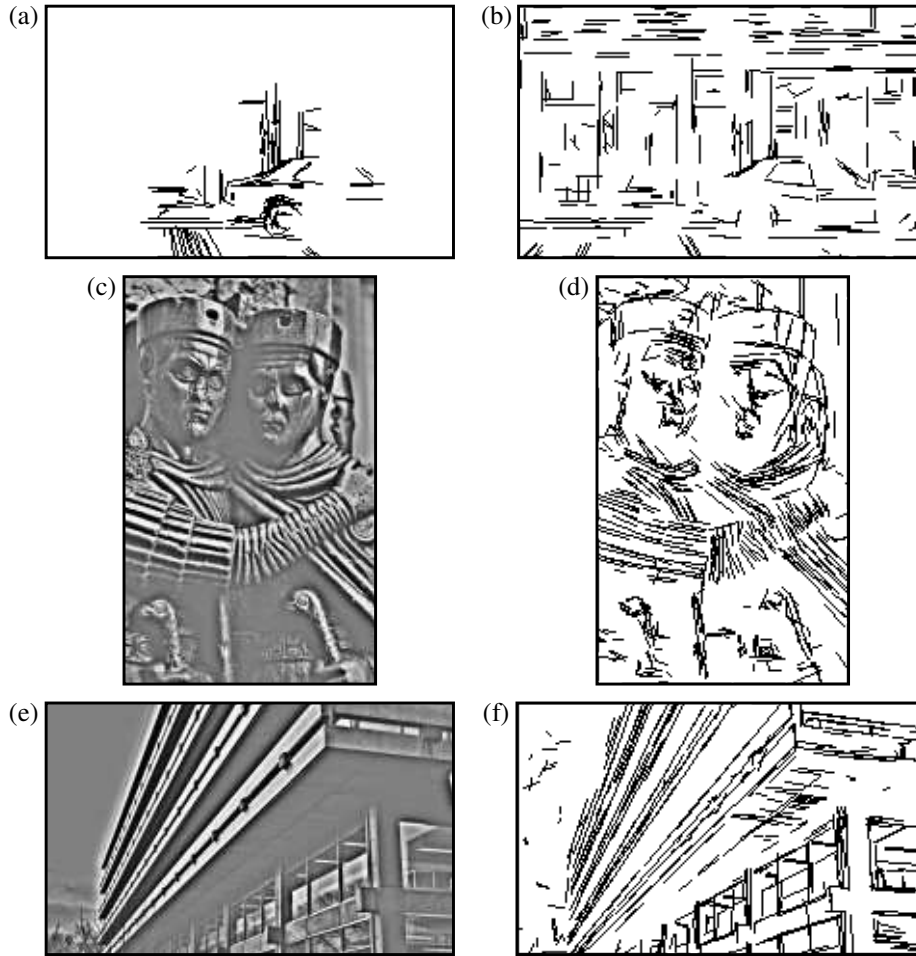
Figure 4: Examples of straight lines accumulated using an eye movement strategy. In all cases $r_{max} = 50$, $n_r = 64$, $n_w = 128$, $r_{min} \approx 2.3$, and the mask is differentiated with respect to R and smoothed with $\sigma=1$. Up to 5 lines from each foveation are drawn, provided their peaks exceed 3 times the s.d. of the convolution output. **(a)** Input image is Fig 2a. Combined line segments from 100 foveations, saccades of type (i) and (iv) with P(type i) = 0.25. **(b)** As (a) but all saccades of type (ii). **(c)** Input image for **(d)**, segments from 500 foveations with all 4 types of saccades equally probable. **(e)**, **(f)** As for (c) and (d) with a different input image.

here, straight line detection is demonstrably effective, but circle detection is hampered by the inaccuracy of the representation. A significant advantage of the approach is that no edge or feature detection precedes the line detection; the main computational cost is an FFT of complexity $n_r n_w \log(n_r + n_w)$ for each foveation, and this could be carried out in suitable hardware.

Processes on log-polar images are not easy to compare with processes on conventional images, since they are designed to be embedded in a system with foveal sampling and an active camera at the hardware level. In particular, the results of log-polar line and circle detection are not comparable with those of,

11

say, edge detectors or the Hough transform operating on conventional images. Weiman [10] has described Hough transform detection of straight lines in log-polar space, but results on real images are not reported.

A full evaluation of the system described here requires further work. Tests on synthetic images show that the method can readily locate the boundaries of polygons where the standard deviation of the noise exceeds the grey-level difference between the interior and exterior, but this is not surprising given the incorporation of smoothing and the integration of evidence from a substantial area of the image. More to the point, Figs 4d and 4f give some indication of the extent to which a combination of log-polar line detection and camera movements might work together to build up a structural representation. An appropriate benchmark would involve a higher-level task which demanded that image structure be extracted.

These processes could play a valuable role if integrated into an active vision system, with spatially variant sampling at the hardware level. However, the most significant challenges are not at the level of feature detection, but lie in developing a strategy for foveation and saccade so as to integrate information effectively. This will require a more task-directed, purposive approach.

References

[1] Bederson, B.B., Wallace, R.S. and Schwartz, E. (1995) A miniaturized space-variant active vision system: Cortex-1. *Machine Vision and Applications.* **8**, 101-109.

[2] Brunnström, K., Eklundh, J-O., and Uhlin, T. (1996) Active fixation for scene exploration. *International Journal of Computer Vision,* **17**, 137-162.

[3] Funt, B.V. (1980) Problem-solving with diagrammatic representations. *Artificial Intelligence,* **13**, 201-230.

[4] Lim, F.L., West, G.A.W. and Venkatesh, S. (1997) Use of log polar space for foveation and feature recognition. *IEE Proc. Vis. Image Signal Process.,* **144**, 323-331.

[5] Peters, R.A. II and Bishay, M. (1996) Centering peripheral features in an indoor environment using a binocular log-polar 4DOF camera head. *Robotics and Autonomous Systems,* **18**, 271-281.

[6] Schwartz, E.L. Spatial mapping in the primate sensory projection: analytic structure and relevance to perception. *Biological Cybernetics,* **25**, 181-194.

[7] Tistarelli, M. and Sandini, G. (1992) Dynamic aspects in Active Vision. *CVGIP: Image Understanding,* **56**, 108-129.

[8] Tistarelli, M. and Sandini, G. (1993) On the advantages of polar and log-polar mapping for direct estimation of time-to-impact from optical flow. *IEEE Trans. Pattern Analysis and Machine Intelligence,* **15**, 401-410.

[9] Tunley, H. and Young, D. (1994) Dynamic fixation of a moving surface using log polar sampling. In Hancock, E. (ed) *Proc. of the 5th British Machine Vision Conf.,* pp 579-588.

[10] Weiman, C.F.R. (1990) Polar exponential sensor arrays unify iconic and Hough space representation. In Casasent, D.P. (ed) *Proc. SPIE Vol 1192: Intelligent Robots and Computer Vision VIII: Algorithms and Techniques,* pp 832-842.

[11] Weiman, C.F.R. and Chaikin, G. (1979) Logarithmic spiral grids for image processing and display. *Computer Graphics and Image Processing,* **11**, 197-226.

[12] Wilson, S.W. (1983) On the retino-cortical mapping. *Int. J. Man-Machine Studies,* **18**, 361-389.

[13] Young, D.S. (1989) Logarithmic sampling of images for computer vision. In Cohn, T. (ed) *Proc. of the 7th Conf. of the Society for the Study of Artificial Intelligence and Simulation of Behaviour,* pp 145-150.