Detecting Halftone Dots for Offset Print Quality Assessment Using Soft Computing

Jens Lundström and Antanas Verikas

Lundström J, Verikas A. Detecting Halftone Dots for Offset Print Quality Assessment Using Soft Computing. In: 2010 IEEE International Conference on Fuzzy Systems (FUZZ). Piscataway, NJ: IEEE; 2010. p. 1145-1151. FUZZ-IEEE, 2010.

DOI: http://dx.doi.org/10.1109/FUZZY.2010.5584433
Copyright: IEEE

Detecting Halftone Dots for Offset Print Quality Assessment Using Soft Computing

Jens Lundström, and Antanas Verikas, *Member, IEEE*

***Abstract*— Nowadays in printing industry most of information processing steps are highly automated, except the last one–print quality assessment and control. We present a way to assess one important aspect of print quality, namely the distortion of halftone dots printed colour pictures are made of. The problem is formulated as assessing the distortion of circles detected in microscale images of halftone dot areas. In this paper several known circle detection techniques are explored in terms of accuracy and robustness. We also present a new circle detection technique based on the fuzzy Hough transform (FHT) extended with k-means clustering for detecting positions of accumulator peaks and with an optional fine-tuning step implemented through unsupervised learning. Prior knowledge about the approximate positions and radii of the circles is utilized in the algorithm. Compared to FHT the proposed technique is shown to increase the estimation accuracy of the position and size of detected circles. The techniques are investigated using synthetic and natural images.**

I. Introduction

Offset printing is the most widely used commercial printing process in the production of newspapers and magazines. Multicolour pictures in offset printing are created by printing cyan (C), magenta (M), yellow (Y), and black (K) dots of varying sizes upon each other having different raster angles. Fig. 1 (*left*) illustrates an example of an image taken from an offset-printed picture. When regarding a region of the same image at a microscopic scale the halftone dots appears, Fig. 1 (*right*).

Fig. 1. A four-colour offset print (*left*) and offset halftone dots (*right*).

Printing press operators are striving to maintain as high and constant quality as possible. However, controlling printing quality is a rather complicated matter since it is influenced by a variety of factors. Printing press settings, parameters of paper and inks, the quality of printing plates, even seasonal variables, such as humidity and temperature, may affect print quality.

To maintain printing quality in a predetermined range, the press operator samples the print manually throughout the job run. The sample is compared to the approved sample print and a great effort is made to compensate for colour deviations detected in the print. Usually, the initial colour assessment is visual. If deviations noticeable to the operator eye are detected, a densitometer or a spectrophotometer is then applied to get a numerical evaluation of the colour deviations and ink density adjustments are performed to compensate for the deviations. Each operator performs the necessary adjustments based on experience gained from working at that particular press. Typically, the perception of the printed result is very subjective and consequently great variations may appear depending on the operator controlling the process.

There are many other print quality attributes, in addition to correct ink density level. Evenness [12], stability of size and shape of halftone dots, dot gain (the increase of halftone dot size from a printing plate to the actual printed result) are several such attributes. Many of print quality attributes, including dot gain and ink density [6], can be assessed by analyzing colour camera images taken from designated printed areas, known as halftone areas or grey bars, shown in Fig. 2.

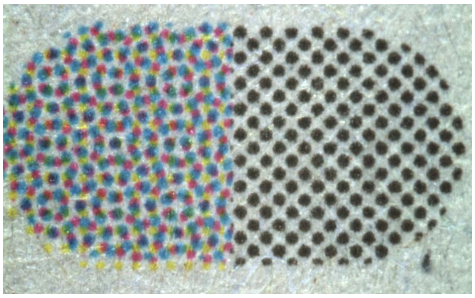

Fig. 2. Double grey bar used to measure quality attributes of offset print.

To assess many of print quality attributes, printed dots need to be accurately detected individually. Fig. 3 presents an example of a grey bar illustrating that shape analysis of individual dots can easily reveal the lateral and circumferential slurring occurring in the print.

Ink trapping in halftone areas is another print quality attribute requiring individual detection of halftone dots. Trapping can be described as the ability of one printed colour to be perceived when printed on top of another colour and can

Jens Lundström is with the Intelligent Systems Laboratory, Halmstad University, S-30118, Box 823, Halmstad, Sweden; (email: jens.lundstrom@hh.se).

Antanas Verikas is with the Intelligent Systems Laboratory, Halmstad University, S-30118, Box 823, Halmstad, Sweden; and with the Department of Electrical & Control Equipment, Kaunas University of Technology LT-51368, Studentu 50, Kaunas, Lithuania (email: antanas.verikas@hh.se).

978-1-4244-8126-2/10/$26.00 ©2010 IEEE

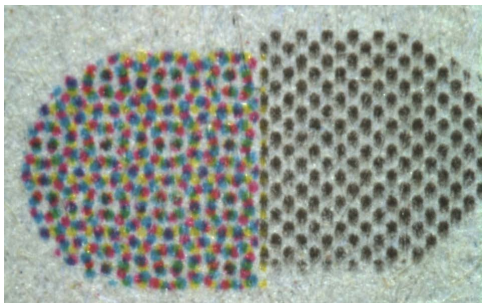

Fig. 3. Circumferential slurring occurring at the black area of a double grey bar.

be calculated by the following quotient:

$$T = \frac{D_{1+2} - D_1}{D_2} 100 \tag{1}$$

where D_1 is the density of the first printed colour, D_2 is the density of the second colour, and D_{1+2} is the density of the colour overprint. Fig. 4 illustrating the measurement of ink trapping in halftone areas, reveals the necessity of identifying the geometric areas occupied by individual halftone dots.

Fig. 4. Illustration of theoretical trapping calculation using printing dots.

Halftone dots are aimed to be printed as they are defined on the printing plates with a certain expectation in dot gain. The dot shape is usually defined as circular with a certain radius. Thus, circle detection algorithms can by applied to identify individual halftone dots.

The objective of this work is to explore the ability of different circle detection techniques to identify halftone dots in grey bars. Some of these techniques can even assess the distortion present in each halftone dot. This is a desired property, since by estimating the center, radius and distortion for each halftone dot in the grey bar area, we could thereby estimate most of the key print quality factors such as the degree of variation of size and shape of halftone dots, register errors, trapping, mottling, and dot gain.

The overall goal of the project this work is part of is to develop an image analysis and computational intelligence-based decision support system to assist printing press operators, manufacturers of ink and paper. Measurements available from online image acquisition system described in [13] create a database containing print quality attributes and numerous parameters characterizing the printing press, printing process, and paper. The collected data will then be mined to explore complex relationships between print quality and the variables involved in the process of colour offset printing.

This paper is organized as follows. In Section II, we introduce the circle detection techniques explored. The way used to assess the accuracy, sensitivity to noise, computational load, and the speed of convergence of the techniques is also presented in this section. Section III presents the data and results of experimental investigations. Conclusions are given in Section IV.

II. Methods

Automatic detection of circles in images can be found in a wide set of applications, ranging from analysis of football games [1], banknote printing [2], exploration of lunar craters [3] to iris detection [4]. Common difficulties for all these algorithms seems to be the lack of precision when subjected to background noise, shading and occlusion of the circular objects. In some cases a priori knowledge, such as knowledge of the sun position when classifying footballs [1], can increase the detection performance. To device a robust algorithm for detecting halftone dots, a priori information, such as centers and radius of the dots, available from the Computer-To-Plate data (CTP) is exploited in this work. The algorithm should also provide the distortion degree of the dots.

A. Circle Detection Based on the Hough Transform

The Hough Tranform (HT) is perhaps the most famous technique for detecting circles. HT was proposed for detecting particle tracks in bubble chambers [10]. Later the technique was amended to be valid for more general curve fitting, e.g. detection of a circle [11]

$$(x-a)^2 + (y-b)^2 = c^2 \tag{2}$$

where a, b, and c are parameters of the circle—the center point coordinates and the radius. A pixel with coordinates x, y satisfying Equation (2) will contribute to the parameter space. This can be considered as a voting mechanism for candidate parameter vectors.

B. The Fuzzy Hough Transform

In the fuzzy Hough transform (FHT) case, a shape is considered as a fuzzy set with a given membership function, where each pixel belongs to a shape defined by a specific parameter vector to some extent. By introducing fuzzy support to the Hough transform even pixels being close but not exactly lying on a circle may be included in the voting process for a certain parameter vector. For example, Basak and Pal [7] define a circle as a fuzzy set $\mathcal{M}$ with the following membership function:

$$h((x,y);(x_0,y_0),\epsilon) = \begin{cases} 1 - 2\left(\frac{d}{\epsilon}\right)^2 & \text{if } |d| < \frac{\epsilon}{2}, \\ 2\left(1 - \frac{|d|}{\epsilon}\right)^2 & \text{if } \frac{\epsilon}{2} \le |d| < \epsilon, \\ 0 & \text{if } |d| \ge \epsilon \end{cases} \tag{3}$$

where x_0, y_0 are the center coordinates of the circle, $d(x,y) = \sqrt{(x-x_0)^2 + (y-y_o)^2} - c$, and ϵ is a parameter defining the width (support) of the membership function.

Basak and Pal [7] extended the FHT to give a quantification of the circle fit for each strong peak in the parameter (accumulator) space. A membership value $h(x, y)$ acts as a weighting factor in the voting process and the vote is discarded when the distance $d(x, y)$ is outside the membership function support. Therefore, the ϵ yielding a sharp peak in the accumulator space one would expect being a correct measurement of the circle fit. These peaks are suggested to be extracted using a pre-defined threshold. After extracting the peaks, the quadratic entropy Q, given by Equation (4), is used to measure the fit. Thus, the sharpness of the peak found in the previous step is measured [7]. By finding a local maximum Q_{max} from a set of ϵ [8] we have a measure of the quality of the circle.

$$Q(\mathbf{g}) = \sum_{(x,y)\in\mathcal{M}} h_{\mathbf{g}}(x, y)(1 - h_{\mathbf{g}}(x, y)) \tag{4}$$

where $\mathbf{g}$ is a vector of scalars a, b, c, and ϵ.

Circle fit quantification is especially interesting when analyzing the quality of printing dots. Unfortunately the computational complexity of the voting is exponentially increasing with the number of parameters [11]. When considering FHT such as in [7] this becomes a problem, since additional parameters arise when pixel intensity and circle center coordinates are also considered as fuzzy sets and parameters ϵ of the sets are included in the search space [7].

C. The Hough Transform Network

Another suitable technique to detect circles is the Hough transform network (HTN) [8]. The technique is based on a two-layer neural network with two input nodes fed with coordinates of edge pixels. The number of output nodes is set equal to the number of circles expected to be found in the input image. The network weights are designed to represent the center coordinates and radius for each of the expected circles and are used to define the activation u_i sent to the ith output node:

$$u_i = [(\mathbf{x} - \mathbf{v}_i)^T \mathbf{A}_i (\mathbf{x} - \mathbf{v}_i)]^{1/2} - c_i \tag{5}$$

where $\mathbf{v}_i$ is a weight vector, c_i is the circle radius, and $\mathbf{A}_i$ defines the distance metric. The output of the ith output node is given by:

$$y_i = \exp\left(-\frac{u_i^2}{\lambda^2}\right) \tag{6}$$

where the parameter λ controls the width of the gaussian—the attraction area. The parameter is crucial for finding a "good" solution. The parameters $\mathbf{v}_i$, c_i, and $\mathbf{A}_i$ are learned in an unsupervised mode by gradient decent using the following error function [8]:

$$E(\mathbf{x}) = \left[\prod_{i=1}^{N}(1 - y_i(\mathbf{x}))\right]^{\alpha} \tag{7}$$

where N is the number circles in the image and $\alpha > 0$ determines the steepness of the error function near minima. The learning process starts with $\mathbf{A}_i$ equal to the identity matrix.

Only the network parameters need to be stored in memory and no accumulator space is required as in the case for ordinary Hough transform. Hence, the requirement of exponentially increasing memory is not the case for HTN. However, the product in the error function (Equation 7) creates a strong dependency among the outputs y_i, i.e. among circle fits. Therefore, input noise may severely affect convergence of the network.

D. Gradient Hough Transform

Instead of using binary images the gradient Hough transform (GHT) operates on grey scale images and uses the gradient field of the input image. Weights of the votes in the GHT are related to the gradient magnitude. A MATLAB implementation of circle Hough transform was used in this work [14].

E. Proposed Technique to Detect Halftone Dots

FHT is a good candidate technique to detect halftone dots. However, the computational burden needs to be reduced, for the technique to be applicable in practice. To achieve this, we exploit the apriori knowledge available from the CTP data—the total number of circles in the input image and their approximate positions.

We start by creating and then thresholding the three-dimensional accumulator space (a, b, c). After the thresholding, a set Z of candidate circles $\mathbf{z}_j = [a_j, b_j, c_j]$ is obtained. Next, points of the three-dimensional accumulator space are projected onto the A, B plane. If the FHT algorithm performed well, we should, on the A, B plane, have as many clusters as there are circles in the image.

To identify the cluster centers (a_j, b_j) we apply the k-means clustering algorithm with k set to the expected number of circles to be found in the image. The cluster centers are initialized with the expected circle centers known from the CTP data.

Next, the radius maximizing the vote $\zeta(a_j, b_j, c_j)$ in the accumulator space needs to found for each cluster. The radius c_q^i for cluster $\mathcal{C}_i$ is given by:

$$c_q^i = \arg\max_{c_j\in\mathcal{C}_i} \zeta(a_j, b_j, c_j) \tag{8}$$

In this way, k circles are identified, whose parameters are then used in Equation (4) to compute the max entropy Q_{max}.

Our experimental tests have shown that at a high image noise level the accuracy of the FHT-based circle detection reduces considerably. To mitigate this drawback, we propose adding a second circle detection phase based on learning. Activation of this second phase is circle dependent. The phase is activated for the jth circle if $Q_{max,j} > \beta$, where β is an experimentally determined parameter. A modified HTN error function given by the following equation can be used to accomplish the learning:

$$E(\mathbf{x}) = \sum_{\mathbf{x}\in\mathcal{N}_i\in\mathcal{C}_i} (1 - y_i(\mathbf{x}))^{\alpha} \tag{9}$$

where $\mathcal{N}_i$ stands for a neighbourhood of the ith halftone dot. The neighbourhood is a circular closed pipe of a given

diameter centered on the expected edge pixels of the halftone dot. In this way, parameters of each circle are fine-tuned using individual training pixels. Other error functions can be used. In this work, we do not present any tests regarding this second circle detection phase.

F. Evaluating the Techniques

Let N_{kno} and N_{det} denote the a priori known and detected number of circles. The number of detected circles matching, with some accuracy, the "true" circles we denote as N_{mat}, $N_{mat} \leq N_{det}$. We use the ratios N_{det}/N_{kno} and N_{mat}/N_{kno} to assess performance of the techniques.

A detected circle is assumed to be matching the true one when the distance between the detected and the true circle centers is equal to or less than 3 pixels. Also the estimated radius must not deviate more than ± 2 pixels.

For each of detected circles N_{det}, the absolute difference between the radii of the detected and nearest known circle is computed along with the Euclidean distance between the center coordinates of the detected and nearest known circle. The average values of the measures, $\overline{|\Delta r|}$ and $\overline{d(\mathbf{v}_{det}, \mathbf{v}_{kno})}$ are also used to assess the techniques.

III. Experimental Investigations

A. Data

Both synthetic images and images taken from real grey bars were used in the tests. An 8-bit 300×300 pixels grey scale synthetic image used is shown in Fig. 5 (*top*). The synthetic image contains 25 filled circles with steep derivatives from background to foreground. The radius of all the circles was set to 12 pixels. This image was subjected to Gaussian noise with the variance ranging from 0.01 to 0.1 in steps of 0.01, Fig. 5 (*bottom*).

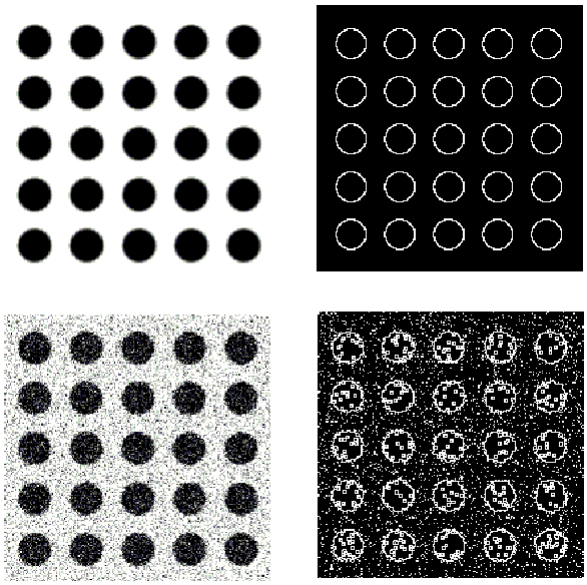

Fig. 5. A synthetic test image (*top*) and the image subjected by Gaussian noise (*bottom*). Detected edge pixels are shown on the right-hand side.

The image was binarized using the Otsu algorithm, which finds the binarization threshold by maximizing the between cluster distance when dividing the pixel intensity distribution into two clusters. The edge pixels were in the next step detected by using morphological filters. Fig. 5 (*right*) presents an example of the "edge image". The grey scale version of the image was used when detecting circles with GHT.

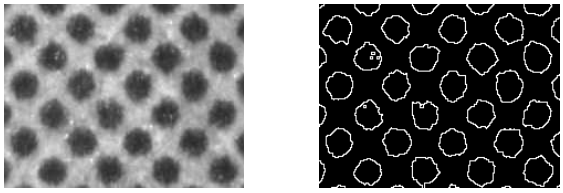

Fig. 6. A region of a real grey bar image (*left*) and edges of the halftone dots (*right*).

A selected region consisting of 24 black dots from a real grey bar image of 215×165 pixels was also used for the tests. The region of the image is shown in Fig. 6 (*left*). The same pre-processing procedures were applied. However, the real grey bar image was not subjected to noise.

Other distortions of the image could be used. For example scaling by a factor greater than one along x or y-axis could be used to simulate effects such as circumferential or lateral slurring. The choice of Gaussian noise in this experiment depends on the following reasons. First, Gaussian noise is appropriate when mimicking the CCD detector noise, i.e. Gaussian noise is added to the grey value of each image pixel. Secondly, the objective is to detect circles in images with various noise levels. Therefore, we distort the already existing perfect circles in the synthetic image by some known noise level. The use of Gaussian noise over the whole input image affects circle edges as much as circle centers. Such deviations put test on both the binarization and Hough Transform steps.

B. Parameters

There are several parameters governing behaviour of the techniques. The appropriate values of the parameters were found experimentally. Thresholds for the accumulator space were experimentally set to 30 for HT and to 80 for FHT and our technique. The gradient-magnitude threshold used by GHT was set to 10. To set constraints on the search space, the radius search was limited from 6 to 17 pixels for all the techniques. When creating the accumulator space of FHT we used $\epsilon = 3$. To find Q_{max}, ϵ was iterated from 0.3 to 10 in steps of 0.3.

The best choice for the HTN parameter λ, governing the attraction area (activation), was $\lambda = 400$. The steepness parameter of the error function α was set to $\alpha = 25$ and the matrix $\mathbf{A}_i$ was equal to the identity matrix. Then network was trained for 200 epochs.

C. Results

First, the sensitivity of the different techniques to noise was studied. Fig. 7 shows the number of detected, N_{det}, and matching, N_{mat}, circles for each technique evaluated on the synthetic test image at a noise level of 0.03, where P1 denotes the proposed technique. As can be seen from Fig. 7, HTN shows low N_{mat} even at this low noise level compared to the other techniques. Observe that N_{det} for HTN and P1 will always be equal to N_{kno}, i.e. equal to 25, because of the prior knowledge incorporated in the algorithms. The ordinary FHT and HT generate very many false detections—large N_{det} values.

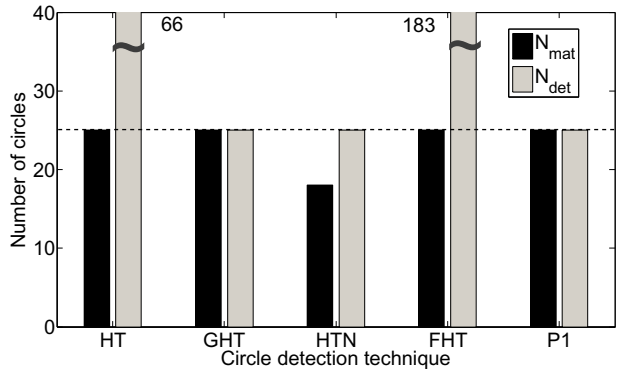

Fig. 7. The number of detected, N_{det}, and matching, N_{mat}, circles for the different techniques evaluated at a noise level of 0.03.

Fig. 8 presents the same parameters at a noise level of 0.09. As can be seen the performance deteriorates for all the techniques. Though HT and FHT maintain the same number of N_{mat}, the number of false detections increases considerably. The proposed technique shows the best performance. Table I summarizes the performance of the techniques assessed using the N_{det}/N_{kno} and N_{mat}/N_{kno} ratios.

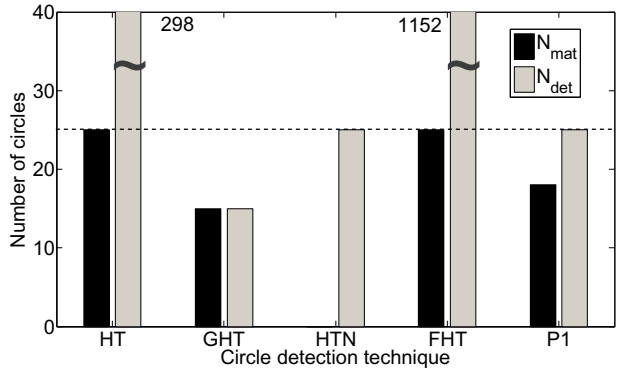

Fig. 8. The number of detected, N_{det}, and matching, N_{mat}, circles for the different techniques evaluated at a noise level of 0.09.

Precision of parameters of detected circles was another aspect used to compare the different techniques. Fig. 9 presents the average absolute difference between the radii of the detected and nearest known circle, $\overline{|\Delta r|}$, for the techniques at different noise levels.

TABLE I

PERFORMANCE OF THE DIFFERENT TECHNIQUES.

Noise	Parameter	HT	GHT	HTN	FHT	P1
0.03	N_{det}/N_{kno}	2.6	1.0	1.0	7.3	1.0
0.03	N_{mat}/N_{kno}	1.0	1.0	0.7	1.0	1.0
0.09	N_{det}/N_{kno}	11.9	0.6	1.0	46.1	1.0
0.09	N_{mat}/N_{kno}	1.0	0.6	0.0	1.0	0.7

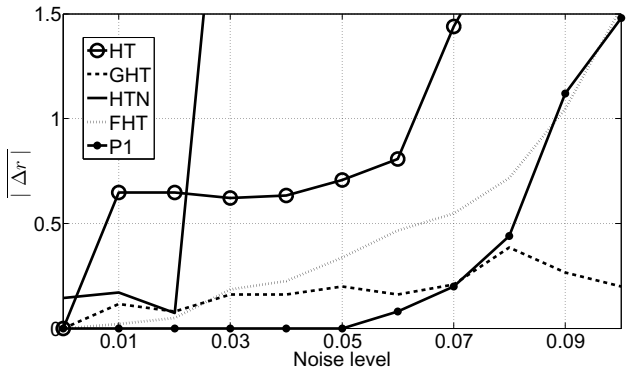

Fig. 9. The average absolute difference between the radii of the detected and nearest known circle, $\overline{|\Delta r|}$, as a function of noise level.

Fig. 7, Fig. 8, and Fig. 9 show that the high number of false detections when using HT and FHT correlates with the lack of precision of circle radius. A very high precision of circle radius was obtained using the proposed technique at low and moderate noise levels. The high deviation of radius for HTN at moderate and high noise levels can be explained by convergence of the algorithm to local minima.

Fig. 10 presents the average Euclidean distance between the center coordinates of the detected and nearest known circle, $\overline{d(\mathbf{v}_{det}, \mathbf{v}_{kno})}$ at different noise levels. Again, the proposed technique was the most accurate.

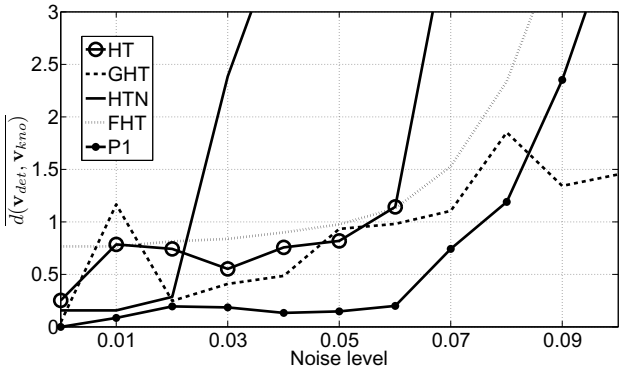

Fig. 10. The average Euclidean distance between the center coordinates of the detected and nearest known circle, $\overline{d(\mathbf{v}_{det}, \mathbf{v}_{kno})}$ as a function of noise level.

Fig. 11 plots the mean width of the membership function (3), $\overline{\epsilon}$, corresponding to Q_{max} as a function of the noise level. A key feature here is the linear relationship ranging from a noise level of 0.00 to 0.06. However, there is a drastic increase of the bandwidth, $\overline{\epsilon}$, required to maximize the quadratic entropy (Q_{max}) when the noise variance exceeds 0.06. At the high noise level circles are heavily distorted, while the initial value of $\epsilon = 3$ is rather low. When choosing a low initial value of ϵ, accumulator peaks are not smooth enough. Hence, the circle centers are estimated with low accuracy, which contributes to the high mean bandwidth, $\overline{\epsilon}$.

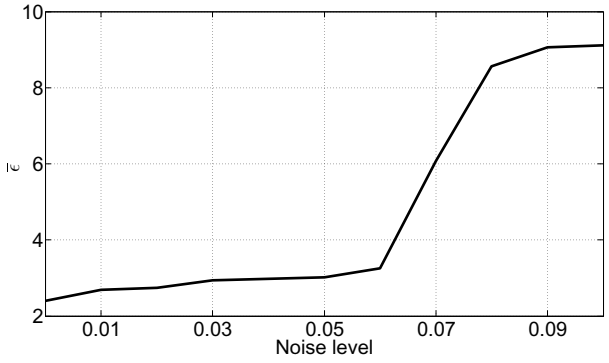

Fig. 11. The mean width of the membership function (3), $\overline{\epsilon}$, corresponding to Q_{max} as a function of noise level.

Fig. 12 shows the mean radius of detected circles as a function of noise level. As can be seen from Fig. 12, the proposed technique provides the most accurate estimate of the radius, which is equal to 12 pixels.

When exploring the performance of GHT and P1 using graphs presented in Fig. 9, 10 and 12 one could see that GHT performs better than P1 in terms of estimating the radius and circle center position when the noise level is equal to and over 0.08. This could partly be explained by the relative low value of ϵ used at these levels of noise. At high noise levels this results in lower accuracy when estimating the cluster center as discussed earlier. Future work may involve adapting ϵ to the level of noise in the image. This may result in an extended linear region of the Q_{max} measure even for noise levels greater than 0.06. The relatively good accuracy of radius and center estimation of GHT at these high noise levels is most likely due to the derivative information that is still present in the image. However less circles are found by GHT at these high noise levels.

Table II summarizes the results obtained when running the techniques on the real image. As in the case of the test image, HT, GHT, and FHT generate a large number of false detections, $N_{det} > N_{mat}$, even though these techniques show accurate values of N_{mat}. All circles detected by the proposed technique matched the true ones. Fig. 13 illustrates the detection results obtained using P1. The value of $\overline{\epsilon} = 2.9$ was computed for the real image, while for the image with the slurred print the value of $\overline{\epsilon} = 8.8$ was obtained.

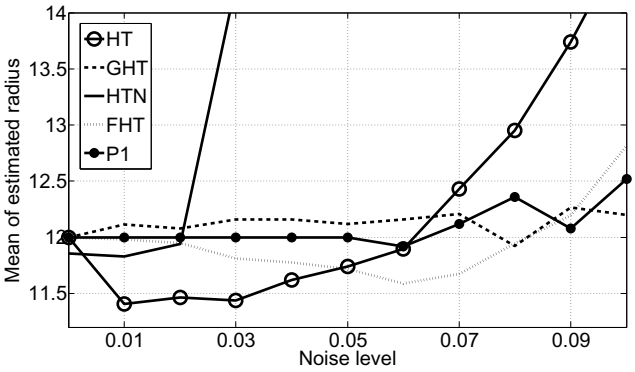

Fig. 12. The mean of estimated radius as a function of noise level.

TABLE II
RESULTS FOR THE REAL IMAGE.

Parameter\Technique	HT	GHT	HTN	FHT	P1
N_{det}	80	62	24	668	24
N_{mat}	24	24	19	24	24
$\overline{R}$	12.8	13.5	14.6	12.8	12.4
$\overline{\epsilon}$					2.9

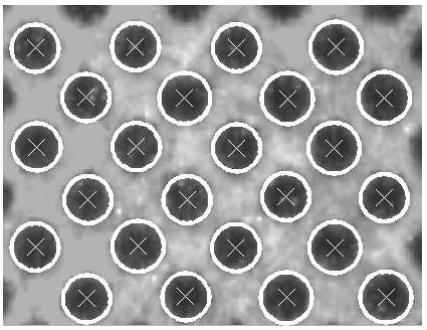

Fig. 13. Circles (white) detected in the real image using P1.

IV. CONCLUSIONS

The fact of linear correlation between ϵ and the noise level corresponds well with earlier findings [7]. This is a desirable correlation when measuring the circle distortion individually for each circle.

The proposed technique outperformed the other techniques in both detection accuracy and precision. The initial selection of ϵ is crucial to get as smooth peaks as possible without loosing circle center accuracy. The proposed clustering step makes the technique much faster if compared to the case of searching the whole accumulator space. FHT without clustering yielded too many false detections.

GHT performed rather well on the synthetic images. However, for the real image, where dots are tightly printed next to each other, many false detections were observed.

HTN performed performed rather well on the real image. However, the sensitivity to noise is too high and the algorithm is too slow to be used on a real grey bar image consisting

of approximately 150 print dots.

REFERENCES

[1] T. D'Orazio, C. Guaragnella, M. Leo and A. Distante, *A new algorithm for ball recognition using circle Hough transform and neural classifier*, Pattern Recognition 37 (2004) 393–408.

[2] A. Verikas, K. Malmqvist and L. Bergman, *Detecting and measuring rings in banknote images*, Engineering Applications of Artificial Intelligence 18 (2005) 363-371.

[3] Y. Sawabe, T. Matsunaga and S. Rokugawa, *Automated detection and classification of lunar craters using multiple approaces*, Advances in Space Research 37 (2006) 21-27.

[4] J.R. Matey, R. Broussard and L. Kennell, *Iris image segmentation and sub-optimal images*, Image and Vision Computing 28 (2010) 215-222.

[5] V. Ayala-Ramirez, C.H. Garcia-Capulin, A. Perez-Garcia and R.E. Sanchez-Yanez, *Circle detection on images using genetic algorithms*, Pattern Recognition Letters 27 (2006) 652-657.

[6] A. Verikas and M. Bacauskiene *Estimating ink density from colour camera RGB values by the local kernel ridge regression*, Engineering Applications of Artificial Intelligence 21 (2008) 35-42.

[7] J. Basak and S. K. Pal, *Theoretical quantification of shape distortion in fuzzy Hough transform*, Fuzzy Sets and Systems 154 (2005) 227-250.

[8] J. Basak and A. Das, *Hough transform network: a class of networks for identifying parametric structures*, Neurocomputing 51 (2003) 125–145.

[9] D. Ioannou, W. Huda and A.F. Laine, *Circle recognition through a 2D Hough Transform and radius histogramming*, Image and Vision Computing, Volume 17, Issue 1, (1999) 15-26.

[10] P.V.C Hough, A. Arbour *Method and means for recognizing complex patterns*, U.S. Patent 3, 3.069.654 (1962).

[11] R.O. Duda, P.E. Hart *Use of the Hough Transformation To Detect Lines and Curves in Pictures*, Communications of the ACM 15 Vol.1 (1972) 11–15.

[12] H. Kipphan, *Handbook of Print Media*, Heidelberg (2001).

[13] J.Ungh, C.M.Nilsson, A.Verikas *Analysis of paper, print and press interaction from online measurements in a press room*, Nordic Pulp and Paper Research Journal Vol 22 no.3 (2007) 383–387.

[14] Detect circles with various radii in grayscale image via Hough Transform *http://www.mathworks.com/matlabcentral/fileexchange/9168*. Accessed at: 2009-10-26.