Bull. Nov. Comp. Center, Comp. Science, 36 (2014), 89–96

Defect detecting in multi-element photodetector by wavelets and neural networks

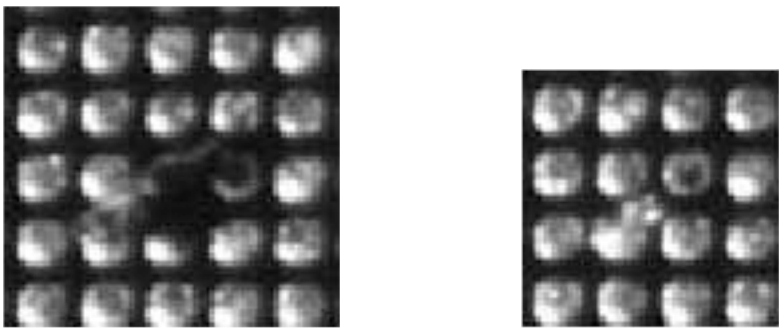
M.S. Tarkov, V.G. Polovinkin, A.S. Bystrov

Abstract. We consider the application of wavelet transform and neural networks to solving the problem of defect detection (the lack of elements and the presence of adhesions elements) in multi-element photodetectors by processing their images. It is shown that both methods can be successfully applied to the detection of defects. Found that a method based on wavelet transform requires the manual selection of parameters depending on the size of the processed image. Due to the ability of neural networks to learn, a method for the search for defects with neural networks, automatically adapts to the processed image.

1. Introduction

Automation of the quality control in mass production and conveyor production is an important area of application of methods of image analysis systems and neurocomputing [1–5]. This paper is aimed at the design and software implementation of wavelet and neural network algorithms for detecting defects in images of factory products. The current practical problem is the defect detection on images of photodetector arrays produced by the Institute of Semiconductor Physics, SB RAS, and used in the thermal imaging device [6]. The matrix photodetector is a semiconductor chip with the size of about 1 square centimeter containing an array of 128×128 similar photosensitive elements.

The matrix image can be viewed as a two-dimensional signal having a regularly repeating component. Defects in this case are violations of the



a b

Figure 1. Defect examples: a) a hole, b) the adhesion of adjacent elements

signal regularity. On the images of the photodetector we can detect defects of the two types: the absence of an element (a hole) (Figure 1a) and the adhesion of the adjacent elements (Figure 1b).

To isolate the singular points of the signal, a discrete wavelet transform [2] can be used due to an accurate representation of local features of the signal, which are absent, for example, in a Fourier series.

2. The wavelet transform

The multiscale wavelet transform methods create an image representation, in which there are both: a spatial and a frequency information. The wavelet transform is well suited to describe local inhomogeneities in the texture image. In the one-dimensional case, the signal $f(x) \in L_2(R)$, having the finite energy, can be decomposed to a wavelet series corresponding to the scale function $\varphi(x)$ and the wavelet function $\psi(x)$:

$$f(x) = \sum_k c_{j_0}(k)\varphi_{j_0,k}(x) + \sum_{j=j_0}^{\infty}\sum_k d_j(k)\psi_{j,k}(x),$$

where $\varphi_{j_0,k}(x) = 2^{j_0/2}\varphi(2^{j_0}x - k)$, $\psi_{j,k}(x) = 2^{j/2}\psi(2^j x - k)$, $\varphi(x)$ is a basic scale function, $\psi(x)$ is a basic wavelet function, $c_{j_0}(k)$ are approximation coefficients, $d_j(k)$ are wavelet coefficients:

$$c_{j_0}(k) = \int f(x)\varphi_{j_0,k}(x)\,dx, \qquad d_j(k) = \int f(x)\psi_{j,k}(x)\,dx. \tag{1}$$

If the decomposed function $f(x)$ is a sequence of numbers, then the result sequence is called a discrete wavelet transform (DWT) of the function $f(x)$. In this case, transformation (1) is converted to a DWT-pair:

$$W_\varphi(j_0,k) = \frac{1}{\sqrt{M}}\sum_{x=0}^{M-1} f(x)\varphi_{j_0,k}(x), \quad W_\psi(j,k) = \frac{1}{\sqrt{M}}\sum_{x=0}^{M-1} f(x)\psi_{j,k}(x).$$

The fast wavelet transform (FWT) uses the correlation of the DWT-coefficients of the neighboring scales:

$$W_\varphi(j,k) = \sum_m h_\varphi(m-2k)W_\varphi(j+1,m),$$
$$W_\psi(j,k) = \sum_m h_\psi(m-2k)W_\psi(j+1,m),$$

where h_φ and h_ψ are weight functions in representing the functions $\varphi_(j,k)$ and $\psi_(j,k)$, respectively, as sum of functions $\varphi_(j+1,k)$ (refinement coefficients).

The wavelet decomposition of the image $f(x, y)$ is based on the one-dimensional scale and wavelet functions, respectively, obtained from the basic scale function $\varphi(x)$ and the basic wavelet $\psi(x)$:

$$\begin{aligned}\varphi(x,y) &= \varphi(x)\varphi(y) \quad \text{is a two-dimensional scale function,}\\ \psi^H(x,y) &= \psi(x)\varphi(y) \quad \text{is a horizontal wavelet,}\\ \psi^V(x,y) &= \varphi(x)\psi(y) \quad \text{is a vertical wavelet,}\\ \psi^D(x,y) &= \psi(x)\psi(y) \quad \text{is a diagonal wavelet.}\end{aligned}$$

If we are given a two-dimensional separable scale function and a wavelet function, we can define a family of basis functions using the operations of shifts and changes in the scale:

$$\begin{aligned}&\varphi_{j,m,n}(x,y) = 2^{j/2}\varphi(2^j x - m, 2^j y - n),\\ &\psi^i_{j,m,n}(x,y) = 2^{j/2}\psi^i(2^j x - m, 2^j y - n), \quad i \in \{H, V, D\}.\end{aligned}$$

Then we can determine the discrete wavelet transform of the image $f(x, y)$ with $M \times N$ size:

$$\begin{aligned}W_\varphi(j_0, m, n) &= \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)\varphi_{j_0,m,n}(x,y),\\ W^i_\psi(j_0, m, n) &= \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)\psi^i_{j_0,m,n}(x,y), \quad i \in \{H, V, D\}.\end{aligned}$$

Here the coefficients $W_\varphi(j_0, m, n)$ determine an approximation of the function $f(x, y)$ in a scale j_0. The coefficients $W^i_\psi(j_0, m, n)$ determine horizontal, vertical, and diagonal details for the scales $j \geq j_0$.

The image $f(x, y)$ can be restored with the coefficients W_φ and W^i_ψ by the inverse wavelet transform:

$$\begin{aligned}f(x,y) = &\frac{1}{\sqrt{MN}} \sum_m \sum_n W_\varphi(j_0, m, n)\varphi_{j_0,m,n}(x,y) +\\ &\frac{1}{\sqrt{MN}} \sum_{i=H,V,D} \sum_{j=j_0}^{+\infty} \sum_m \sum_n W^i_\psi(j, m, n)\psi^i_{j,m,n}(x,y).\end{aligned}$$

Since the scale function and the wavelet function are separable, first the one-dimensional fast wavelet transform by rows of the function $f(x, y)$ is evaluated with decimation of the transform result, and then one-dimensional transformation by columns is implemented with decimation of the result. As a result we have the four parts W_φ, W^H_ψ, W^V_ψ and W^D_ψ of the original image, each part twice the linear dimensions smaller than the original image: an approximation image W_φ and three images W^H_ψ, W^V_ψ and W^D_ψ of details, with horizontal, vertical and diagonal orientation, respectively (Figure 2).

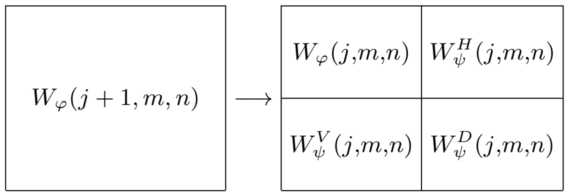


Figure 2. The image decomposition obtained by a step of the fast wavelet transform

3. Defect detection using the wavelet transform

The wavelet transform is well suited to describe the local inhomogeneities in the texture image. Choosing an approximate picture and vanishing image details, we remove regular repetitive textural elements from the reconstructed image and thereby strengthen local anomalies. The image of Figure 3b is obtained from the image of Figure 3a by two steps of forward wavelet transform using the Daubechies wavelets [2] of eighth order followed by zeroing the detail coefficients and inverse wavelet transform. Then thresholding can be used (Figures 3c, 3d) to separate defective and homogeneous areas in the reconstructed image. The threshold transformation has the form

$$f_b(x,y) = \begin{cases} 1, & f(x,y) \geq T, \\ 0, & f(x,y) < T, \end{cases}$$

where $f(x,y)$ and $f_b(x,y)$ are the original and the resulting binary image, respectively. The threshold value $T \in (0,255)$ is chosen experimentally. For

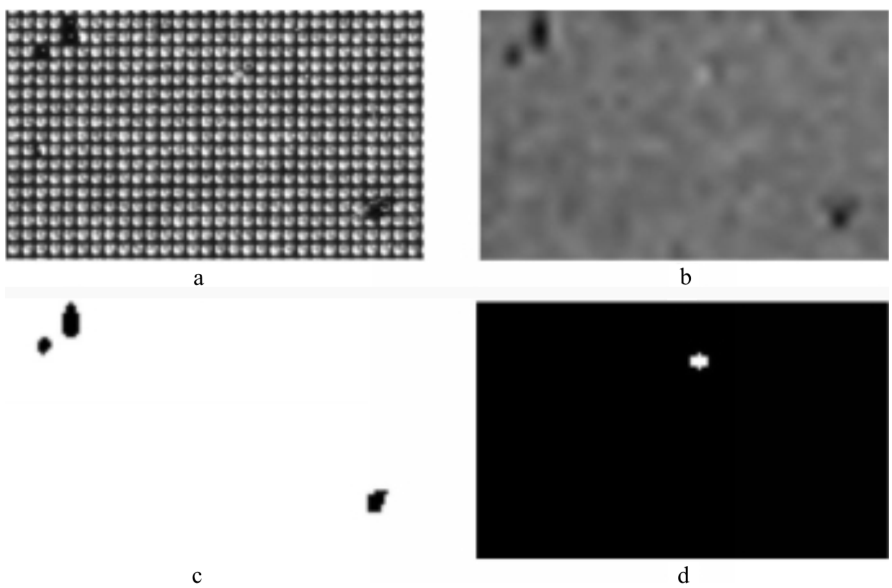


Figure 3. Detection of defects on the phororeceiver matrix

$T = 70$ (see Figure 3c) the "holes" are allocated on the image in the form of black spots, and for $t = 150$ (see Figure 3d), adhesions are allocated as white spots. This approach converts a complex task of detecting defects on complex texture images to a simple task of thresholding untextured images.

In order to have a good result of the defect detection method using the wavelet transform, the following is necessary:

1. State the linear dimensions for the image to a power of two (in the experiments image processing made with linear dimensions equal to 512);
2. Choose the number of steps of the wavelet transform; and
3. Pick thresholds to highlight defects.

Note that the number of transformation steps and thresholds depend on the size of an image. These reasons make the above method be not always appropriate for finding defects in an image. An alternative is to use neural networks [4, 5] as they are suitable for processing images of arbitrary sizes.

4. The neural network

The artificial neural network is a system of interconnected simple processors — neurons. Each neuron receives the input signals x_i, $i = 1, \dots, N$, and generates the output signal $y = f(u)$, where $f(u)$ is a nonlinear activation function, $u = \sum_{i=0}^{N} w_i x_i$ is a neuron activation, w_i, $i = 0, 1, \dots, N$, are the neuron weight coefficients, w_0 is a threshold value, and $x_0 = 1$ [4].

Training the neuron consists in choosing the weight coefficients w_i such that the output signal y should coincide with the reference value d. The supervised training used a set of training samples, i.e., a set of pairs of the type (x, d), where x is a vector of input signals.

For the unipolar sigmoid neuron, the activation function is given by the expression

$$f(u) = \frac{1}{1 + \exp(-\beta u)},$$

where β is the activation function parameter. In a multi-layer network, the first layer of neurons receive input signals and converts them to passing to the second layer of neurons. Next, the second layer is activated, etc. Finally, the last layer generates output signals of the neural network.

Due to the differentiability of the activation function in the training, we can use the gradient optimization methods. In particular, we use the steepest descent method (the error back-propagation method) [4], according to which the weight vector refinement is performed in the direction of the negative gradient of the object function $E(w) = (y - d)^2/2$, i.e.,

$$w(t+1) = w(t) - \alpha \nabla E(w(t)),$$

where $\alpha \in (0,1]$ is a training step.

Application of the gradient method for training the neuron network only ensures the achievement of a local minimum. To exit from the vicinity of a local minimum, the training with the moment can be productive. In this method, the process of refining the weights is determined not only with the information on the gradient of the function, but also with a previous change in the weights. Such a method can be expressed as

$$w(t+1) = w(t) - \alpha \nabla E(w(t)) + \gamma \Delta w(t), \quad \gamma \in (0,1),$$

where the first term corresponds to the usual method of steepest descent, while the second term, called the moment, reflects the latest update weights and does not depend on the actual value of the gradient.

5. The defect detection using neural networks

Let us consider features of the neural network used in this paper. The input of the neural network is generated based on the neighborhood of a point to be tested. The training of a neural network is based on the reference images of defective areas (defect maps). The neural network outputs are processed to determine the threshold at which we can clearly judge of the defect.

There are the following stages of the neural network algorithm for detecting defects:

1. Creating the feature vectors based on training a digital image;
2. Submitting feature vectors to the neural network; and
3. Smoothing the output of the neural network by the median filter and making thresholding for a map of defects.

Let us consider every stage and separately discuss the method of neural network training. The feature vectors are obtained from the neighborhoods of pixels converted to the grayscale. As a mask by which the neighborhood is selected, we use a square window: the central pixel is characterized by its environment [5]. The window size should be large enough to capture the brightness variation of points, but also sufficiently small to describe the texture. Too large windows increase the computation time. The feature vector is constructed as concatenation of pixels in the window.

We use the double-layered feed-forward network with a single output neuron. The number of neurons in the hidden layer (8 neurons) is chosen experimentally based on the fact that an insufficient number of them will not achieve an acceptable error of training, and with an increase in their number above the optimal value ceases improving the network performance.

The training set of feature vectors is composed of images of the photomatrix with defects (see Figure 1a). For all the neurons we use a unipolar sigmoidal activation function. As a reference for training, we use a map of defects (Figure 4) in which defective areas are shown in black (the value is 0) and the area without defects is shown in white (the value is 1).

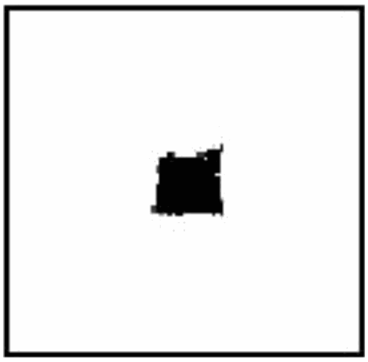


Figure 4. A map of defects

When training and image processing are done by the neural network, the original image is subject to the median filtering with a window of the size of 5×5 pixels and contrast stretching by the expression [3]:

$$s = \frac{1}{1 + (m/r)^E},$$

where $m = 50$ is the threshold of contrasting, r is the intensity of the input pixel, s is the intensity of the corresponding output pixel, and a parameter $E = 10$ controls the slope of the transformation.

The result of the image processing by the neural network was also handled by the median filter with a window of pixels to remove the noise which is not a defect of the photodetector array. To make the defects on the resulting image more visible, the erosion operation with a circular structural element of a radius of 3 pixels is applied.

6. Conclusion

The application of the wavelet transform and the neural networks to solving the problem of defect detection (the lack of elements and the presence of adhesion elements) of multi-element photodetectors by processing their images is considered. It is shown that both methods can be successfully applied to the detection of defects. It is found that a method based on the wavelet transform requires the manual selection of parameters depending on the size of the processed image. Due to the ability of neural networks to learn, a method for detecting defects using the neural networks, automatically adapts to the processed image. Based on the use of the wavelet transform and the neural networks, an automated system for detecting defects of matrix photodetectors can be created. An experimental sample of such a system is implemented in Visual Studio C++ using the image processing library OpenCV [7].

References

[1] Tsai D.-M., Hsiao B. Automatic surface inspection using wavelet reconstruction // Pattern Recognition. — 2001. — Vol. 34. — P. 1285–1305.

[2] Gonzalez R.C., Woods R.E. Digital Image Processing. — Prentice Hall, 2002.

[3] Gonzalez R.C., Woods R.E., Eddins S.L. Digital Image Processing using MATLAB. — Prentice Hall, 2004.

[4] Haykin S. Neural Networks. A Comprehensive Foundation. — Prentice Hall, 1999.

[5] Castilho H.P., Pinto J.R.C., Serafim A.L. NN Automated Defect Detection Based on Optimized Thresholding // Image Analysis and Recognition. — 2006. — P. 790–801. — (LNCS; 4142).

[6] Kuryshev G.L., Kovchavtsev A.P., Vainer B.G., et al. Medical infrared imaging system based on a 128×128 focal plane array for 2.8–3.05 pm spectral range // Optoelectronics, Instrumentation and Data Processing. — 1998. — P. 5.

[7] Bradski G., Kaehler A. Learning OpenCV. — O'Reilly Media, Inc., USA, 2008.