

LOCAL CORRELATION AND ENTROPY MAPS AS TOOLS FOR DETECTING DEFECTS IN INDUSTRIAL IMAGES

EWA SKUBALSKA-RAFAJŁOWICZ

Institute of Computer Engineering, Control and Robotics
Wrocław University of Technology
ul. Wybrzeże Wyspiańskiego 27, 50–370 Wrocław, Poland
e-mail: ewa.rafajlowicz@pwr.wroc.pl

The aim of this paper is to propose two methods of detecting defects in industrial products by an analysis of gray level images with low contrast between the defects and their background. An additional difficulty is the high nonuniformity of the background in different parts of the same image. The first method is based on correlating subimages with a nondefective reference subimage and searching for pixels with low correlation. To speed up calculations, correlations are replaced by a map of locally computed inner products. The second approach does not require a reference subimage and is based on estimating local entropies and searching for areas with maximum entropy. A nonparametric estimator of local entropy is also proposed, together with its realization as a bank of RBF neural networks. The performance of both methods is illustrated with an industrial image.

Keywords: Defects detection, image processing, local correlation, entropy map.

1. Introduction

Let I be an $I_x \times I_y$ array of gray levels of an image. Although the content and meaning of I can be arbitrary, we confine ourselves to a class of images which have more or less uniform gray levels with possible areas of nonuniform intensities. These regions should be detected and their areas or boundaries should be marked. Industrial images provide many examples and they are mainly considered in this paper. The second class of images with these properties is provided by some classes of medical images, when abnormalities (e.g., cancer cells) of otherwise uniform background should be detected. Major difficulties, common for industrial and medical images, arise from a low contrast between proper areas and nonuniform areas, which are further interpreted as defects. Furthermore, in industrial images we are frequently faced with defects, which have different intensity levels in different parts of the same image, e.g., as a result of different temperatures. This feature precludes or at least highly reduces the possibility of applying simple thresholding for defect detection, as illustrated in the next section. In addition, more advanced thresholding with hysteresis is not applicable, since the result in its first stage strongly depends on a selected threshold. The well-established tech-

niques of edge detection frequently fail for the same reasons as pointed out above. We refer the reader, e.g., to (Faugeras, 1993; Forsyth and Ponce, 2003; Kittler *et al.*, 1985; Otsu, 1979; Pratt, 2001; Ritter and Wilson, 2001), for a detailed description of thresholding and edge detection algorithms.

The aim of this paper is to propose two related techniques of detecting defects, which are based on local correlations and the small area Shannon entropy. Correlation and entropy based techniques are well established in the image processing field see, e.g., (Altmann and Reitbock, 1984; Brink and Pendock, 1996; Gosh-tasby *et al.*, 1984; Pal, 1996; Pratt, 2001; Sezgin and Sankur, 2004; Tsai *et al.*, 2003) as global or semi-global techniques, which are applied to the whole image or to its relatively large parts. Here these techniques are used locally in a moving window. Furthermore, it is proposed to use the correlation technique in a way which is quite opposite to common usage. Namely, we look for areas which have a low correlation with a suitably chosen reference sub-image. Additionally, we propose the structure of a neural network for estimating the Shannon entropy which allows fast FPGA/FPAA hardware implementations (Dong *et al.*, 2006; Maher *et al.*, 2006).

2. Further motivations

To further motivate the need for deriving algorithms in detecting defects in industrial images, consider the image shown in the top panel of Fig. 1. This image presents a piece of hot metal with defects, which are partly visible to an eye.

Trying to find these defects by a well-known thresholding technique, we arrive at the image shown in the middle panel of this figure. This image was obtained using the threshold level 0.315. The number of detected defects is equal to 310, and it is in fact much larger than the true number of defects, which is about one hundred. The reason for the large number of false detections lies in a nonuniform temperature distribution—the boundaries are several degrees colder than the interior. As a result, the upper left and the lower left corners are erroneously marked as defects.

An additional difficulty comes from the oversensitivity of low contrast images to small changes of the threshold. This is illustrated in the bottom image of Fig. 1. This image was obtained by rising the threshold by 0.015. One can observe that false detections occurred not only at the left-hand end of the image but also on its right-hand side. The number of “detected” defects increased to 617.

Summarizing, we need methods which amplify the contrast between defects and their background, independently of its nonuniformity. Let us note that high changes in the background gray levels in different parts of the image preclude the usage of a simple technique based on the extraction of the estimated background from the original image.

An additional requirement is that such methods should not be oversensitive to the choice of a threshold which is used for binarization.

3. Detecting defects using local correlation

Select an $M \times N$ reference subimage which does not contain defects. Denote it by W and its element by w_{mn} , $m = 1, 2, \dots, M$, $n = 1, 2, \dots, N$.

Denote by C_{ij} an $M \times N$, $1 < M \ll I_x$, $1 < N \ll I_y$ subimage of image I with its upper left corner at pixel (i, j) . Gray levels of C_{ij} are denoted by $c_{i+m, j+n}$, $m = 1, 2, \dots, M$, $n = 1, 2, \dots, N$.

The main idea of detecting defects is to calculate the empirical correlation coefficient, denoted as $\rho(W, C_{ij})$, between gray levels in W and in the moving window C_{ij} , i.e.,

$$\rho(W, C_{ij}) = \frac{\sum_{m=1}^M \sum_{n=1}^N \Delta w_{mn} \Delta c_{mn}}{(MN) \hat{\sigma}_w \hat{\sigma}_{ij}}, \quad (1)$$

where $i = 1, 2, \dots, I_x - M$, $j = 1, 2, \dots, I_y - N$,

$$\Delta w_{mn} \stackrel{\text{def}}{=} (w_{mn} - \bar{W}),$$

$$\Delta c_{mn} \stackrel{\text{def}}{=} (c_{i+m, j+n} - \bar{C}_{ij}),$$

while \bar{W} and \bar{C}_{ij} denote respectively the means of gray levels in the reference window W and in the current window C_{ij} , while $\hat{\sigma}_w$ and $\hat{\sigma}_{ij}$ are the corresponding empirical dispersions:

$$\hat{\sigma}_w^2 = (MN)^{-1} \sum_{m=1}^M \sum_{n=1}^N (w_{mn} - \bar{W})^2, \quad (2)$$

$$\hat{\sigma}_{ij}^2 = (MN)^{-1} \sum_{m=1}^M \sum_{n=1}^N (c_{i+m, j+n} - \bar{C}_{ij})^2. \quad (3)$$

One can expect that $|\rho(W, C_{ij})|$ is close to 1 if C_{ij} does not contain defects. Conversely, if a defect is present in the area covered by C_{ij} , then $|\rho(W, C_{ij})|$ is expected to be closer to zero, since W was chosen to represent a nondefective subimage.

Remark 1. In the field of image processing, the formula (1) is usually simplified to speed up calculations. As far as low contrast industrial images are concerned, there is one more reason to simplify this formula. Namely, if W does not contain defects, then $\hat{\sigma}_w$ is close to zero. If, additionally, the current window C_{ij} contains approximately the same gray levels, i.e., it is a nondefective part, then also $\hat{\sigma}_{ij}$ is close to zero and arithmetic overflows are frequently met. For these reasons, we use the inner product between gray levels in W and C_{ij} (see (4)) instead of $\rho(W, C_{ij})$.

Summarizing, the proposed local correlation method runs as follows:

Step 1. Select an $M \times N$ reference subimage $W = [w_{mn}]$ which does not contain defects.

Step 2. For $i = 1, 2, \dots, I_x - M$, $j = 1, 2, \dots, I_y - N$ calculate the inner product, denoted as q_{ij} , between the gray levels in the current window at (i, j) and those in the reference subimage W , i.e.,

$$q_{ij} = \sum_{m=1}^M \sum_{n=1}^N w_{mn} c_{i+m, j+n}. \quad (4)$$

Step 3. Select a threshold $\varrho > 0$. For each pixel decide whether

$$|q_{ij}| < \varrho, \quad (5)$$

and mark pixel (i, j) as defective. Otherwise, mark pixel (i, j) as nondefective.

One can make the selection of ϱ easier by applying the following normalization: Replace each q_{ij} by

$$q_{ij}^* \stackrel{\text{def}}{=} \frac{|\min q_{ij}| + q_{ij}}{|\min q_{ij}| + \max |q_{ij}|}, \quad (6)$$

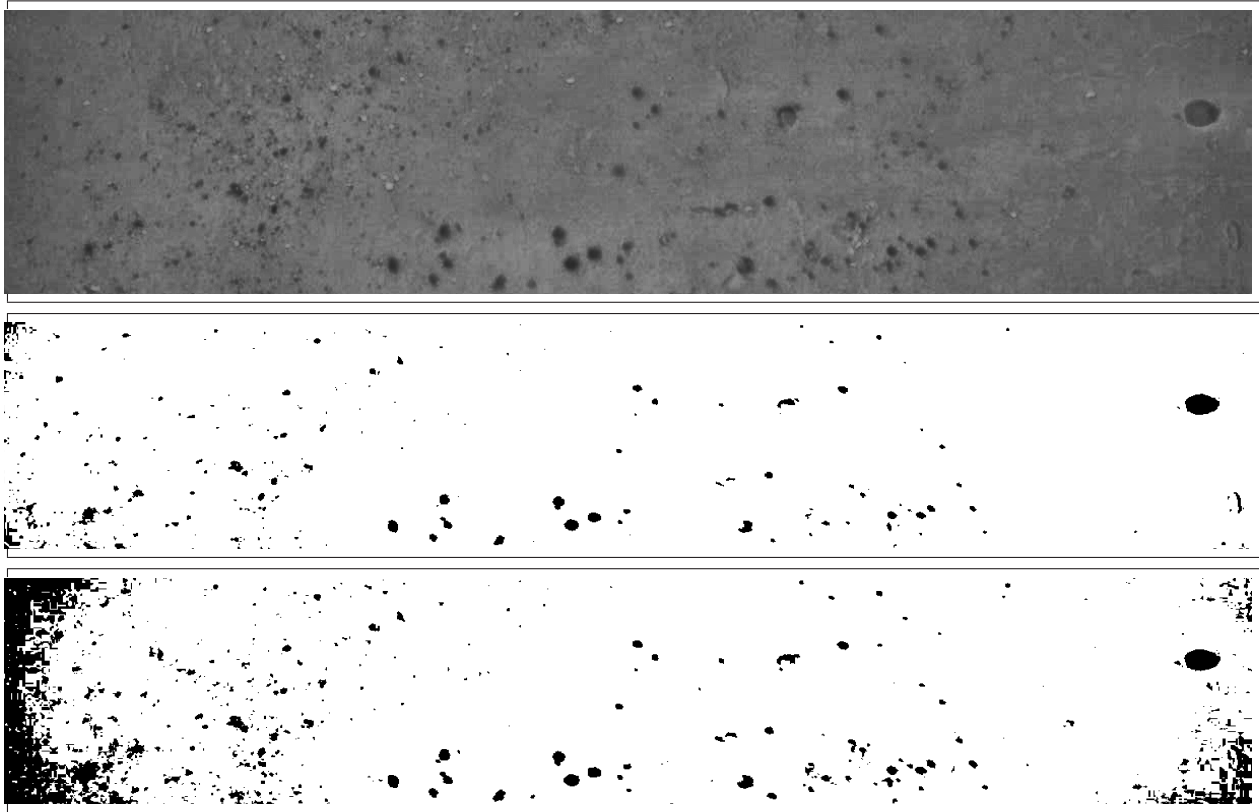


Fig. 1. Hot piece of metal (top panel) binarized with a threshold of 0.315 (middle panel) and using a threshold of 0.33 (bottom panel).

where the max and min operations are taken over all pixels (i, j) in the image. The reason for applying the normalization as in (9) instead of the one which is commonly used in the correlation coefficient is the same as explained in Remark 1.

The performance of the above algorithm when applied to the same industrial image as described in the previous section is shown in Fig. 2. As can be noticed, almost all major defects are detected (see the bottom panel of this figure). Simultaneously, in the middle panel one can observe that the contrast between the background and defects was essentially increased, leading to an easier and more reliable choice of the threshold which serves for detecting defects.

An analysis of (4) and (5) immediately reveals that Steps 2 and 3 of the above algorithm can easily be implemented as a classical feedforward neural network with hard thresholding. Equation (4) can be flattened to one neuron with MN inputs $c_{i+m,j+n}$ and w_{mn} as weights. Alternatively, one can implement (4) as a structure which consists of M neurons with inputs $c_{i+m,j+n}$ and weights w_{mn} , $n = 1, 2, \dots, N$. In both cases, level ϱ is a tunable parameter. Weights w_{mn} have clear interpretation as the gray levels in the reference window, which represents a typical nondefective subimage. Such a subimage can be prepared by selecting and averaging typical examples.

Alternatively, one can apply the well-known techniques of training perceptrons, using examples of nondefective and defective subimages as training samples. In the example described above the former approach was used.

4. Detecting defects by estimating local entropy

The aim of this section is to propose an approach to detect defects by estimating the Shannon entropy of subimages C_{ij} of an $M \times N$ image I , $1 < M \ll I_x$, $1 < N \ll I_y$ with its upper left corner at pixel (i, j) . Denote by $f_{ij}(c) \geq 0$ the one-dimensional probability density function (p.d.f.) of gray levels of the pixels contained in C_{ij} . Gray levels c are usually scaled to intervals $[0, 1]$ or $[0, 255]$. Later on, we shall use the former convention. In this case the Shannon entropy of subimages C_{ij} is defined as

$$H_{ij} = - \int_0^1 \ln(f_{ij}(c)) f_{ij}(c) dc. \quad (7)$$

In the discrete case, where c takes discrete values c_k with probability $p_{ij}(c_k)$, it is given by

$$H_{ij} = - \sum_k \ln(p_{ij}(c_k)) p_{ij}(c_k). \quad (8)$$

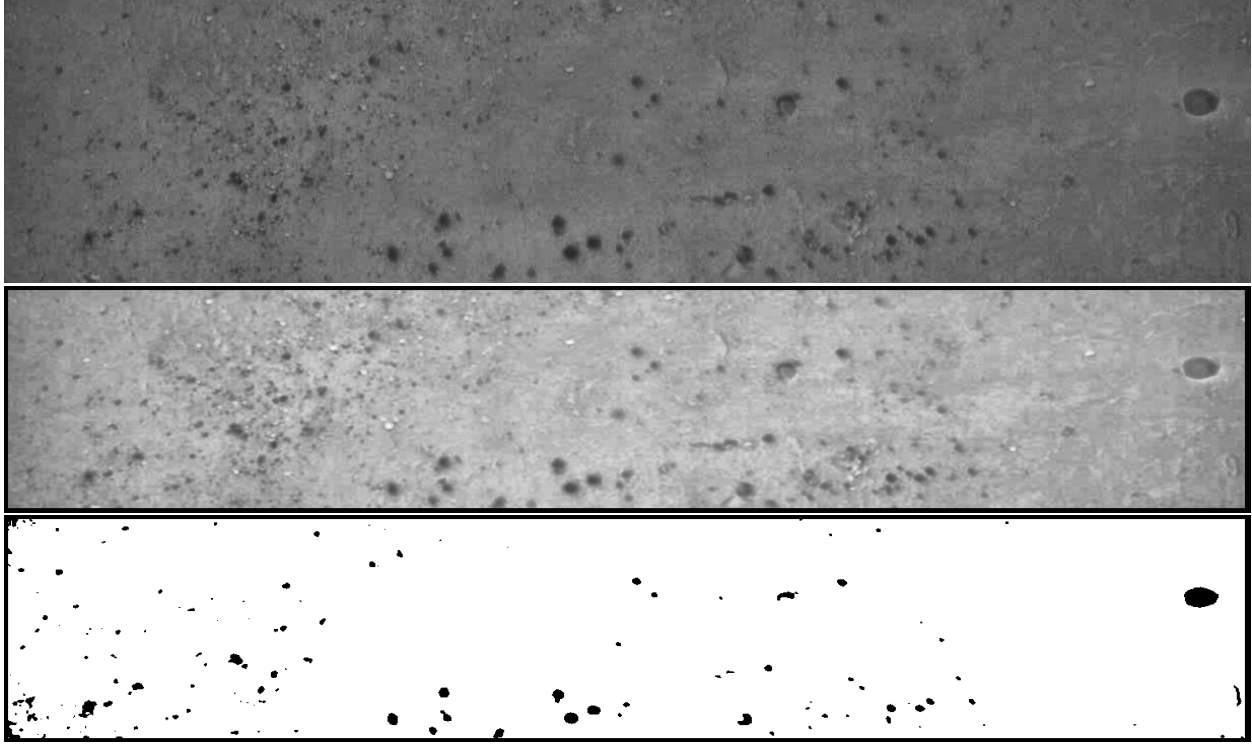


Fig. 2. Performance of the local correlation method: original image (top), normalized inner product map q_{ij}^* (middle), same map binarized below the threshold 0.6 (bottom).

The rationale behind using H_{ij} as indicators of defects is as follows: H_{ij} is large if a variety¹ of gray levels in C_{ij} is present. This case corresponds to defective subimages, since they usually contain almost black gray levels of defects and the brighter part of a nondefective background. Conversely, if only an almost constant background is present, then entropy is relatively small, attaining its minimum if only one gray level is present in C_{ij} . Thus, areas with large entropy are suspected to contain defects. Although entropy is frequently used in image processing, e.g., in texture classification (see (Zhu *et al.*, 1998; Zhu *et al.*, 1997) and the bibliography cited therein), the above usage of entropy as an indicator of defects seems to be new.

Below, it will be more convenient to normalize entropies H_{ij} , $i = 1, 2, \dots, I_x - M$, $j = 1, 2, \dots, I_y - N$ to the interval $[0, 1]$ as follows:

$$H_{ij}^* \stackrel{\text{def}}{=} \frac{|\min H_{ij}| + H_{ij}}{|\min H_{ij}| + \max |H_{ij}|}, \quad (9)$$

and to treat low values of $1 - H_{ij}^*$ as indicators of defects.

Functions $f_{ij}(c)$ are usually unknown and they have to be estimated from observations (gray levels). Below we propose a nonparametric estimator of the Shannon entropy (Berlant *et al.*, 1997; Hero and Michel, 1999; Mokkadem, 1989). The estimator proposed here estimates the

¹ H_{ij} attains its maximum if f_{ij} is the uniform distribution in $[0, 1]$.

Shannon entropy directly and it is well suited for a neural network implementation.

Our starting point is to rewrite (7) as follows:

$$H_{ij} = -\mathbf{E} [\ln f_{ij}(\Gamma_{ij})], \quad (10)$$

where Γ_{ij} is a random variable with the p.d.f. f_{ij} , \mathbf{E} denotes the expectation with respect to this random variable. Also note that we have observations $c_{i+m,j+n}$, $m = 1, 2, \dots, M$, and $n = 1, 2, \dots, N$ of Γ_{ij} at our disposal. The first step toward constructing an estimator \hat{H}_{ij} of H_{ij} is to replace the expectation in (10) by the empirical mean, which leads to

$$-\frac{1}{MN} \sum_{n=1}^N \sum_{m=1}^M \ln f_{ij}(c_{i+m,j+n}). \quad (11)$$

Note, however, that f_{ij} in (11) is still unknown. Our idea is to approximate $\ln f_{ij}(c)$ by a function $s_{ij}(c)$, which is defined as follows:

$$s_{ij}(c) = \sum_{n=1}^N \sum_{m=1}^M \omega_{ij}^{(mn)} K \left(\frac{c - c_{i+m,j+n}}{h_{ij}} \right), \quad (12)$$

where $K(t) \geq 0$ is a kernel function which should be selected so as to fulfill the following requirements:

$$\int_{-\infty}^{\infty} K(t) dt = 1, \quad \int_{-\infty}^{\infty} t K(t) dt = 0. \quad (13)$$

Frequently also the condition $\int_{-\infty}^{\infty} t^2 K(t) dt < \infty$ is imposed on K , especially when the asymptotic behavior is investigated, but this is outside the scope of our paper. Concerning the smoothing parameter $h_{ij} > 0$, its choice is crucial for proper approximation in (12). It should be selected so as $h_{ij} \rightarrow 0$ as $M N \rightarrow \infty$, but in our case of a relatively small M and N , h_{ij} is selected experimentally. In (12), $\omega_{ij}^{(mn)}$ denote tunable weights, which should be selected in a learning process. We shall comment more on this later.

Summarizing, the proposed estimator of entropy has the form

$$\hat{H}_{ij} = -\frac{1}{M N} \sum_{n'=1}^N \sum_{m'=1}^M s_{ij}(c_{i+m',j+n'}), \quad (14)$$

where s_{ij} is defined by (12).

Let us note that (12) can be realized as a radial basis function (RBF) neural net. For each (i, j) we have to calculate $s_{ij}(c)$ for all $c_{i+m',j+n'}$, $m' = 1, 2, \dots, M$, and $n' = 1, 2, \dots, N$. This task can be accomplished using the same RBF net serially. Alternatively, one constructs a bank of RBF nets, each of them calculating one value $s_{ij}(c_{i+m',j+n'})$ in parallel (see Fig. 3). Taking into account that we have to repeat these calculations for each pixel (i, j) , the parallel realization is more appropriate.

It remains to discuss the choice of the weights $\omega_{ij}^{(mn)}$ in (12). Interpreting (12) and (14) as a neural net, we can form the learning sequence of subimages and properly calculated entropies, and select the weights by a learning procedure. The second approach, which we briefly present below, is to find a statistical interpretation of $\omega_{ij}^{(mn)}$ s and to estimate them directly. To this end, let us rewrite (12) and (14) as follows:

$$\hat{H}_{ij} = -h_{ij}^2 \sum_{n=1}^N \sum_{m=1}^M \omega_{ij}^{(mn)} \hat{f}(c_{i+m,j+n}), \quad (15)$$

where \hat{f} is defined as

$$\hat{f}_{ij}(c) = \frac{1}{M N h_{ij}^2} \sum_{n'=1}^N \sum_{m'=1}^M K\left(\frac{c - c_{i+m',j+n'}}{h_{ij}}\right). \quad (16)$$

In $\hat{f}_{ij}(c)$ one can easily recognize the well-known Parzen-Rosenblatt kernel estimator of the p.d.f. f_{ij} . Bearing this fact in mind, one can guess a proper way of interpreting $\omega_{ij}^{(mn)}$ s. Namely, \hat{H}_{ij} in (15) will be directly interpretable as estimator of the entropy if $\omega_{ij}^{(mn)}$ is proportional to $\ln(\hat{f}_{ij}(c_{i+m,j+n}))$. More precisely,

$$\omega_{ij}^{(mn)} = \frac{1}{M N h_{ij}^2} \ln(\hat{f}_{ij}(c_{i+m,j+n})). \quad (17)$$

Substituting this equality into (15), we obtain

$$\hat{H}_{ij} = -\frac{1}{M N} \sum_{n=1}^N \sum_{m=1}^M \ln(\hat{f}(c_{i+m,j+n})) \times \hat{f}(c_{i+m,j+n}). \quad (18)$$

Clearly, it is possible to obtain the same estimator directly, just by substituting the Parzen-Rosenblatt kernel density estimator into (7), but then the above interpretation of estimating entropy by RBF nets would not be possible.

Selecting K as the uniform kernel in the above formulas, we obtain the well-known histogram estimator of entropy, which is also used in the simulations presented below. Note, however, that if we have high resolution images, then we can choose larger subimages for estimating entropy. In such a case one can obtain more precise estimates of entropy selecting $K(\cdot)$ as the Gaussian or other kernel, which estimates f better than the histogram estimator.

It remains to discuss the choice of the smoothing parameter h_{ij} in (12) and (16). In the simulations reported below, $h_{ij} = 1$ was used for the gray scale $[0, 255]$. In general, the choice of the smoothing parameter seems to be less critical than in nonparametric density estimation, due to the averaging operation in (18). Cross-validation techniques of selecting h_{ij} are not recommended here, since they are too time-consuming in image processing. As a rule of thumb, we propose to select h_{ij} sufficiently large so as to avoid intervals in which $\hat{f}(c)$ is so close to zero that numerical difficulties in calculating the logarithmic function could arise.

In Fig. 3 the local entropy map (middle panel) of the industrial image (top panel) is shown. The map was obtained using the estimator (18) with the rectangular kernel and the window size 3×3 . As one can notice, the areas containing defects are strongly contrasted with clean areas, making the choice of the threshold for binarization easier and more reliable. The binarized image is shown in the bottom panel of this figure. Comparing this image with the lower panel in Fig. 2, we conclude that the binarized entropy map reveals almost the same defects, but there is an important difference between these two approaches. Namely, the binarized entropy map reveals boundaries of defected areas, while the binarized correlation map marks all the areas containing defects. Which approach of the two is better depends on the application at hand. At the expense of an additional computational effort, e.g., by using morphological operators, one can force the binarized entropy map to cover all the areas of defects.

5. Concluding remarks

The local maps of inner products and entropy were investigated as tools for detecting defects of products from industrial images. Both techniques provide comparable de-

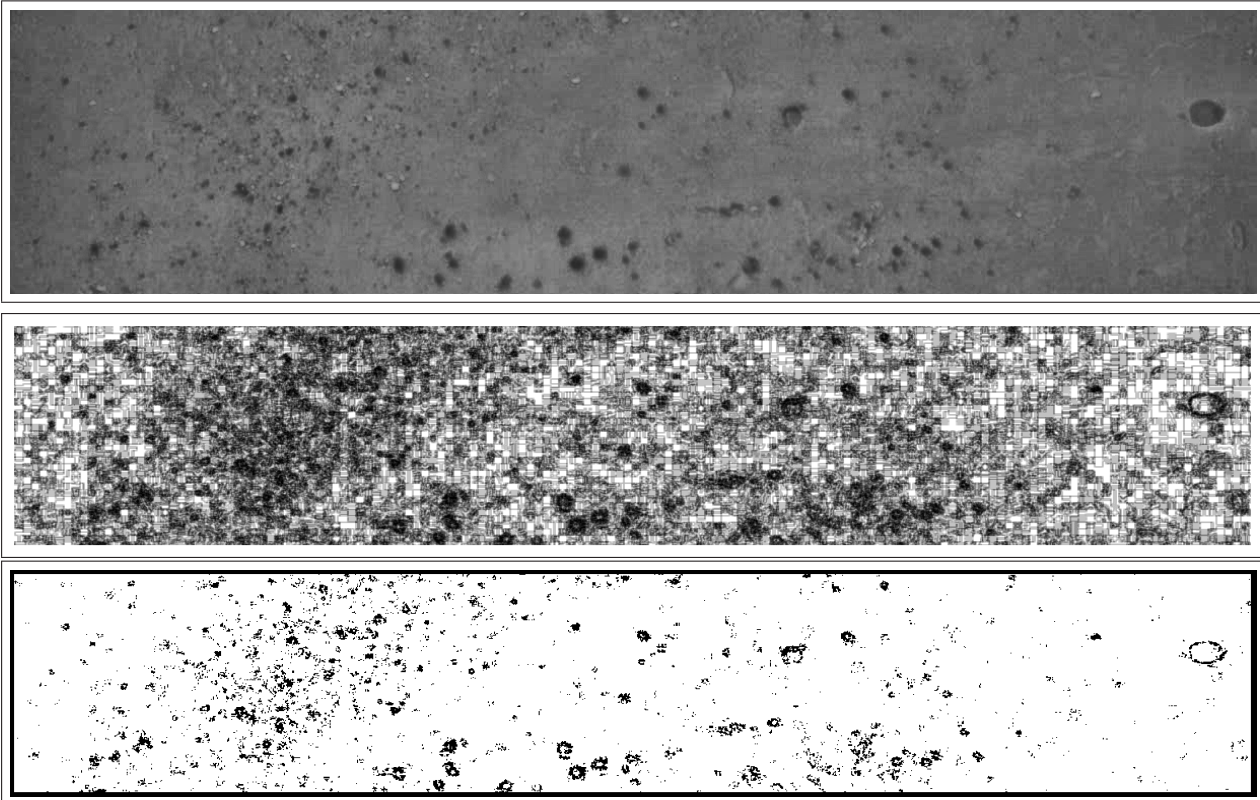


Fig. 3. Original image (top), local entropy map (middle), binarized entropy map (bottom, threshold at 0.15).

tection abilities and provide approximately the same number of detected defects (about one hundred in our example). The choice between them should take into account the following factors:

1. The technique based on inner products is computationally less demanding. It is about four times faster than the technique based on estimating local entropy. Also its neural network realization is extremely simple. On the other hand, the inner product technique requires that we have a good example of a pure (non-defective) subimage or a sequence of such images when different parts of a nondefective image have different gray levels, e.g., as a result of different temperatures.
2. The technique based on estimating local entropy is computationally more demanding and its neural network realization is more complicated. Its advantage is the fact that a reference subimage is not necessary.

Acknowledgements

This work was sponsored by a grant of the Polish Ministry of Science and Higher Education for the years 2006–2009. The acquisition and initial preparation of the industrial

images which were used for the simulation experiments were sponsored by a grant from the Foundation for Polish Science.

References

- Altmann J. and Reitbock H.J.P. (1984). A fast correlation method for scale and translation invariant pattern recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **6**(1): 46–57.
- Beirlant J., Dudewicz E., Györfi L., van der Meulen EC (1997). Nonparametric entropy estimation: An overview, *International Journal of Mathematical and Statistical Sciences* **6**(1):17–39.
- Bishop C.M. 1995. *Neural Networks for Pattern Recognition*, Oxford Press.
- Brink A. D. and Pendock N. E. (1996). Minimum cross-entropy threshold selection, *Pattern Recognition* **29**(1): 179–188.
- Dong P., Bilbro G.L. and Mo-Yuen Chow (2006). Implementation of artificial neural network for real time applications using field programmable analog arrays, *Proceedings of the International Joint Conference on Neural Networks, Vancouver, BC, Canada*, pp. 1518–1524.
- Faugeras O. (1993). *Three-Dimensional Computer Vision*, MIT Press, Cambridge.
- Forsyth D.A. and Ponce J. (2003). *Computer Vision: Modern Approach*, Prentice Hall, Upper Saddle River, NJ.

- Goshtasby A., Gage S. H. and Bartolic J. F. (1984). A two-stage cross-correlation approach to template matching, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **6**(3): 374–378.
- Haykin S. (1999). *Neural Networks. A Comprehensive Foundation, 2nd Ed.* Prentice Hall, Upper Saddle River, NJ.
- Hero A.O. and Michel O.J.J. (1999). Asymptotic theory of greedy approximations to minimal-point random graphs, *IEEE Transactions on Information Theory* **45**(6): 1921–1938.
- Kittler J., Illingworth J. and Foglein J. (1985). Threshold selection based on a simple image statistic, *Computer Vision, Graphics, and Image Processing* **30**(2): pp. 125–147.
- Maher J., Mc Ginley B., Rocke P. and Morgan F. (2006). Intrinsic hardware evolution of neural networks in reconfigurable analogue and digital devices, *Proceedings of 14th Annual Symposium on Field-Programmable Custom Computing Machines FCCM' 06*, Napa, USA, pp. 321–322.
- Mokkadem A. (1989). Estimation of the entropy and information of absolutely continuous random variables, *IEEE Transactions on Information Theory* **35**(1): 193–196.
- Otsu N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man and Cybernetics* **9**(1): 62–66.
- Pal N.R. (1996). On minimum cross-entropy thresholding, *Pattern Recognition* **29**(4): 575–580.
- Pratt W.K. (2001). *Digital Image Processing: PIKS Inside, 3rd Ed.*, John Wiley and Sons, New York.
- Ritter G.X. and Wilson J.N. (2001). *Handbook of Computer Vision Algorithms in Image Algebra, 2nd Ed.*, CRC Press, Boca Raton, FL.
- Sezgin M. and Sankur B. (2004). Survey over image thresholding techniques and quantitative performance evaluation, *Journal of Electronic Imaging* **13**(1): 146–168.
- Tsai D., Lin Ch., and Chen J. (2003). The evolution of normalized cross correlation for defect detection, *Pattern Recognition Letters* **24**(15): 2525–2535.
- Zhu S.C., Wu Y. and Mumford D. (1997). Minimax entropy principle and its application to texture modeling, *Neural Computation* **9**(8): 1627–1660.
- Zhu S.C., Wu Y. Mumford D. (1998). Filters, random fields and maximum entropy (FRAME): Towards a unified theory for texture modeling, *International Journal of Computer Vision* **27**(2): 107–126 .