

Circle recognition through a 2D Hough Transform and radius histogramming

Dimitrios Ioannou^a, Walter Huda^b, Andrew F. Laine^{c,*}

^aDepartment of Nuclear Engineering Sciences, University of Florida, Gainesville, FL 32611, USA

^bDepartment of Radiology, University of Florida, Gainesville, FL 32611, USA

^cColumbia University, Center for Biomedical Engineering, 416 CEPSR, MC 8904, 530 West 120th Street, New York, NY 10027, USA

Received 10 February 1997; received in revised form 11 February 1998; accepted 16 February 1998

Abstract

We present a two-step algorithm for the recognition of circles. The first step uses a 2D Hough Transform for the detection of the centres of the circles and the second step validates their existence by radius histogramming. The 2D Hough Transform technique makes use of the property that every chord of a circle passes through its centre. We present results of experiments with synthetic data demonstrating that our method is more robust to noise than standard gradient based methods. The promise of the method is demonstrated with its application on a natural image and on a digitized mammogram. © 1999 Elsevier Science B.V. All rights reserved.

Keywords: Hough Transform; Circle detection; Digitization errors; Radius histogramming; Filtering

1. Introduction

The Hough Transform (HT) is a standard method for shape recognition in digital images [1,2]. It was first applied to the recognition of straight lines [3,4] and later extended to circles [5], ellipses [6] and arbitrarily shaped objects [7]. Its advantages include robustness to noise, robustness to shape distortions and to occlusions/missing parts of an object. Its main disadvantage is the fact that computational and storage requirements of the algorithm increase as a power of the dimensionality of the curve. This means that for straight lines the computational complexity and storage requirements are $O(n^2)$, for circles $O(n^3)$ and for ellipses $O(n^5)$.

In this article we study the problem of circle recognition with the use of the HT. Even though there have been attempts towards the recognition of circles via the standard 3D HT [8], it has been recognized that there is a need for a decomposition of the search space, to simplify the problem both in terms of computation and storage. Many attempts have been based on the property that the normal to a point of the circumference of a circle passes through the centre of the circle [9]. This approach works well for high signal to noise ratios and/or simple environments. As the signal to noise ratio decreases, the accuracy of the gradient estimation decreases [9,10]. The fact that gradient-based methods are

heavily dependent upon the accuracy of the gradient estimation explains why they are not robust to noise. Another disadvantage is that, sometimes, the edge detector of choice does not provide gradient information. A comparative study of various HT based techniques for circle recognition has been performed by Yuen et al. [11].

The approach taken in this article is to decompose the 3D search into a 2D HT and 1D radius histogramming. For the first part, instead of relying on a 2D gradient-based HT we use a 2D bisection based HT. The property we exploit is that the line that perpendicularly bisects any chord of a circle passes through its centre (see Fig. 1). Experimentation with synthetic data demonstrated that our approach is more robust to noise than gradient-based techniques.

The second part of the algorithm, 1D radius histogramming, is used to validate the existence of these circles and calculate their radius. We show that extracting information from the radius histogram is not a trivial task and we devise a filtering technique that solves this problem.

In Section 2, we discuss the steps of our algorithm. In this section we give details about the implementation of the algorithm and analyse the effects of various factors, such as digitization of the image, discretization of the parameter space, noise, etc., on the accuracy and the computational efficiency of the method. In Section 3, we provide test demonstrations of the algorithm with synthetic and real images. In the first part of this section, we use synthetic data to demonstrate the robustness of the technique and

* Corresponding author.

compare it with the gradient-based 2D HT. In the second part of this section, we present results with a natural image and a digitized phantom. Finally, in Section 4, we present conclusions and summary.

2. Algorithm

2.1. A 2D Hough transform based on bisection

After edge detection, the resulting connected components are labelled. Each connected component is expressed as a chain of the coordinates of its points. We connect pairs of points, of the same component, using a sliding window (see Fig. 2). If the coordinates of the points are $A(x_A, y_A)$ and $B(x_B, y_B)$, the equation of the line that perpendicularly bisects AB is

$$y = \frac{x_A - x_B}{y_A - y_B}x + \frac{x_A^2 + y_A^2 - x_B^2 - y_B^2}{2(y_A - y_B)} \quad (1)$$

All points (members of the parameter space) belonging to this line have their votes increased by one. Highly voted points provide an indication of the existence digital circles. These points are the centres of these circles.

Since we are dealing with digital circles the points of their circumference are affected by digitization and, therefore, do not exactly satisfy the standard circle equation:

$$r^2 = (x - x_0)^2 + (y - y_0)^2, \quad (2)$$

where r is the radius of the circle and (x_0, y_0) are the coordinates of the centre of the circle. Further, there is a need for a discretization of the parameter space for two reasons:

1. *Computational efficiency.* It is impossible to account for all the digital circles that may exist in the image, and
2. *The line that perpendicularly bisects a chord of the circle is highly unlikely to pass through its centre.* As a result of the digitization of the image, pixels belonging to a digital circle do not exactly satisfy Eq. (2) and, therefore, their chords do not coincide with the chords of the true circle.

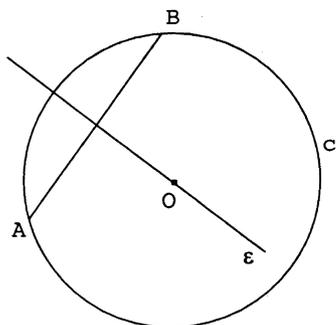


Fig. 1. Demonstration of the fact that the center of circle c , O , belongs to the line that perpendicularly bisects the segment defined by points A and B (both points belong to the circumference of the circle).

The discretization of the parameter space makes the voting scheme robust to noise and errors induced during the edge detection and other preprocessing operations. However, it introduces some error in the determination of the position of centres of circles. The larger the size of the members of the parameter space, the higher the uncertainty of the estimation of the position of the centre of the circle. In our implementation the parameter space is congruent with the *image space* (this is dictated by the adopted parameterization) and the size of a member of the parameter space is the same as the size of a pixel. This choice is reasonable but not necessary. For example, one may choose a more robust scheme where the size of a member of the parameter space is $2h$ by $2h$ (h is the size of the pixel). Such a discretization will not only increase the uncertainty of the estimation of the position of the centre of the circle but will also increase the probability of getting accidental peaks. Its advantage is that there is a smaller chance of missing real centres.

Simple detection of centres of circles is not enough. The reasons are as follows.

1. Highly voted pixels provide only an indication of the existence of a circle. There is a need for verification of this hypothesis. Highly voted local maxima may be formed accidentally, and
2. The need for a determination of the third parameter of the circle (its radius r).

Next we describe an efficient method towards the verification of the existence of a circle, and the extraction of its radius via radius-histogramming.

2.2. Analysis of radius histogram

After detecting possible centres, we can use the histogram of the distances of all feature points from the centres to verify the existence of circles and extract their radii (see Fig. 3). In the ideal case (continuous circle, exactly determined centre) the analysis of the radius histogram would be an easy goal. This is the case because circles would show up as sharp local maxima in a noisy background in the radius

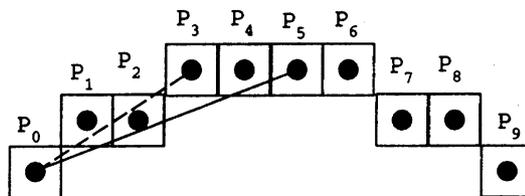


Fig. 2. Figure illustrating the idea of our implementation of a 2D HT for circle center localization. Points P_0 and P_3 of the connected component P_0, P_1, \dots, P_9 , are assumed to belong to the same circle and all points that belong to the line that perpendicularly bisects the line segment they define get voted. The same process is repeated for the pairs $P_1 - P_4, P_2 - P_5$ etc. For this particular case the length of the window is equal to three. If one chooses to connect points $P_0 - P_5, P_1 - P_6$, etc., the length of the window is equal to five.

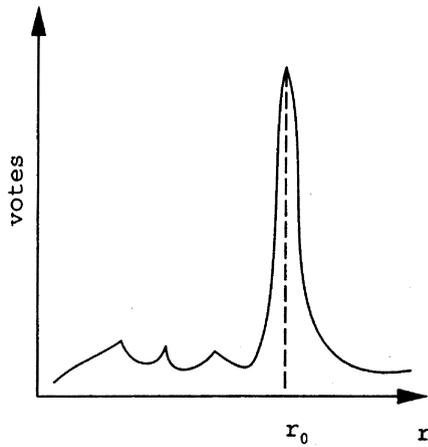


Fig. 3. The radius histogram of center of a circle. All bins get a small number of votes except for the bin to which the circle belongs. A sharp maximum appears for the bin that contains the circle.

histogram. The reasons why this does not happen in the discrete space are:

1. Digitization/discretization errors. Even for the case where we can exactly determine the centre of the circle the digitization of the circle, the digitization of the image combined with the discretization of the histogram make almost certain that the votes of the pixels will be spread to a number of neighbouring bins of the histogram. As one can see from Fig. 4, this effect is similar to the spreading of the straight line standard HT [12]. Under the boundary quantization scheme [13], it can be shown that *digital* circles can be bounded by two *euclidean* circles with the same centre and radii that differ by h [14]. This is illustrated in Fig. 4 where a digital circle is bounded by continuous circles c_1 and c_2 . If Δr is the size of the bins of the histogram the maximum spreading for a circle is equal to:

$$n_r = \lfloor \frac{h}{\Delta r} \rfloor + 2, \tag{3}$$

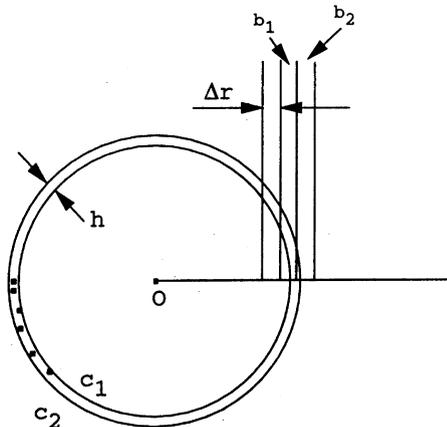


Fig. 4. All pixels belonging to a digital circle can be bounded by two concentric circles, c_1 and c_2 , whose radii differ by h . For the digital circle of this figure, all pixels are spread between two neighboring bins, b_1 and b_2 .

- where the symbol $\lfloor z \rfloor$ denotes the largest integer less than z ,
2. Distortion of the shape caused by imperfections in the image formation, errors during the edge detection stage, and imperfections in the boundary of the object.
3. Noisy pixels and other objects that appear in the neighbourhood of the circle.
4. Pixels missing from the boundary. This can happen because of occlusions, missing parts of the object, etc.
5. Errors in the localization of the centre during the previous steps, mainly during the 2D Hough voting step. If the localization of the centre of the circle is not accurate, two peaks will appear in the radius histogram of the estimated centre (Fig. 5). This was discussed by Yuen et al. [11]. For a digital image if the error in the estimation is small, say less than two pixels, there appears a single extended peak. The length of the peak is 3–4 bins. If the error in the position estimation of the centre is larger, two local maxima appear in the histogram. As the error increases so does the distance between the two peaks. Fig. 5 illustrates these ideas.
6. The number of pixels belonging to a digital circle and a digital ring increases almost linearly. Kulpa showed that as $r \rightarrow +\infty$

$$P_c(r) = 4\sqrt{2}r, \tag{4}$$

where P_c is the number of points of a digital circle with radius r . He also provided experimental evidence that good approximation of the number of pixels belonging to a digital ring (the digital object bounded by two continuous circles whose radii differ by h) is given by

$$P_r(r) = 2\pi r. \tag{5}$$

We should emphasize here, that the equivalent of a bin in a digital image is a digital ring and *not* a digital circle.

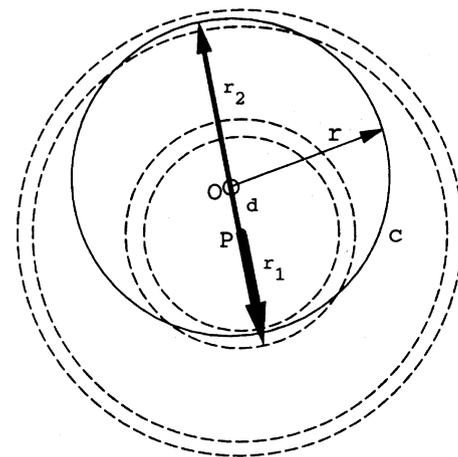


Fig. 5. If an inaccurate estimate of the centre of the circle is provided by the first step, the radius histogram will give two peaks. In this example, the estimated centre is point P , while the true centre is point O . Obviously, these peaks will appear at bins with distance r_1 and r_2 from the estimated centre.

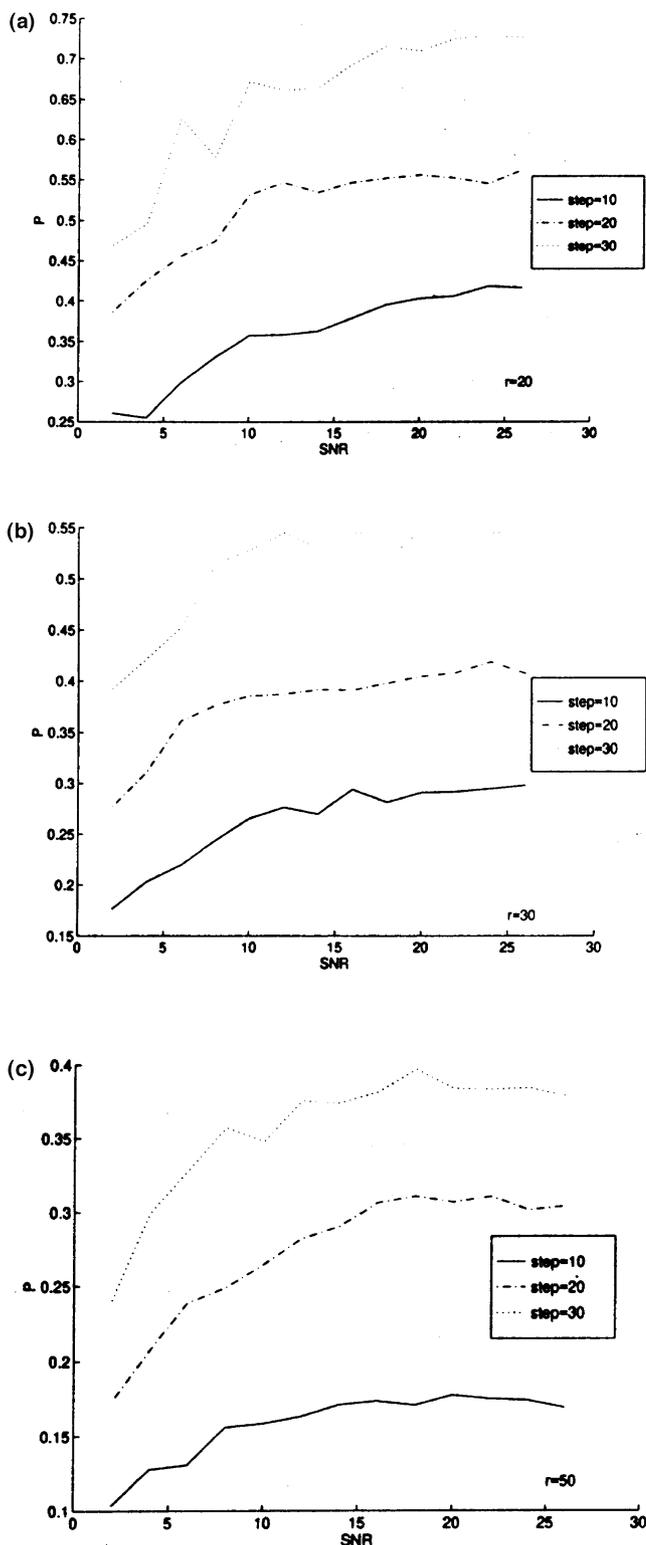


Fig. 6. Comparison of the robustness P as a function of the SNR for three circles of radii: (a) $r = 20$; (b) $r = 30$; and (c) $r = 50$ pixels. As the step becomes larger P increases.

All these factors complicate the assessment of the radius histogram. Our approach towards the extraction of information from the radius histogram is based on the previously presented arguments.

The filter we propose is given by the following equation

$$t = \frac{1}{4\sqrt{2r}} \begin{bmatrix} -\frac{3r}{2(r-2)} & 1 & 1 & \frac{-3r}{2(r+2)} \end{bmatrix}, \quad (6)$$

The details of the derivation of the coefficients of this filter are presented in Appendix A. The divisor of the right hand side of Eq. (6) is equal to the number of pixels belonging to a circle of radius r (see Eq. (4)). We should add that the proposed filtering technique gives an unbiased estimator of the *normalized, unoccluded part* of the circumference of the circle. In other words if no circular features exist in the image and the image is corrupted with uniform noise the members of the histogram will have zero mean.

The resulting filtered histogram is searched for local peaks whose values exceed a certain threshold. As a result of the normalization of the filtered histogram the threshold does not depend on the radius of the circle. This threshold represents the percentage of the circumference of the circle present in the image. For instance, if we want to detect circles with 70% unoccluded circumference, the threshold local maxima have to exceed is 0.70. One should also consider the effects of shape distortion, which would result in a spreading of the votes. If we want to take this effect into account we have to set the threshold to a lower value (say 0.5).

3. Results

In this section we present results from our experiments with the algorithm with real and synthetic images. Our purpose is to study the variation of the robustness of the bisection-based HT with increasing noise and how it compares with gradient-based HTs. As a measure of the robustness we use the robustness ratio P , defined as

$$P = \frac{v_{\text{peak}}}{v_{\text{total}}}, \quad (7)$$

where v_{peak} is the number of votes of the peak cell and v_{total} is the number of pixels of the circumference of the circle. Obviously, because of the digitization of the image, even in the ideal case, P is less than 1.

3.1. Length of window

The first thing we would like to study is the dependence of P with the size of the window that is slid along the boundary of the object. Fig. 2 shows examples of windows of different sizes slid along a digital curve. Fig. 6 shows plots of P for three circles of radii 20, 30, and 50 pixels. We notice that as the radius increases so does P . We also notice

that as the size of the window increases so does P . Theoretical analysis that would indicate the optimum window size is difficult because errors in the coordinates of the two extreme points of the window are not independent to each other. Such a study was presented by Amir [15]. The assumptions, this study relied upon, were independent errors in the coordinates of the two extreme points of the window and an a priori knowledge of the radius of the circle. Our effort is towards the recognition of circles of *varying radii*. The length of the window should be kept as low as possible. The reason for that is the need for recognition of occluded or distorted circles. For these two cases the larger the window the less points that contribute to the creation of peaks in the parameter space. For the purpose of this research we chose a window of length equal to twenty as a good compromise between robustness to noise and robustness to missing parts.

3.2. Synthetic images

3.2.1. Image formation

The purpose of using synthetic images is to compare the bisection-based HT with gradient-based HT in terms of robustness and to extract conclusions about the accuracy of the technique as a function of the Signal to Noise Ratio (SNR). Our effort was to replicate the image acquisition model presented by Lyvers and Mitchell [10]. According to this model the grey level value of a pixel is given by the following equation

$$I(k, l) = \int_{(k-0.5)\Delta x}^{(k+0.5)\Delta x} \int_{(l-0.5)\Delta y}^{(l+0.5)\Delta y} f(x, y) dx dy, \quad (8)$$

where $f(x, y)$ denotes the continuous image and $I(k, l)$ denotes the digital image. An image containing a circle, modelled with this scheme, is presented in Fig. 7(a). One can easily notice the smoothness of the edges of the circle. The image shown in this figure is corrupted with Gaussian noise of standard deviation 1. The background value is equal to 120 and the foreground 140. The SNR is calculated by the following equation [10]:

$$SNR = 20 \log_{10} \frac{c}{\sigma} \quad (9)$$

where c is the contrast and σ is the standard deviation of the noise.

3.2.2. Examples

Fig. 7(b) shows the resulting image after edge detection and thresholding of the image of Fig. 7(a). The edge detection method of choice was the one proposed by Canny [16]. In Fig. 7(c) we present the resulting voting space in a mesh format. The coordinates of the identified peak were (128,128) and they exactly coincided with the coordinates of the original circle. In Fig. 7(d) we show the radius histogram and in Fig. 7(e) we show the filtered histogram. Finally, in Fig. 7(f), we superimpose the detected circle on the original image.

The same process was followed for an image of a low SNR and the results are presented in Fig. 8. We note here that:

1. the number of votes of the centre is smaller in the low SNR case than in the high SNR , and
2. the spreading of the votes in the radius histogram is higher in the low SNR (more bins share the same votes). To counter these two problems, one has to set the vote threshold of the 2D HT to lower values and to use a larger window during the histogram filtering stage. A final note is that the fact that the value of the filtered histogram is greater than one is attributed to the votes caused by noisy pixels.

3.2.3. Comparison of bisection-based with gradient-based HT

Robustness to noise and accuracy in the determination of the centre of the circle are the two main issues one should be concerned when he is using the bisection-based 2D Hough Transform for centre detection. We use the robustness ratio P , defined by Eq. (7), and the distance of the detected centre and the true centre as measures of the performance of the algorithms. We performed experiments for a wide range of SNRs (2 to 26) for various radii. Experiments were repeated for 20 times and mean values are plotted in the figures that follow.

In Fig. 9 we present plots of the robustness P , as defined by Eq. (7) as a function of the SNR for circle of radii 20, 30, and 50 pixels. Comparing the P of the gradient-based HT [17, p. 212] and the bisection-based HT, we can conclude that the former is advantageous for high $SNRs$, while the latter is far more robust for low $SNRs$.

It was also found that the accuracy in the determination of the centre for low SNR with the bisection-based method did not decrease considerably while the gradient-based method gave large errors (see Fig. 10).

3.3. Real images

3.3.1. Ball

In Fig. 11(a) a 256 by 256 image containing a soccer ball is shown. Fig. 11(b) shows the edge detected image. Fig. 11(c) shows a plot of the histogram and Fig. 11(d) shows a plot of the filtered histogram. A local maximum was found for $r = 21$. Fig. 11(e) shows the results of filtering with a Laplacian filter, proposed by Kierkegaard [18]. The problems of this filter are apparent. While it correctly identifies a peak for the true circle it enhances the accidental peak that appears for $r = 75$. Finally, in Fig. 11(f) the detected circle is superimposed on the original image. The relatively low value of the local peak of the filtered histogram, $h_{\max} = 0.38$, is attributed to the increased spreading of the digital object (soccer ball). A more robust version of the filter of Eq. (6) given by the following equation

$$t = \frac{1}{4\sqrt{2}r} \left[-\frac{5r}{2(r-3)} \quad 1 \quad 1 \quad 1 \quad 1 \quad \frac{-5r}{2(r+3)} \right], \quad (10)$$

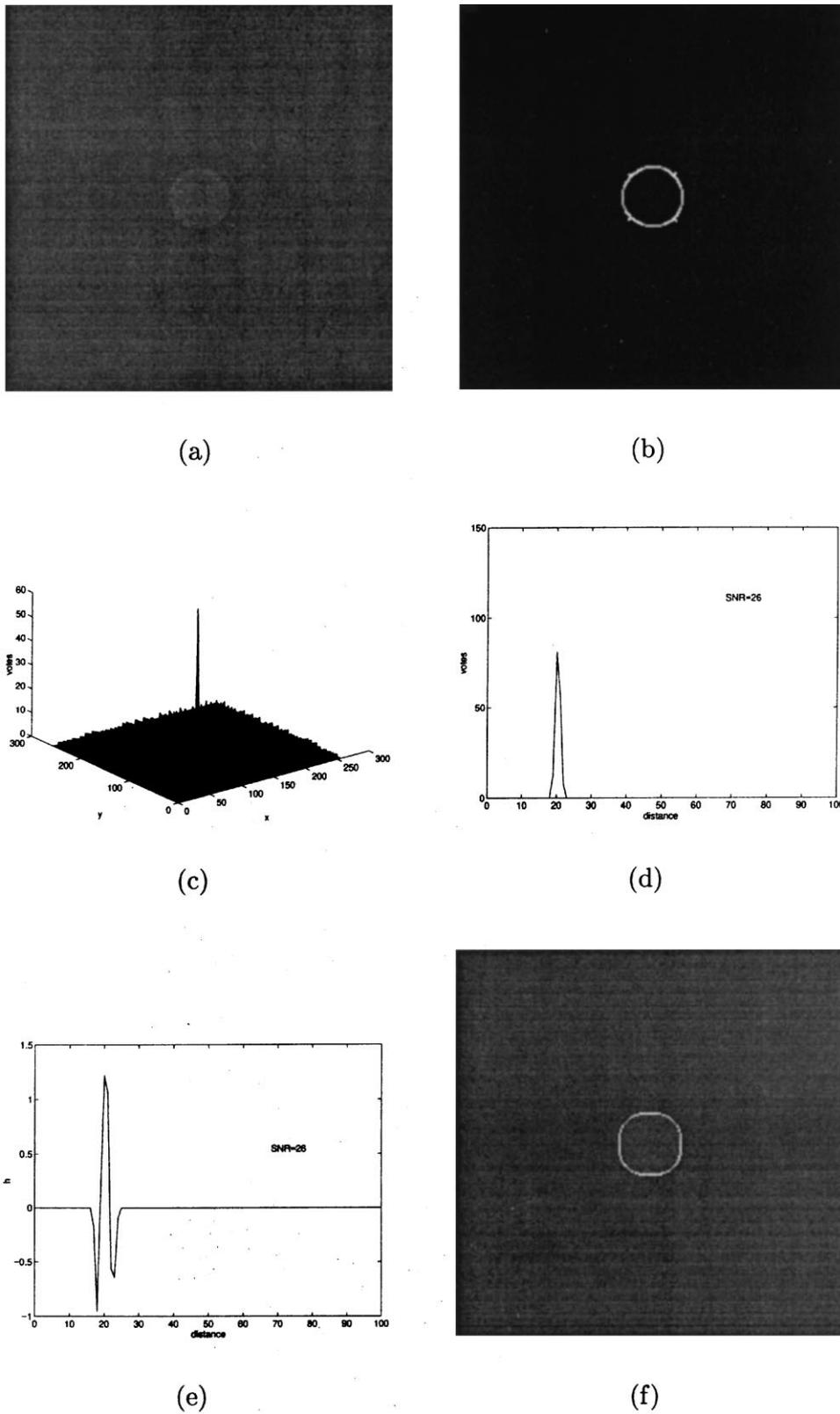


Fig. 7. Figure illustrating the various stages of the presented algorithm for a high *SNR* synthetic image ($SNR = 26$). (a) Original image with a circle at its centre. The radius of the circle was 20 pixels; (b) edge detected image; (c) plotting of the voting space in a mesh format. The peak indicates the possible existence of a circle. The coordinates of the peak provide an estimation of the coordinates of the centre of the circle; (d) histogram of the feature pixels as a function of their distance from the identified centre (peak in the voting space); (e) filtered histogram. The identified peak was at $r = 20$; (f) detected circle superimposed on the original image.

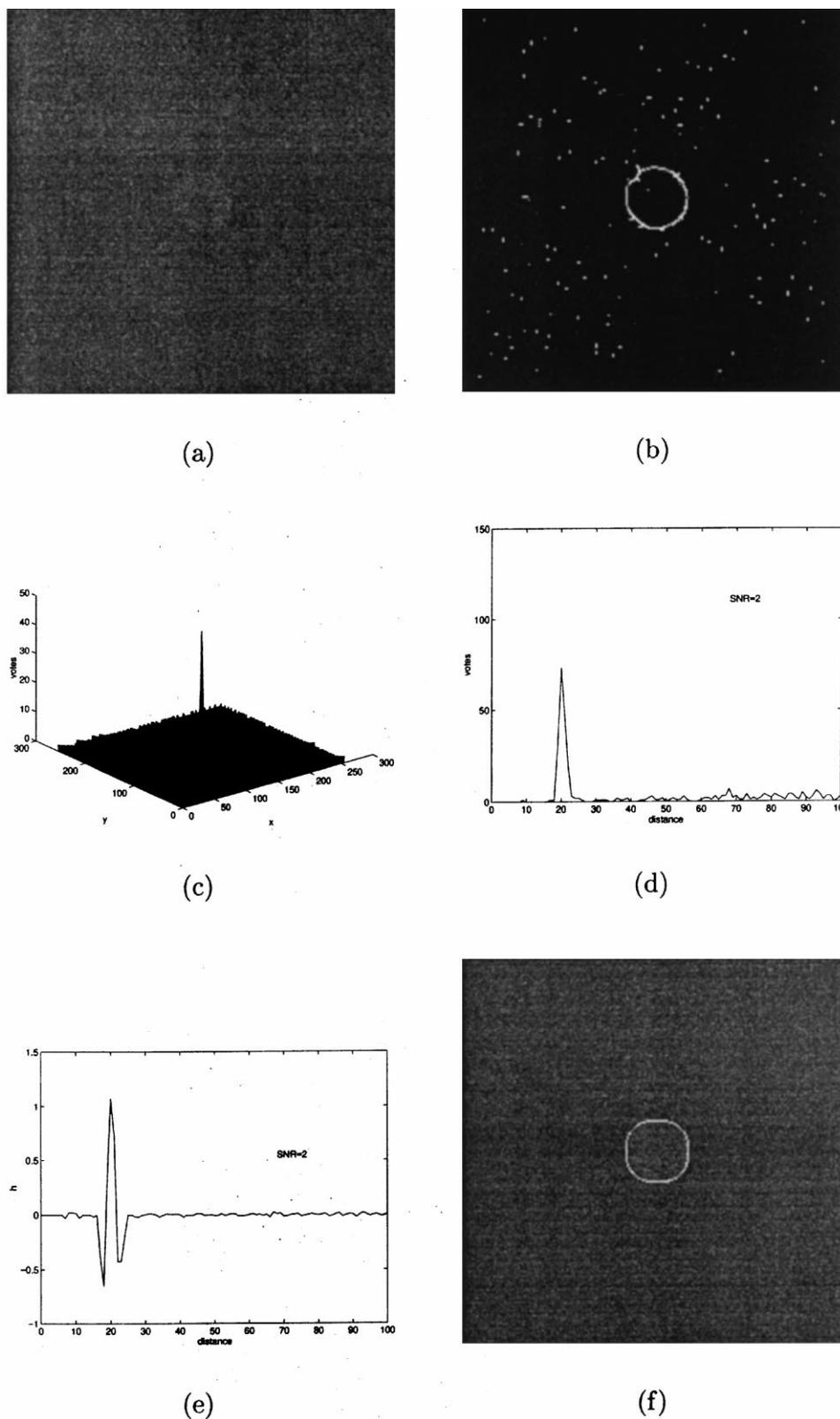


Fig. 8. Figure illustrating the various stages of the presented algorithm for a low SNR synthetic image ($SNR = 2$). (a) Original image with a circle at its centre. The radius of the circle was 20 pixels; (b) edge detected image; (c) plotting of the voting space in a mesh format. The peak indicates the possible existence of a circle. The coordinates of the peak provide an estimation of the coordinates of the centre of the circle; (d) histogram of the feature pixels as a function of their distance from the identified centre (peak in the voting space); (e) filtered histogram. The identified peak was at $r = 20$; (f) detected circle superimposed on the original image.

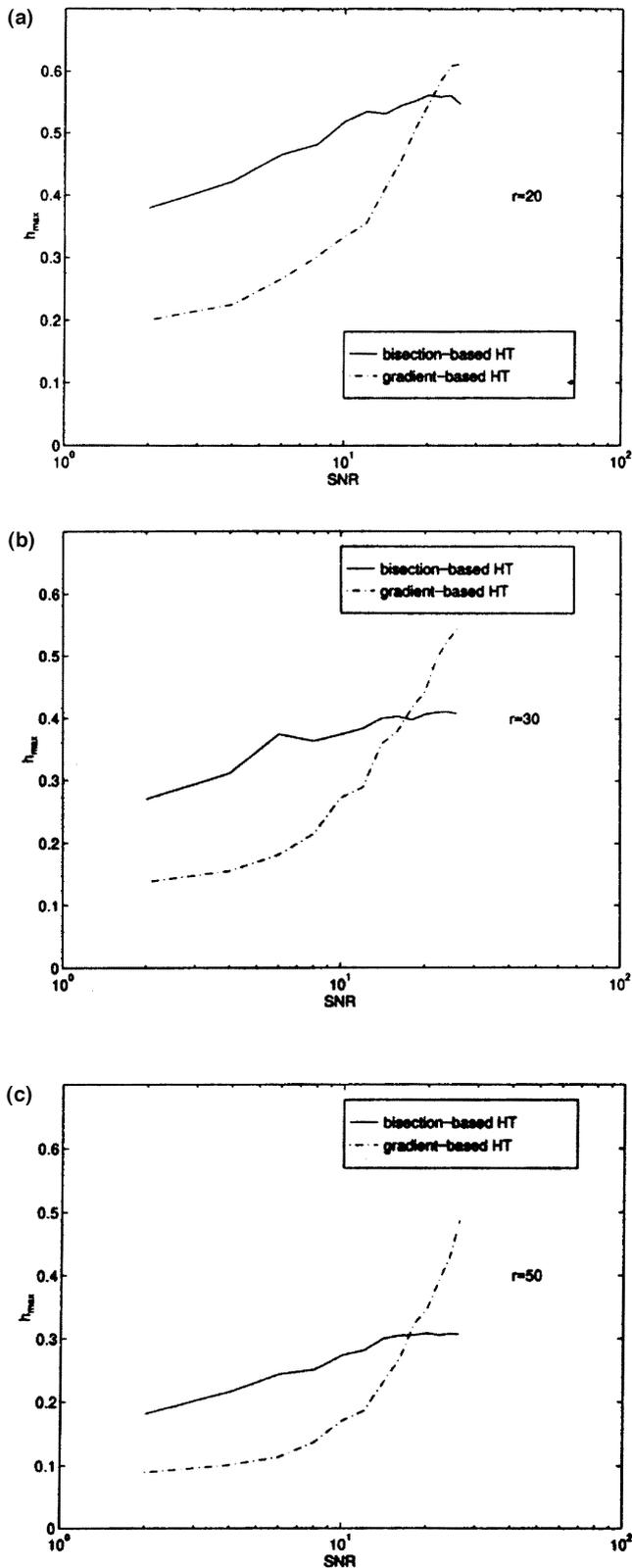


Fig. 9. Plots of the mean value of the maximum value of the filtered histogram h_{\max} , for 20 experiments, as a function of the SNR. Continuous lines denote results using the bisection-based HT, while dashed lines denote results of the gradient-based HT, for circles of radii: (a) $r = 20$; (b) $r = 30$; and (c) $r = 50$. While for high SNR (i.e. $SNR > 20$) the gradient-based HT works better, for low SNR drop the bisection-based HT gives better results.

gave $h_{\max} = 0.89$. What was said for the filter of Eq. (6) holds also for the filter of Eq. (10) with the only exception that, because the votes of five, instead of three, bins are added it is more robust to shape distortions (spreading of the circle).

Fig. 12(a) shows a radiographic image of an RMI 156 (Gamex Inc., Middleton, WI) mammographic accredited phantom. Three types of mammographic features are visible within this phantom image, simulating different sized circular malignant masses as well as small microcalcifications and fibres of varying contrast. Small circumscribed lesions are one of the more important signs of breast cancer which may be detected by mammography in asymptomatic women. The detection of lesions in digital mammograms, is one of the main areas digital mammography has focused on [19,20]. Mammographic phantoms like the one shown in Fig. 12(a) are used to measure the performance of mammographic systems and a mammography unit needs to be able to visualize at least four fibres, three groups of microcalcification specks and three masses to pass the American College of Radiology accreditation program in the United States. The size of the image is 512 by 512 pixels. Fig. 12(b) shows the results of edge detection. Eighteen circle centres were detected, when the voting space was searched for local maxima that had more than 30 votes. After radius histogramming, histogram filtering with the filter of Eq. (6) four circles were detected. These four circles are superimposed on the original image in Fig. 12(c).

4. Conclusions

We presented a two-step HT for the detection and localization of circles in digital images. The first uses a bisection-based 2D HT. Experiments with synthetic data suggest that the bisection-based HT is more robust to noise than the gradient-based HT while being more accurate. Davies was the first to use this property for the localization of centres of circles [21]. His implementation is very efficient in terms of computation but since it only uses vertical and horizontal chords it lacks robustness.

The second step uses radius histogramming to detect the circle and extract its radius. Enhancement of local maxima with the use of a Laplacian filter as suggested by Kierkegaard [18], has the significant shortcoming that it does not normalize for the increasing number of pixels belonging to a bin as we go away from the centre of the circle (see Eq. (5) and (4)). Further, there is no intuitive method for the selection of the threshold of the filtered histogram. Our filtering scheme, considers votes caused by noisy pixels, shape distortions and normalizes to account for the dependence of the number of pixels on the radius of the circle.

Sample results were provided and showed that the method is capable of detecting circle in various environments and of varying sizes and shape distortions.

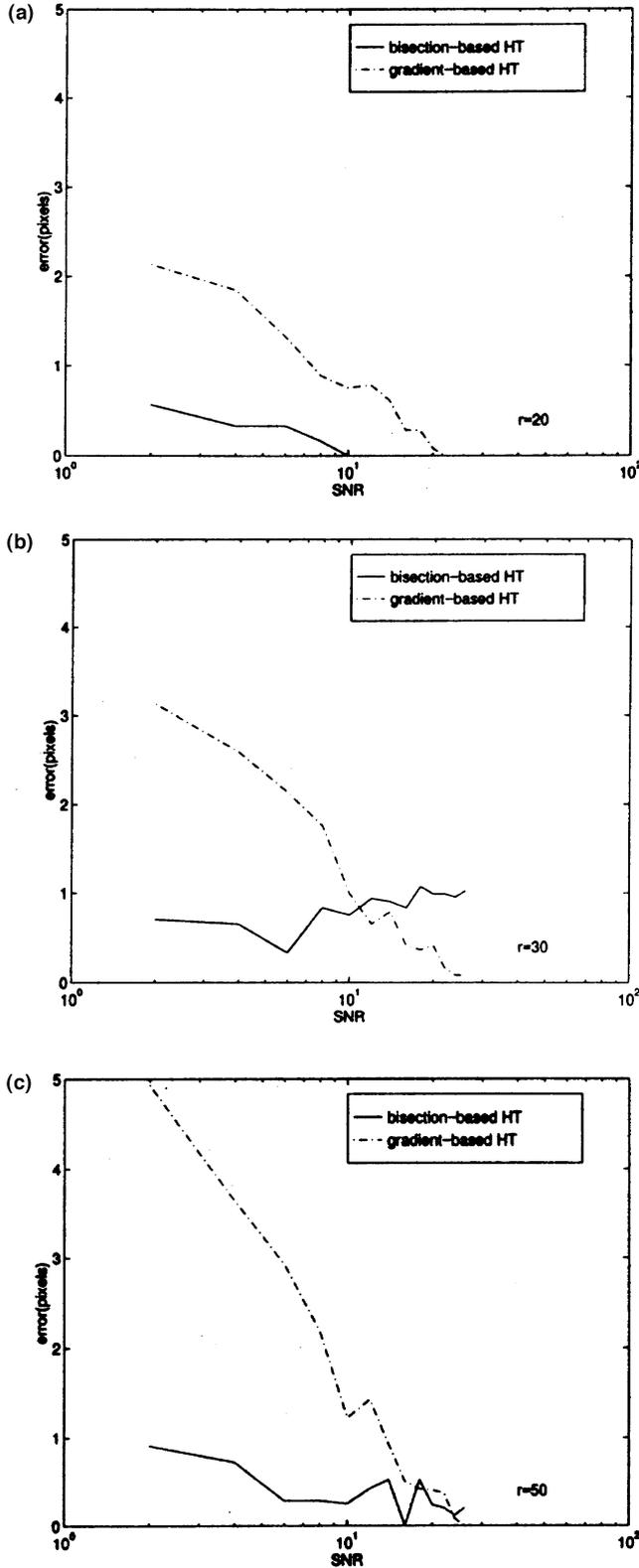


Fig. 10. Plots of the error in the determination of the centre of the circle with the use of the bisection-based method (continuous lines) and the gradient-based method (dashed lines) for circles of radii: (a) $r = 20$; (b) $r = 30$; and (c) $r = 50$. For most of the cases the error of the gradient-based method is larger than the error of the bisection-based method. As it is obvious from the three plots, the error of the bisection-based method for a wide range of noise levels remained smaller than a pixel.

Appendix A Derivations

In this Appendix we will derive the filter shown in Eq. (6). We will rely on the assumption that we are dealing with an image corrupted with uniform noise and that we are looking for a perfect digital circle. As we noticed (see Eq. (3)) earlier for $\Delta r = h$ the maximum number of bins a circle spreads its votes is equal to 2. Therefore, if we pass the radius histogram with the filter

$$t = [11]$$

we will get an estimate of the pixels the circle contains. We would like the filter to be symmetric around the central bin therefore we choose

$$t = [111].$$

The resulting array contains the number of pixels of the circle plus votes caused by noise, other objects etc. We need to enhance our scheme with a module that estimates the votes due to noise and subtracts them. For this purpose we will use two more bins (the outer ones). Therefore our filter is

$$t = [\alpha 111\beta]. \quad (11)$$

To get values for α and β we will require for our filter to give an unbiased estimator with minimum uncertainty (standard deviation). We assume that noise is spatially uniform. To have an unbiased estimator means that if the image is corrupted with noise and no circular object is present the resulting (processed) histogram will have mean value equal to zero.

It is easy to see that the r th bin of the histogram represents a ring [14] whose radius is equal to r . Kulpa provided experimental evidence that a good approximation to the number of pixels belonging to such a ring is $2\pi r$. This estimation is based on empirical evidence [14] and until now no theoretical justification to it was found. We verified, through simulations, that this formula holds with adequate accuracy for radii in the range $[h, 512h]$. It is easy to show, that the number of votes the r th bin takes caused by noise follows a binomial distribution with mean value:

$$\mu = p(2\pi r) \quad (12)$$

and standard deviation:

$$\sigma = \sqrt{p(1-p)(2\pi r)}, \quad (13)$$

where p is the probability that a pixel is noisy.

From Eq. (12) and the constraint we imposed that our estimator should be unbiased we get

$$\mu_F = \alpha(r - 2h) + (r - h) + (r) + (r + h) + \beta(r + 2h) = 0. \quad (14)$$

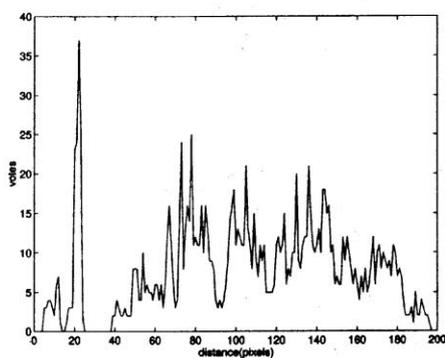
The second equation that we will use comes from the requirement of minimization of the standard deviation of the resulting filtered histogram. The standard deviation of



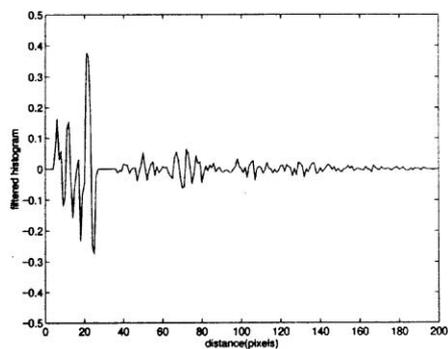
(a)



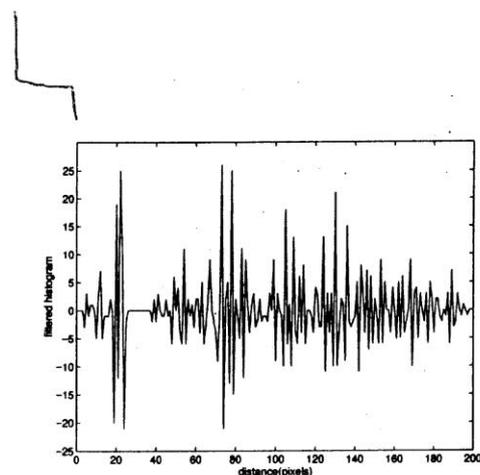
(b)



(c)



(d)



(e)



(f)

Fig. 11. Figure illustrating the steps of the algorithm on real data. (a) Image containing a circular soccer ball; (b) results of Canny edge detection; (c) plot of the radius histogram; (d) plot of the filtered radius histogram; (e) plot of the filtered radius histogram with the use of the Laplacian filter as proposed by Kierkegaard [18]; (f) detected circle superimposed on the original image.

the filtered space is given by the following formula:

$$\sigma_F = 2\pi \sqrt{\alpha^2(r-2h)^2 + (r-h)^2 + (r)^2 + (r+h)^2 + \beta^2(r+2h)^2}.$$

If we make the following substitutions:

$$X1 = \alpha(r-2h)$$

$$X2 = r-h$$

$$X3 = r$$

$$X4 = r+h$$

$$X5 = \beta(r+2h)$$

to the Eq. (14) and (15) we get

$$X1 + X2 + X3 + X4 + X5 = 0, \quad (16)$$

and

$$\sigma_F = 2\pi \sqrt{X1^2 + X2^2 + X3^2 + X4^2 + X5^2} \quad (17)$$

If we solve Eq. (16) for $X1$, and substitute into Eq. (17), and differentiate Eq. (17) with respect to β we get:

$$\frac{\partial \sigma_F}{\partial \beta} = (2X5 + 2(X2 + X3 + X4 + X5))f = 0, \quad (18)$$

where f is a factor that cannot become zero (a constant divided by a function of $X5$).

From Eq. (14) and (18) we get α and β . The resulting filter is given in the main part of the paper (Eq. (6)).

References

- [1] V.F. Leavers, Survey: which hough transform?, *Computer Vision Graphics and Image Processing:Image Understanding* 58 (1993) 250–264.
- [2] J. Illingworth, J. Kittler, Survey: a survey of the Hough Transform, *Computer Vision, Graphics and Image Processing* 44 (1988) 87–116.
- [3] A. Rosenfeld, *Picture Processing by Computer*, Academic Press, New York, 1969.
- [4] R.O. Duda, P.E. Hart, Use of the Hough Transformation to detect lines and curves in pictures, *Communications of the Association of Computing Machinery* 15 (1972) 11–15.
- [5] E.R. Davies, A modified Hough scheme for general circle location, *Pattern Recognition Letters* 7 (1987) 37–43.
- [6] R.K.K. Yip, P.K.S. Tam, D.N.K. Leung, Modification of Hough transform for circles and ellipses detection using a 2-dimensional array, *Pattern Recognition* 25 (1992) 1007–1022.
- [7] D.C.W. Pao, H.F. Li, R. Jayakumar, Shapes recognition using the straight line Hough transform: theory and generalizaion, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14 (1992) 1076–1089.
- [8] G.L. Foresti, C.S. Regazzoni, G. Vernazza, Circular arc extraction by direct clustering in a 3d Hough parameter space, *Signal Processing* 41 (1995) 203–224.
- [9] E.R. Davies, The effect of noise on edge orientation computations, *Pattern Recognition Letters* 6 (1987) 315–322.
- [10] E.P. Lyvers, R. Mitchell, Precision edge contrast and orientation estimation, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 10 (1988) 927–937.

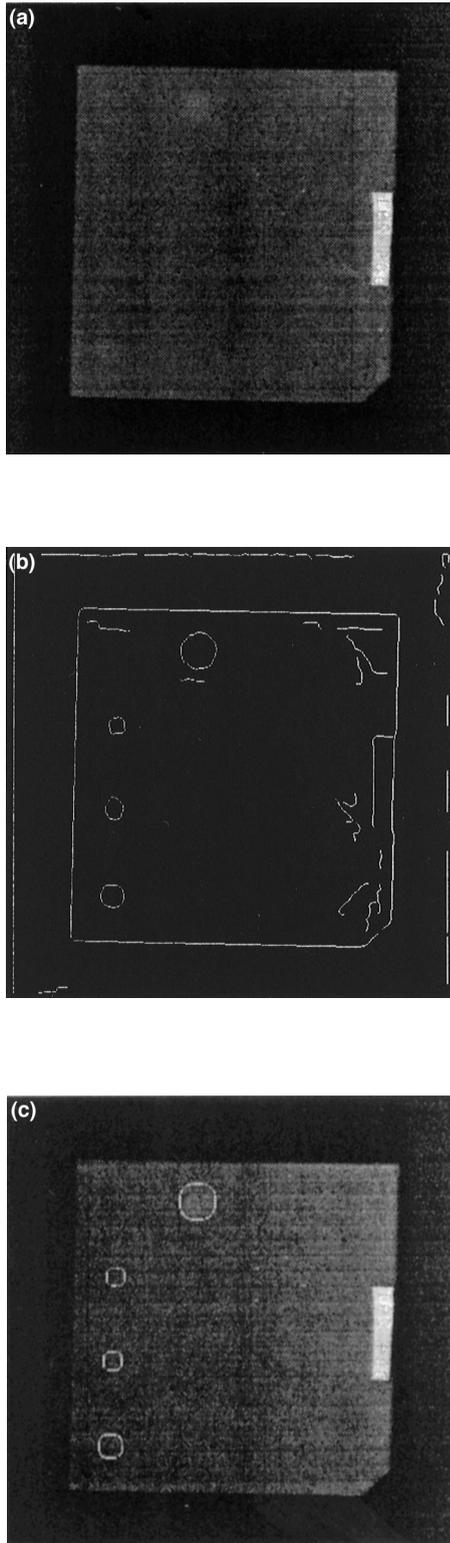


Fig. 12. Figure illustrating the steps of the algorithm on real data. (a) Image of a phantom; (b) results of Canny edge detection; (c) detected circles superimposed on the original image.

- [11] H.K. Yuen, J. Princen, J. Illingworth, J. Kittler, Comparative study of Hough Transform methods for circle finding, *Image and Vision Computing* 8 (1990) 71–77.
- [12] T.M. van Veen, F.C. Groen, Discretization errors in the Hough Transform, *Pattern Recognition* 14 (1981) 137–145.
- [13] M. Worring, W.M. Smeulders, Digitized circular arcs: characterization and parameter estimation, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 17 (1995) 587–598.
- [14] Z. Kulpa, On the properties of discrete circles, rings, and disks, *Computer Graphics and Image Processing* 10 (1979) 348–365.
- [15] I. Amir, Algorithm for finding the centre of circular fiducials, *Computer Vision, Graphics and Image Processing* 49 (1990) 398–406.
- [16] J. Canny, A computational approach to edge detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 8 (1986) 679–698.
- [17] E.R. Davies, *Machine Vision*, Academic Press, London, 1990.
- [18] P. Kierkegaard, A method for detection of circular arcs based on the Hough Transform, *Machine Vision and Applications* 5 (1992) 249–263.
- [19] A.F. Laine, S. Schuler, J. Fan, W. Huda, Mammographic feature enhancement by multiscale analysis, *IEEE Transactions on Medical Imaging* 13 (4) (1994) 725–740.
- [20] K. Doi, M.L. Giger, R.M. Nishikawa and R.A. Smith (Eds), *Evaluation of a Hough Transform method for circumscribed lesion detection*, Elsevier Science B.V., 1996.
- [21] E.R. Davies, A high speed algorithm for circular object location, *Pattern Recognition Letters* 6 (1987) 323–333.