PCB Image Defects Detection by Artificial Neural Networks and Resistance Analysis

ROMAN MELNYK, VITALII VOROBII
Software Department,
Lviv Polytechnic National University,
12 Stepan Bandera St., Lviv, 79013,
UKRAINE

Abstract: - The approach contains the sequence of algorithms and formulas for image processing. They are single-layer neural networks, thinning, clustering, mathematical image comparison, and measurements of the trace length and width. All these procedures solve the task of selection and separation of the main objects in the printed circuit board: contacts, traces, and defects. The calculated features connect the conductance resistance of traces with the characteristics of defects. Imposing a tolerance on the distributed or concentrated changes of resistance it is possible to mark the defective and suspicious printed circuit boards.

Key Words: - Printed circuit board, chain, trace, defects, clustering, flood-filling, artificial neural networks, skeleton, edge, thinning, cumulative histogram, resistance, width of trace, overlay, tolerance.

Received: August 25, 2023. Revised: February 19, 2024. Accepted: March 16, 2024. Published: May 23, 2024.

1 Introduction

There are many special devices based on lasers, ultrasound, and X-ray radiation which are used to detect defects in printed circuit boards. Their purpose is to measure the thickness, the amount of solder, the quality of the layers of the components, as well as the shape of the drawing. The collected data are input for image recognition algorithms in the context of post-production defect detection, [1], [2], [3], [4]. Such devices and algorithms are implemented in powerful complex control systems to prevent damage to manufactured printed circuit boards and surfaces, [5], [6].

Many works consider the visual presentation of printed circuit boards. They contain traditional image processing algorithms including subtraction algorithms, wavelet, and a table of connection traces based on contacts from the etalon image, [7], [8], [9]. In the last time, for inspection of inaccuracies in PCB production, neural networks were used, and the approaches were described in articles, [10], [11], [12], [13], [14], [15]. Three surveys contain many references to publications describing approaches and methods for the detection and classification of defects in printed circuit boards, [16], [17]. [18].

Short, open circuits and depression changes in computer vision and artificial intelligence inspection systems are subjects in the publications, [19], [20], [21], [22]. The subtraction operations between PCB images used in many works are sensible to a deviation of the sizes of traces and contacts from the etalon. They give extra or lacking pixels that do not affect the correct operation of the electrical scheme. Defects in traces and contacts of such types as shift and extra metal are not analyzed.

All mentioned approaches differ between themselves by complexity, input data, and characteristics of their implementation.

In this work, the following approaches to solving the given problem are applied: mathematical comparison formulas, a filling algorithm for dividing and selecting circuits, artificial neural networks for the classification of printed circuit board images, and formulas for calculating the resistance characteristics of traces and contacts.

The advantages of the proposed method are the preliminary division by artificial neural networks of images of printed circuit boards into two classes, followed by the detection of three classes of defects in the defective class: connections, excess, and lack of metal on traces and contacts. The influence of these defects on the conduction resistance is calculated. Areas with detected defects are marked on traces and contacts. This creates the possibility of additional analysis of the circuits for repairability or failure.

2 Separation of Chains in PCB Image

2.1 Determination of chain coordinates

Each printed circuit board has a geometry such that all pins assigned to one circuit are connected, and two or more pins assigned to different circuits are not connected. As a result, they are characterized by a planar topology of wires or traces. Automatic access to each chain is possible only if the coordinates of the pixels in the traces are known and with their help, one chain can be selected and separated from the entire set of black traces on the board. This can be done by changing the color of the selected chain.

The Hilditch thinning algorithm, [8], works with binary (black and white) images and is reliable and easy to program. It is modified to find the pixel coordinates of specific points, which are taken as the output pixels for the fill algorithms, which select and separate circuits in PCB images by painting them with specified colors.

Figure 1 shows a skeleton image created for a template PCB. It has two types of specific points: switches (circled in blue) and terminations (circled in red). Specific points on the image next to the coordinates are not identified, that is, they are not assigned the identification number of the chain that contains them.

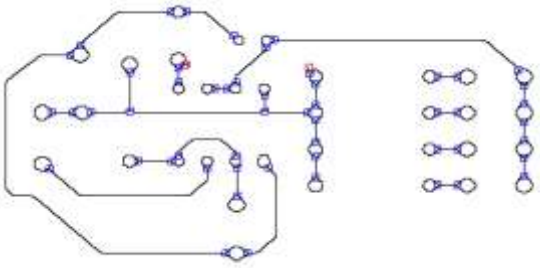


Fig. 1: Skeleton with endings and switches

The specific points are united in the set P. All elements in the set denote only the coordinates of their location. Taking elements from the set as starting points the flood-fill algorithm fills chains with different colors. The filling algorithm is applied as many times as there are chains in the scheme.

For illustration, the skeleton is overlaid with the template PCB image. The two chains are filled with green and blue and are shown in Figure 2.

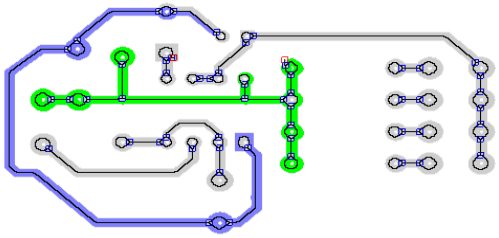


Fig. 2: Overlay of PCB image with skeleton, and filled chains (green and blue)

Each color gives the identification number for the chain and its specific points. This allows to split the set P into subsets of specific points connected with the concrete chain:

$$P(p_1, p_2,\ldots,p_N)=P(P_1,P_2,\ldots,P_n),$$

where $p_1, p_2,\ldots,p_N$ are coordinates of all specific points, $P_1, P_2,\ldots, P_n$ are sets of points assigned to the chains with the sequence numbers 1, 2, …,n.

In the extended version they are as follows:

$$P_1=\{(x_{11},y_{11}, c_{11}), (x_{12},y_{12}, c_{12}), (x_{1n(1)},y_{1n(1)}, c_{1n(1),})\}\rightarrow 1,$$
$$P_2=\{(x_{21},y_{21}, c_{21}), (x_{22},y_{22}, c_{22}),., (x_{2n(1)},y_{2n(1)}, c_{2n(1)})\}\rightarrow 2,$$
$$\ldots,$$
$$P_n=\{(x_{n1},y_{n1}, c_{n1}), (x_{n2},y_{n2}, c_{n2}),., (x_{nn(1)},y_{nn(1)}, c_{nn(1)})\}\rightarrow n,$$

where c_{i1} is a variable of color, and the number of all points is $N=n(1)+n(2)+\ldots+n(n)$.

The determined array is the input data for all the following processing algorithms working with chains of the circuit.

2.2 The Flood-Filling Algorithm

The flood fill algorithm is sometimes called initial fill. The seed is the starting pixel and then other seeds are planted in the area to change the color of the pixels. The algorithm replaces the internal color I_o of the object with the fill color I_n. When there are no more pixels of the original interior color, the algorithm terminates. This algorithm is based on a four-join or eight-join method for pixel filling. It searches for all adjacent pixels that are part of the interior.

The standard flood-filling algorithm requires for its work a uniform color surface when all pixels are of equal intensity. To improve the properties of the algorithm to apply to PCB images its principal formula is modified. For the available intensity

value I_o and the target intensity value I_n the algorithm fulfills the modified formulas:
a) tolerance only on lower intensities

$$\text{If } I(x_i, y_i) = I_o - (I_o * Tol), \text{ then } I(x_i,y_i) = I_n ,$$

b) tolerance only on higher intensities

$$\text{If } I(x_i, y_i) = I_o + (I_o * Tol), \text{ then } I(x_i,y_i) = I_n ,$$

c) tolerance on lower and higher intensities

$$\text{If } I(x_i, y_i) = I_o \pm (I_o * Tol), \text{ then } I(x_i,y_i) = I_n ,$$

where *Tol* is a relative tolerance to regulate rules on what surrounding pixel intensities must be changed (filled).

An example of the image to be filled and its distributed cumulative histograms from the OX and OY axes is shown in Figure 3. Arrows indicate the intensity values for red and blue. The intensity value of black is 0.

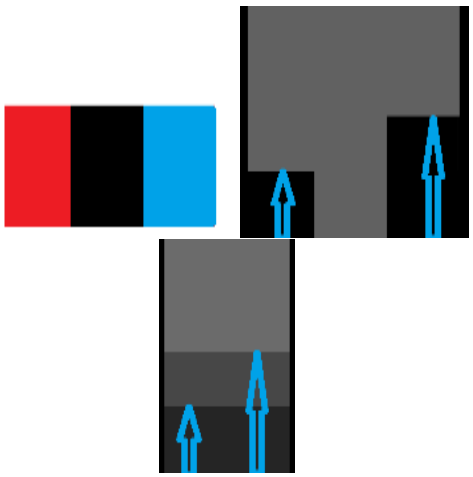


Fig. 3: Image "Three srips" (a) and its DCH images from OX and OY axes with an indication of the color intensity (b)

Two examples of filling of areas with equal and lower intensities in the image "Three strips" are shown in Figure 4. Starting from black with a big tolerance the algorithm fills red and black rectangles in orange. For a bigger tolerance starting from blue, the algorithm fills three rectangles in orange. So, the tolerance allows to marking of different components with the same color.

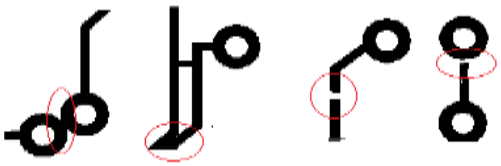
a b

Fig. 4: Two flood-filled images: red pixel as starting point (a) and blue pixel as starting point (b)

An example of the result of the filling algorithm when selecting circuits with different colors on a printed circuit board is shown above in Figure 2.

3 Separation of Defects

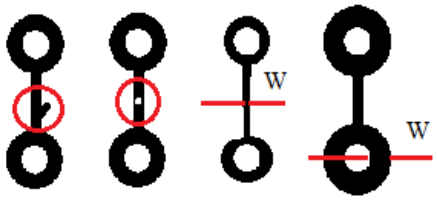
Defects are of various sizes and shapes. By their appearance, it is difficult to determine what effect they have on the functionality of the scheme. A very small strip can disrupt the transition of a signal, and a large round excess of connecting metal makes small changes in the operation of the circuit. This especially applies to short circuits and their breaks. The shape, size, and location of defects are random and may vary from one printed circuit board to another. Some examples of the most important defects as short and open are shown in Figure 5.



a b

Fig. 5: Two short (a) and two open (b) defects

A more difficult problem is to determine the type of track and contact defects. What amount of excess or insufficient metal is not just a defect, but an inaccuracy that affects the operation of the circuit and may or may not be corrected? Some examples of metal inaccuracies, such as metal excess and metal shortage, are encircled and shown in Figure 6. Examples of thinner (the width *W* is less than standard) and wider lines (the width *W* is greater than standard) are also shown in Figure 6.



a b c d

Fig. 6: Extra metal (a), shortage of metal (b), thinner (c) and wider (d) circuits

The first two defects in Figure 6 affect the passage of current and, in principle, are subject to repair. The third and fourth defects also change the value of the electric current, but they are difficult to eliminate.

When the PCB image is presented in a binary form possible defects are detected by the XOR

operations. They work with pixels by following formulas:

$$0 \wedge 1 = 1, 1 \wedge 0 = 1, 0 \wedge 0 = 0, 1 \wedge 1 = 0,$$

where 0 denotes white and 1 denotes black or vice versa.

These operations show surpluses and shortages of metal on the traces and contacts of the manufactured printed circuit board about the reference image. All defects belong to one class according to the way of their detection.

Subtraction operations with the binary image allow the separation of positive and negative defects:

$$0 - 1 = 0, \; 1 - 0 = 1, \; 0 - 0 = 0, \; 1 - 1 = 0.$$

Such operations are repeated twice: to a pair of the input images and swapped images. Positive and negative results form two classes of defects according to the way of their detection.

The third type of the subtraction operation is planned to realize the following tasks: to keep tracks and pins on the board, to separate defects into two classes by the intensity value, to separate constructive elements and defects by the intensity value, to keep defects visible on their places near constructive elements.

The operations to realize these tasks are as follows:

$$0\text{-}1=red,\ 1\text{-}0=blue,\ 0\text{-}0=0,\ 1\text{-}1=1,$$

where 0 denotes white and 1 denotes black or vice versa. *Red* and *blue* are of the constant intensity values.

An example of the last subtraction operation is shown in Figure 7.

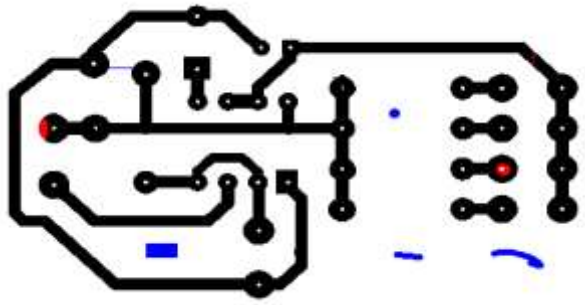


Fig. 7: The comparison difference between the two PCB images

The resulting image contains red and blue pixels reflecting associated and isolated areas that mark different types of defects.

4 Artificial Neural Networks for Defect Recognition

4.1 Defects Recognition by Cumulative Histograms

Reducing the inspection time in the production process of many printed circuit boards is possible thanks to the use of artificial neural networks (ANNs). They carry out a preliminary division of schemes into two classes: without defects and with defects. Later, the suspected circuit defects are processed by the developed software.

So, both the reference and the test images are prepared for the artificial neural networks to train and identify defective PCBs. But instead of images from two classes, the following approach uses the different images or their extracted features.

This approach is based on a single-layer neural network in Figure 8 with a reverse spread learning algorithm.

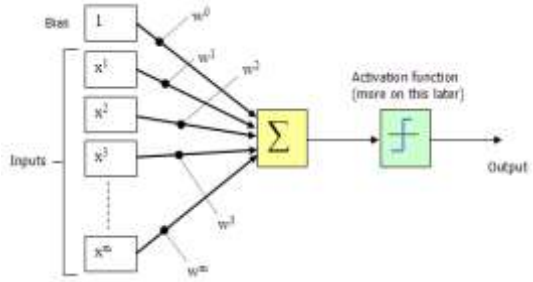
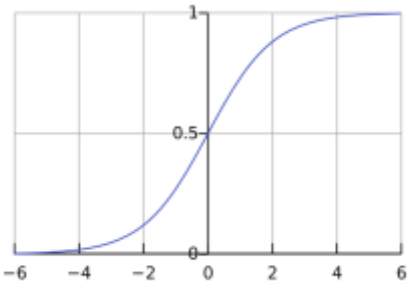


Fig. 8: Architecture of a single-layer ANN.

The single-layer ANN consists of *m* entries (vector *X*) connected to neural elements by a weight ratio. Neural elements, in their turn, consist of NET–elements and an activation function. The output neurons (vector OUT) represent the classification result for the input image. An important role in training and recognizing plays an activation function in which values are at the output neurons. The most common type of activation function used in neural networks is the exponential sigmoid, shown in Figure 9.



$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

Fig. 9: Exponential sigmoid curve and the formula

The value range of this function is (0; 1), obviously, during the recognition process, the closer the value of the output neuron is to 1, the more likely the input image belongs to a certain class.

The cumulative histogram was the first experimental feature for learning and recognition. It contains information about how many pixels belong to a certain brightness range (from 0 to 255).

The neural network was implemented in the Python programming language, in addition to the PyTorch and NumPy libraries, and was trained using the DeepPCB dataset [23], which contains 1500 pairs of images, each consisting of a defect-free template image and an aligned test image with annotations , including the location of the six most common types of PCB defects: open, short, mouse bite, spurs, pinholes, and false copper.

The accuracy of the neural network is calculated using this formula:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

where *TP* are instances belonging to the positive class that are accurately predicted as positive, *FP* are instances belonging to the negative class that are incorrectly predicted as positive, *TN* are instances belonging to the negative class that is accurately predicted as negative, *FN* are instances belonging to the positive class that is incorrectly predicted as negative.

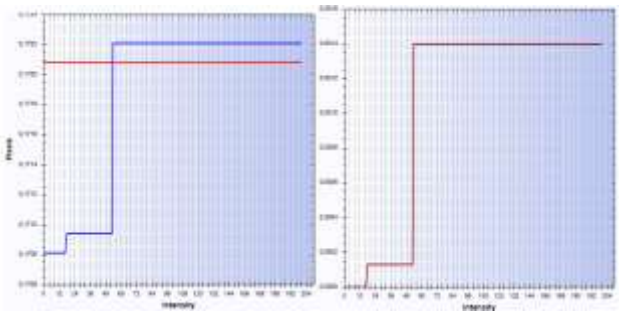
The cross-entropy loss function was employed as the loss metric during the training of the neural network:

$$Loss = -\frac{1}{n}\sum_{i=1}^{n} [\, y_i \, log(\underline{y_i}) + (1 - y_i) \, log(1 - \underline{y_i}) \,]$$

where n is the size of the dataset, y_i is a true label for the instance i, y_i is the predicted probability of the instance i belonging to the positive class.

When $y_i = 0$ a loss for the instance i will be lower when y_i approaches 0. When $y_i = 1$ loss for the instance i will be lower when y_i approaches 1, [15].

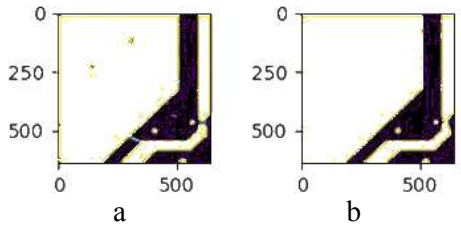
In the first experiment, cumulative histograms for the reference, sample, and image of the difference were calculated. They are shown in Figure 10 for one pair of tested images. Analysis of the charts demonstrates that two pairs of charts can be used as input in the artificial neural network. These pairs are cumulative histograms of the original template and sample images or cumulative histograms of the image differences. In the second case the difference between the two correct images does not contain pixels of defects and the cumulative histogram has only zero values. The second case was taken for the experiment because it allows to increase in the distance between the zero-value chart and the chart of the difference.



a b

Fig. 10: Cumulative histograms of two PCBs (a) and of the difference image (b)

For the experiment, the two classes were selected, which ANN should recognize and distinguish from one another – images with defects (fragments of PCBs), and without defects (two examples are shown in Figure 11).



a b

Fig. 11: Two fragments of PCB images: with defects (a), and without defects (b)

In the first experiment, 90 percent of the dataset was used for training the neural network, and 10 percent for testing it. The neural network was trained over 200 epochs, and the training process required a total duration of approximately 3 hours. As a result, the accuracy of the test dataset was approximately 75 percent. The change in the accuracy of the training process is shown in Figure 12.

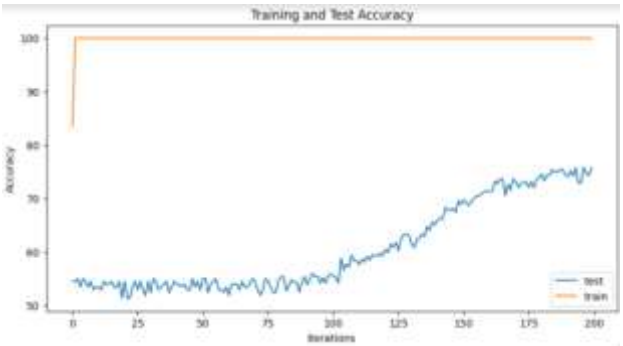
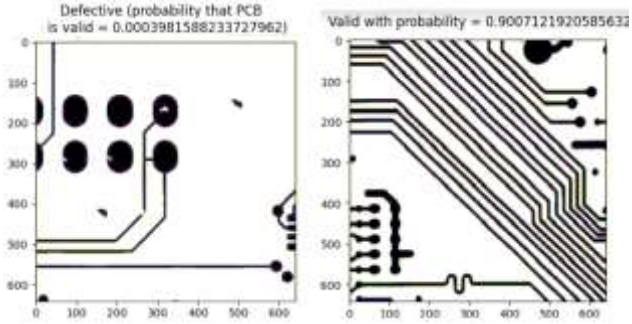


Fig. 12: Neural network accuracy evolution over epochs for training and testing data

As a result, fragments are divided into two classes. Two examples from them are shown in Figure 13, defective and valid circuits with a probability of their recognition.



a b

Fig. 13: Correct classification result of the printed circuit board with defects (a) and without defects (b)

The second experiment involves the use of the average mathematical value of the brightness of pixels. Each corresponds to a separate grid cell. This grid covers and divides the image into small parts. This feature encapsulates a variety of information, including the brightness in certain areas and the corresponding coordinates associated with that brightness.

The grid covers not the sample but the different image containing positive and negative pixels. Figure 14 illustrates the input image for the calculation of the mathematical means and later for processing by ANN. For illustration, a grid has a dimension of 10×10. In practice 20×20.The input chart for the ANN is shown in Figure 15.

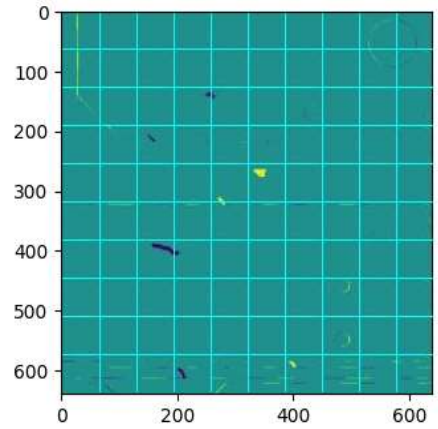


Fig. 14: Grid 10×10 on the difference image (its sizes are on the OX, and OY axes)

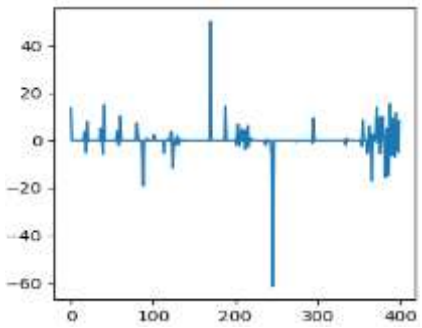
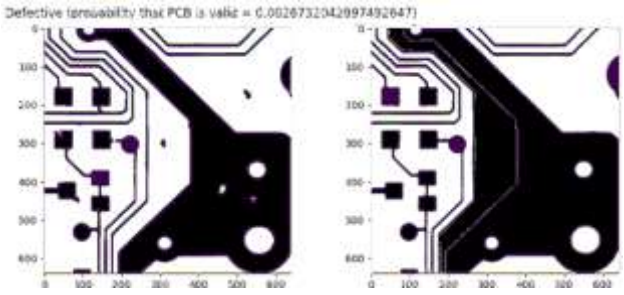


Fig. 15: Mathematical means in cells (opposite signs mark the two colors)

For the second experiment, 90% of the dataset was used for training of the neural network, 10% for testing of it. The cell size was set to 32 pixels in two coordinates. With a configuration of (640 / 32) * (640 / 32) = 400 input nodes, the neural network was trained over 200 epochs, which took approximately 4 hours. As a result, the accuracy of the test dataset was approximately 90 percent and fragments were divided into two classes. Two examples from them are shown in Figure 16.



a b

Fig. 16: Correct classification result of the printed circuit board without defects (a) and with defects (b)

In summary, the more pixels the difference image contains, the higher the probability observed

in both experiments when classifying defective samples.

5 Separation of Defects

After the subtraction-comparison operation, two sets of defects are distinguished. The first is characterized by metal deficiency and the second by excess metal. They are denoted as positive (red *p*) and negative (blue *n*) defects if the reference image is reduced and the sample image is subtracted:

$$P = P_e + P_s \text{ or } P = P_p + P_n.$$

All shortage metal defects are neighboring to the constructive elements: tracks and pins. Among extra metals, defects are neighboring to the constructive elements and some are isolated:

$$P = P_{es} + P_{ei} + P_s ,$$

where P_{ei} is a set of isolated defects, and P_{es} is a set of extra metal defects having contact with tracks or pins.

All defects from the previous PCB image are segmented and shown in Figure 17. By colors they are grouped into two classes encircled by red and blue.

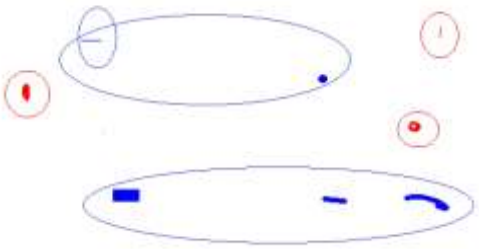
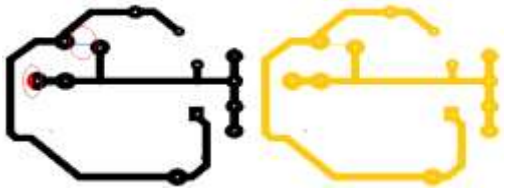


Fig. 17: Two classes of defects: encircled with blue is an extra metal, and with red is a shortage of metal

Colors are auxiliaries for defects, which otherwise are grouped into two classes: those isolated from the circuit and those associated with traces or contacts. In turn, the second class is divided into two subclasses: connection defects and defects in the size and shape of structural elements.

Isolated defects are separated from the image difference by an additional procedure. The procedure applies the flood-filling algorithm to *n* starting points representing each chain of the circuit. The target color is white. Due to the tolerance value black traces, red extra-metal defects, and shortage of metal become white which is invisible for other processing algorithms. An example of one filled component is shown in Figure 18. For visual demonstration, yellow is used in the figure instead of white.



a b

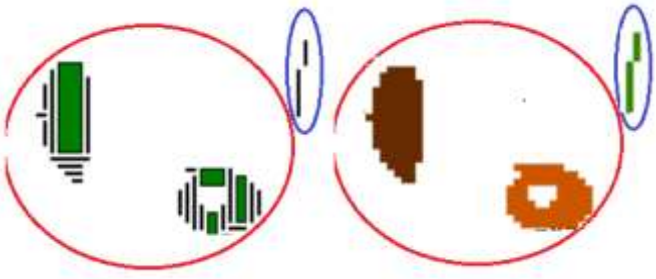
Fig. 18: Two chains with defects before (a) and after filling (b)

When all components are filled in white, defects from P_{es} , P_s are invisible, and other defects from P_{ei} are marked in green to distinguish them from all others previously discovered.

No one can tell if it is a short circuit, an open circuit, or some other type of defect. In automatic mode, all defects are grouped into only three classes by color. As a rule, in the production process, all defects are of different types, shapes, and sizes. Neural networks trained on one type of defect will necessarily miss other types.

All defects are visual, but there is no access to their pixels. In such a presentation, they cannot be taken as starting points or separated for analysis. They cannot be transferred from one image to another.

In this case, the hierarchical clustering algorithm with a step 1×1 (one pixel) over a black image is applied to form all connected areas of defects. Two classes of defects D_{es} and D_s associated with traces and contacts are separated and then clustered. This is illustrated in Figure 19(a) by examples with clustered defects in two classes. In Figure 19(b) defects are marked and selected with different colors.



a b

Fig: 19: Increased clustered defects (a) and marked with full colors (b)

All clustered defects are described as objects with the coordinates of the pixels that form these defects:

$$P_p(i) = \{(x_{1i}, y_{1i}), \ldots, (x_{ki}, y_{ki})\}, \; k = |P_p(i)|,$$

$$i=1,2,...,|P_p|$$

Identifiers are assigned to these objects. Class numbers and the sequence positions in the class serve as keys for access to them. They are planned to be transferred to the canvas of the reference PCB image. In this stage, defects are not identified by the circuit number to which they belong.

6 Determination of Type and Intensity of Defects

6.1 The Metal Deficiency Class

In automatic mode to identify these defects by the trace sequence number there is no other way except to fill the chain with a color different from the existing one. The reference image of the printed circuit board is a working model for the procedure to determine open and short defects. It serves also for the investigation of the influence of trace and contact inaccuracies on the stability of the circuit.

Firstly, blue, and then red defects from corresponding classes one by one are transferred to the reference image. An example of one transferred defect is shown in Figure 20.

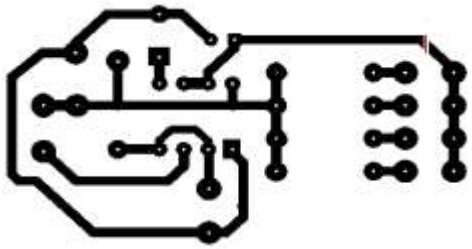


Fig. 20: The reference PCB covered by an object (defect)

Initial points are formed earlier according to the thinning algorithm. They form the so-called chain status monitor. It is shown in Figure 21 only for the two *k*-th and (*k*+1)-th chains.

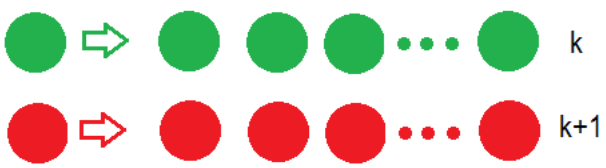


Fig. 21: Two status monitors of two chains: the (*k*+1)-th chain is filled

It's like a traffic light: green - no fill result, red - all identification points of the chain have changed color. Since the monitors are linked to tracking numbers, the transmitted defect receives its identification number when all points of the chain have changed color.

The received identifier makes it possible to select a clean reference chain and a manufactured chain with a defect by adding to the reference a transferred defect. These two circuits are the basis for analyzing the effect of the defect on the operation of the circuit.

Such a procedure is implemented for each object belonging to the metal deficiency class P_s and the additional metal class P_e. Elements from the first class insert defects such as circuit breaks and reduction of the conductive surface, and from the second-class short circuits and an increase in the conductive surface.

The input data are the starting points of the k-th chain, coordinates of the pixels forming the s-th defect, and a traffic green light with all green components.

When the object defect is transferred the filling algorithm is applied to its pixels marking it with white. This operation returns the state of the circuit to what it was in the produced circuit with one defect. Figure 22 shows one chain in which the red defect was marked in white.

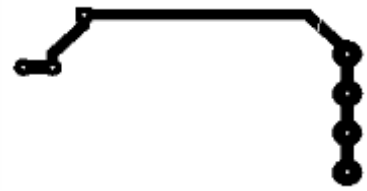


Fig. 22: One chain for testing

Now, taking on the element from the set P_k as the starting point the flood-fill algorithm fills this chain with red. After filling, the traffic light is yellow because its components are red and green (Figure 23(a)). The *k*-th chain receives two components of contacts and traces marked with black and red (Figure 23(b)).

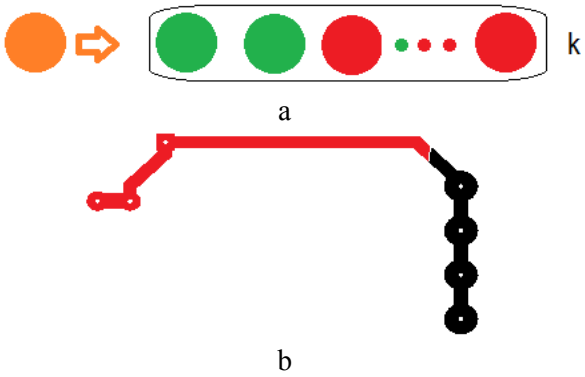


Fig. 23: The chain status monitor (a) of the partially filled chain (b)

As a result, after one iteration, a type of defect and its coordinates are determined.

6.2 The Extra Metal Class

When an additional metal object representing a short defect is transferred to the image, a similar process takes place. This involves barrier-free filling from the pixel of the extra metal object to ascertain the number of connected chains, whether it's one or a few. Filling the blue object with black, and then together with the rest of the associated chain with red, selects the circuit. Figure 24(a) demonstrates how the short defect causes the two chains to be filled with red. Status monitors of these chains signal this occurrence. An example is shown in Figure 24(b) indicating the number of the filled chains (or numbers of some filled chains).

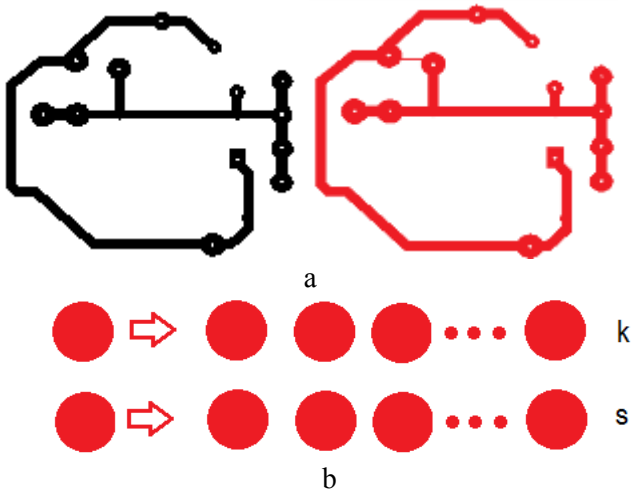


Fig. 24: Two chains with short circuit (a) and their filled status monitors (b)

After termination of the procedure, all tested objects are classified as three types of defects: open, short circuit, and other associated with traces but not affecting the connectivity of contacts.

6.3 Trace Resistance Defects

If the next object is not a connection defect, it displays an increase or decrease in conduction resistance due to a decrease or increase in metal surface area. It is difficult to find an accurate estimate of the value that indicates a hundred percent failure because the defect can be concentrated in one place or distributed over traces and contacts. Therefore, the defect is measured as distributed, which is calculated by the surface square and the surrounding perimeter. Also, the defect can be measured as concentrated when calculated only in areas of its placement.

A distributed change in the resistance of the entire track is indicated by a change in the width of the entire track D_W. A local concentrated change in resistance in a certain section of the track is indicated by a local change in the width W_d of the track in this section.

For example, Figure 25 shows two fragments of traces without and with defects of a lack of metal. The width of the track and its length are marked. Defects with a lack of metal, which causes a change in resistance, are circled. Sections with a decreased width are marked.

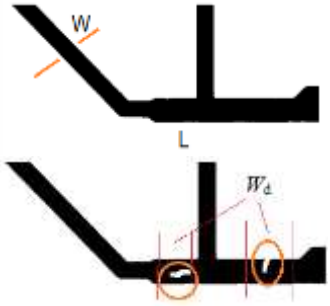


Fig. 25: Two chains: reference (a), with defects (b)

The trace conductance resistance R_s is proportional to the length L of the trace and inversely proportional to its width W:

$$R_s \approx (L/W)$$

Traces in the reference and manufactured PCBs have almost the same length but due to defects can differ by a width in some places. This difference can be measured approximately. For that squares S_1, S_2 of two chains are calculated. Borders of two chains are found by the edge detection algorithm [24] and their two lengths are measured. A border's length is twice as large as a trace's length. For two traces in Figure 26(a) and Figure 26(c), measured values for the reference circuit are S_1=865, L_1=204, and for the circuit with defects S_2= 780, L_2=236, i.e., in the second case, the area is smaller, and the length is longer.

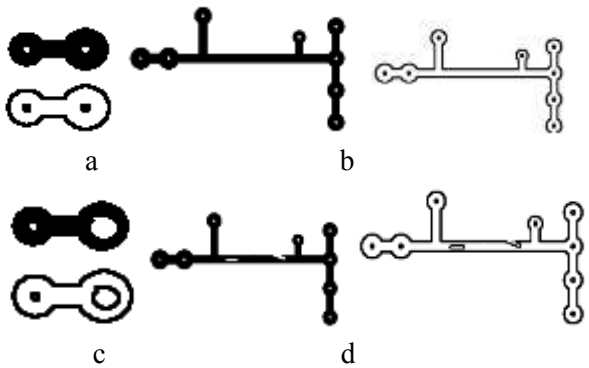


Fig. 26: Chains and their edges with and without defects: small chains (a, c), large chains (b, d)

A trace's length is twice as short as a border's length:

$$L_1(t)=L_1/2,\ L_2(t)=L_2/2.$$

The approximated average widths of the two traces are as follows:

$$W_1=S_1/L_1(t),\ W_2=S_2/\ L_2(t).$$

These widths are average and approximate. They are not equal exactly to real widths. They are required only for the estimation of losses in the resistance.

In the case of the small trace in Figure 26(a) W_1=8.4, W_2=6.6. For the whole length the integrated distributed width difference $D_W=W_2-W_1$=-1.8 is essential due to the small sizes of the traces (only two close contacts). For large sizes, it is much smaller.

The concentrated difference caused by a change in squares and lengths in chains with defects is calculated by the following formulas:

$$S_d= S_2 -S_1,\ L_d= L_2 -L_1\ ,\ W_d= S_d/\ L_d.$$

For the example in Figure 26(c) a change in the width caused by a defect approximately is equal:

$$S_d=-85,\ L_d=32,\ \text{and}\ W_d=-2.7.$$

The minus sign in W_d indicates that the defect is due to a lack of metal. The obtained value is compared to the average width W_1 of the reference chain (8.4) to determine whether a new width or a new resistance will affect the work of the manufactured circuit. In the considered case the new width is as follows:

$$W_{1n}=W_1+W_d=8.4-2.7=\ 5.7.$$

So, the width difference in a place of defects is 2.7 which exceeds 30 percent of the reference value. The conduction resistance of the trace increases in accordance.

One more example with the larger chain is shown in Figure 26(b) and Figure 26(d). For this sample, the calculated features are of the following values:

$$S_1=5529,\ S_2=5421,\ L_1=1348,\ L_2=1410,$$
$$L_1(t)=674,\ L_2(t)=705,$$
$$W_1=8.20,\ W_2=7.68,$$
$$S_d=-108,\ L_d=62,\ \text{and}\ W_d=-1.74$$

The integrated difference W_2-W_1=-0.52 is small compared with the reference width W_1=8.20 (6 percent). However, the concentrated average difference W_d=-1.74 is about 22 percent of the reference value. Defects in some places can increase the resistance by a larger percentage. In this case, the scheme's failure is a matter of time.

All measured values are in pixels and easily converted to real values. The difference in width is negative in the case of a lack of metal and positive in the case of an excess of metal in traces and contacts.

The last case is demonstrated in Figure 27.

a b

Fig. 27: Fragments and their edges without defects (a), and with defects (b)

For separated fragments S_1=775, L_1=120 (240/2), and for the circuit with defects S_2= 870, L_2=124 (247/2), i.e., in the second case, the area is greater, and the length is longer. After the calculation of intermediate values, the widths of the chains and the difference are as follows:

$$W_1=6.46,\ W_2=7.04,\ D_W= W_2- W_1=0.58.$$

The difference between the lengths

$$L_d= L_2 – L_1= 7$$

does not correspond to the real size of a defect because a part of the defect length belongs also to the reference trace and is subtracted.

So, in the case of an extra metal, a defect must be presented in its determined color: red or blue. An example is shown in Figure 28(a). Then, the determined edge contains three components: the main native black, red for the external border of the defect, and green for the common border between the trace and defect.

a b

Fig. 28: A fragment with defect (a) and its edge (b)

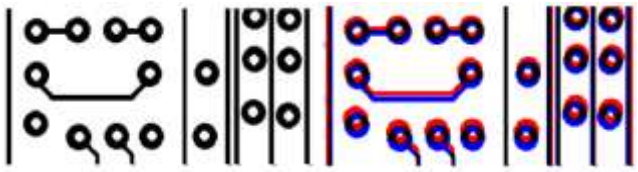
Now the difference between trace areas is S_d=95 which is a square of the defect, and the length of the red edge is L_d=43. Under these conditions, the concentrated width of the trace in one place is W_d=95/43=2. Additional metal causes the final difference W_d=2 which is more than 30 percent of the reference width. The conduction resistance is reduced by one-third in this place. On the contrary, the distributed losses of resistance show less than 10 percent.

In conclusion, two criteria of the failure caused by defects in traces are possible: tolerance on the

distributed width of the trace D_W and tolerance on the concentrated width of the trace W_d.

6.4 Division of Traces into Fragments

The conduction resistance can be accepted as a criterion function for detecting defects and checking the functionality of the circuit by analyzing its printed circuit board. In this case, the shift of tracks and small changes in their size are not considered automatically. For example, Figure 29 shows two fragments: one is the reference image and the second is the difference between it and the shifted-by-pixels fragment. Are there defects in the traces? It is difficult to answer. The summary number of red or blue pixels is large enough.



a b

Fig. 29: Fragments of PCB image (a) and its difference with shifted one (b)

When considering the traces as reference, with a lack of metal and with an excess of metal in Figure 30 their approximate widths are of the following values W_1=*3.46*, W_2=*3.26*, W_3=4.35.

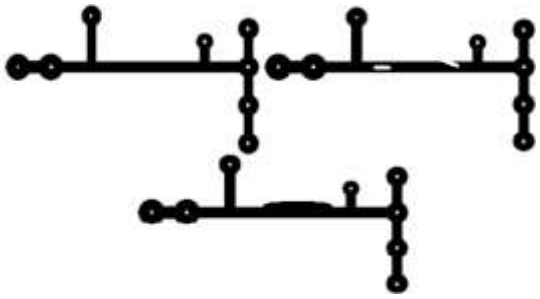


Fig. 30: Chains without and with defects

Unfortunately, in this approach, only one parameter characterizes the chain in the reference and sample PCB image. Changes in the local conductance resistance could be distributed along the resistance of other parts belonging to the long trace. Then the defect is compensated by the correct values of the parameter. So, in the previous approach, only places with defects were analyzed.

It is possible to exclude the preliminary detection of the location of defects if the concept of distributed conductive resistance of the track is used. Namely the values of the width of the track in its various sections. Instead of one parameter W, the trace should be characterized by the vector W= (W_1, W_2,…,W_n), where n is the number of sections of a trace.

In general, trace division is a complex and separate problem, [25], since the traces have a horizontal, vertical, and hybrid arrangement. They are of different sizes and irregular shapes. Traces are uniform in color. The number of parts is planned according to the area of the entire trace and the size of the separated part. Therefore, each trace is considered for division into parts separately.

The main idea used for the division of a trace is flood-filling them with different colors. At the beginning the mask image with n parts – for example, rectangles is generated. Each rectangle is filled in its random color. An example with 6 parts is shown in Figure 31(a).

a b

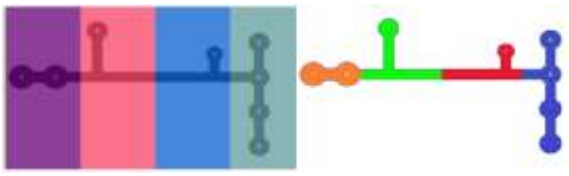
Fig. 31: Rectangles with 6 random colors (a) and overlay with a black chain (b) by its sizes

Then an image with the black trace is overlayed with rectangles by the following formula applied to every channel of the pixel:

$$I_r(\mathrm{x,y})=\alpha \mathrm{I}_c(x_i,\mathrm{y}_i) + (1-\alpha)I_b(x_i,\mathrm{y}_i)(r),$$

where I_r, I_c, I_b are the pixel intensity of the resulting, rectangles, and chain images, α is the coefficient of transparency. The resulting image with four rectangles is shown in Figure 32(a).

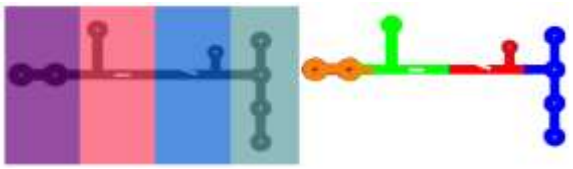
Now, four fragments are characterized by four various colors. They are easy to select by applying the flood-filling algorithm to fragments and backgrounds. They are shown in Figure 32(b). Fragments of the trace are ready for the principal processing: edge detection and measurements of the area and length of the perimeter.



a b

Fig. 32: Overlay of the chain and rectangles (a) and its filled sections (b)

In the case of the trace with defects from Figure 30 rectangles and the trace are overlayed, flood-filled, and separated too (Figure 33).



a b

Fig. 33: Overlay of the chain with defects and rectangles (a) and its filled sections (b)

After processing the two traces in the reference and sample images they are characterized by two vectors of fragmentary conductance resistance (in this case, the trace width):

$$W_r = (W_{r1}, W_{r2}, \ldots, W_{rn}) , W_s = (W_{s1}, W_{s2}, \ldots, W_{sn}),$$

where n is the number of fragments, the same for similar traces and different for other traces.

The two conduction resistance vectors are calculated and have the following components:

$$W_r = (8.20, 7.42, 9.16, 9.41),$$
$$W_r = (7.20, 7.53, 8.96, 7.12).$$

The vectors show that the two metal deficiency defects increase resistance on the green and red sections of the trace. Losses are 12 and 24 percent in accordance.

The two types of defects affect resistance diametrically oppositely: a lack of metal increases, and an excess of metal decreases the resistance in the trace. Therefore, criterion functions are formed separately for each case:

$$M = \text{sqrt}\ \{ 1/N^- * \sum (W_{rk}(i) - W_{sk}^-(i))^2\},$$
$$P = \text{sqrt}\{ 1/N^+ * \sum (W_{rk}(i) - W_{sk}^+(i))^2\},$$

where M is a standard deviation for defects $W_{sk}^-(i)$ reducing the resistance, and P is a standard deviation for defects $W_{sk}^+(i)$ increasing the resistance, N^-, N^+ are the numbers of correspondent defects in all traces determined after they were processed, i is the sequence number of the trace, k is the sequence number of the section in the trace.

As a result, all inaccuracies in the components of the printed circuit board are measured and prepared to decide the readiness of the circuit for operation or not.

To visualize the location and type of defects, the difference in the resistance parameters is checked in comparison with the tolerance value set by the user: tol^+ for additional resistance, and tol^- for reduced resistance:

$$W_{rk}(i) - W_{sk}^-(i) > tol^-,$$
$$W_{rk}(i) - W_{sk}^+(i) > tol^+.$$

If one of the equations is satisfied, the section $W_{sk}^-(i)$ is filled in blue, or the section $W_{sk}^+(i)$ is filled in red. the tolerance can be chosen such a value that the scheme should not only be marked but also rejected.

7 Conclusion

The work is devoted to the investigation of the ANN application to learn and recognize defective PCB images by their features, particularly: the cumulative histograms of defect pixels, and the grid cell intensity mean. Testing of the implemented algorithms has revealed, that trained single-layer ANN can successfully classify the PCB images, where the aspect is placed on the range of defect colors and their location.

The approach includes such algorithms and methods as mathematical formulas of comparison, the filling algorithm for selecting, separating, and dividing chains, formulas for calculating the resistance characteristics of circuit components, and K-means clustering for the selection of objects in the PCB image.

The proposed approach contains operations for the selection of three classes of defects: negative, positive, and connection. Marked with different colors, these defects are used to train the single-layer neural network for further automatic division of the PCB image into two classes: having defects and without defects. It reduces the time consuming of the developed software which is planned to indicate the location and determine the intensity of defects. In conclusion, such software will reduce hardware and make the PCB diagnostic process cheaper.

The goal of future work is to modify the subtraction operation using an approach that is independent of the bias and dimensions of the tracks and pins. To do this, it is necessary to investigate new features of printed circuit boards, such as the description of connection defects, excess and lack of metal, resistance properties, and the absence of contact areas. The development of the automation of circuit division into parts, the creation of specific neural networks for printed circuit board images, and their application to recognize defective circuits is envisaged.

References:

[1] Elena Jasiuniene, Renaldas Raišutis, Vykintas Samaitis and Audrius Jankauskas. Comparison of Different NDT Techniques for Evaluation of the Quality of PCBs Produced Using Traditional vs. Additive Manufacturing Technologies. A survey. *Sensors,* 2024, 24, 1719, DOI: 10.3390/s24061719.

[2] Cai L., Li J., PCB defect detection system based on image processing, *Journal of Physics: Conference Series*, MMEAT-2022, May 27-29, 2022 , Qingdao, China,2022, Vol. 2383, pp. 012077, IOP Publishing, DOI: 10.1088/1742-6596/2383/1/012077.

[3] Scott P.D., Krastev E., Nordson Dage, P.E., 2D X-ray Inspection with Materials and Thickness Identification, *Proceedings of SMTA International, Sep. 25 - 29, 2016, Rosemont, IL, USA*, pp. 470-475, DOI: 0.1134/S1061830917080058.

[4] Zahia Guezoui, Aicha Baya Goumeidane, Nafaa Nacereddine, Weld Defect Radiographic Image Segmentation with Finite Mixture Model (FMM), *Engineering World*, 2020, Vol. 2, Art. #20, pp. 134-138.

[5] Ebayyeh A.A.R.M.A., Mousavi, A., A Review and Analysis of Automatic Optical Inspection and Quality Monitoring Methods in Electronics Industry. *IEEE Access,* 2020, Vol. *8*, pp.183192–183271, DOI: 10.1109/ACCESS.2020.3029127

[6] Zhu A. Wu and X. Liu, Printed circuit board defect visual detection based on wavelet denoising, *IOP Conference Series: Materials Science and Engineering*, MTMCE-2018, 22–24 June, Zhuhai, China,2018, Vol. 392, pp. 062055, DOI: 10.1088/1757-899X/392/6/062055.

[7] Y. Hanlin and W. Jun, Automatic Detection Method of Circuit Boards Defect Based on Partition Enhanced Matching, *Information Technology Journal*, 2013, Vol.12, Iss. 11, pp. 2256-2260, DOI: 10.3923/itj.2013.2256.2260.

[8] Vikas Chaudhary, Ishan R. Dave and Kishor P. Upla, S. V., Visual Inspection of Printed Circuit Board for Defect Detection and Classification. *International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET),* Chennai, India, 22-24 March, 2017, pp. 732-737, DOI**:** 10.1109/WiSPNET.2017.8299858.

[9] Stella Vetova, Covid Image Classification using Wavelet Feature Vectors and NN, *Engineering World*, 2021, Vol. 3, pp. 38-42.

[10] Ling Q., Isa N.A.M., Printed Circuit Board Defect Detection Methods Based on Image Processing, Machine Learning and Deep Learning: A Survey. *IEEE Access,* 2023, Vol. 11, pp. 15921–15944, DOI: 10.1109/ACCESS.2023.3245093.

[11] Medium. S. McClure, Building an End-to-End Defect Classifier Application for Printed Circuit Boards Detecting, Extracting and Classifying Circuit Board Defects using Image Processing and Deep Learning, 2020, [Online], https://towardsdatascience.com/building-an-end-to-end-deep-learning-defect-classifier-application-for-printed-circuit-board-pcb-6361b3a76232 (Accessed Date: April 28 2024).

[12] V. A. Adibhatla, H.-C. Chih, C.-C. Hsu, J. Cheng, M. F. Abbod, and J.-S. Shieh, Defect Detection in Printed Circuit Boards Using You-Only-Look-Once Convolutional Neural Networks, *Electronics*, 2020, Vol. 9, No. 9, pp. 1547, DOI: 10.3390/electronics9091547.

[13] Abhiroop Bhattacharya, Sylvain G. Cloutier, End-to-end deep learning framework for printed circuit board manufacturing defect classification, *Scientific Reports*, 2022, [Online], https://www.nature.com/articles/s41598-022-16302-3 (Accessed Date: April 27, 2024).

[14] Jungsuk Kim, Jungbeom Ko, Hojong Choi and Hyunchul Kim, Printed Circuit Board Defect Detection Using Deep Learning via A Skip-Connected Convolutional Autoencoder, *Sensors,* No.21(15), 2021, 4968, DOI: 10.3390/s21154968.

[15] Charu C. Aggarwal, Neural Networks and Deep Learning, A Textbook, *Springer*, 2018, 495 p. DOI: 10.1007/978-3-031-29642-0.

[16] M. H.Tatibana, R. and de A. Lotufo, Novel Automatic PCB Inspection Technique Based on Connectivity, *Proceedings X Brazilian Symposium on Computer Graphics and Image Processing*, Campos do Jordao, Brazil, 1997, pp. 187-194, DOI: 10.1109/SIGRA.1997.625174

[17] K. P. Anoop, N.S. Sarath and V. V. Sasi Kumar, Review of PCB Defect Detection Using Image Processing, *International Journal of Engineering and Innovative technology (IJEIT),* 2015, Vol. 4, Is. 11, pp. 188-192, ISSN: 2277-3754, DOI: 10.1088/1742-6596/2383/1/012077.

[18] D.B. Anitha, and M. Rao, A survey on Defect Detection in Bare PCB and Assembled PCB

using Image Processing Techniques, *International Conference on Wireless Communications, Signal Processing and Networking*, Chennai, India, 2017, pp. 39-43, DOI: 10.1109/WiSPNET.2017.8299715.

[19] F. B. Nadaf and V. S. Kolkure, Detection of Bare PCB Defects by using Morphology Technique, *Morphology Technique International Journal of Electronics and Communication Engineering*, 2016, Vol. 9, No. 1, pp. 63-76, DOI: 10.23919/ChiCC.2017.8029117.

[20] Y. Hanlin, W. Jun, Automatic Detection Method of Circuit Boards Defect Based on Partition Enhanced Matching, *Information Technology Journal*, 2013, Vol. 12(11), pp. 2256-2260, DOI: 10.3923/itj.2013.2256.2260.

[21] J. Nayaka, K. Anitha, B.D. Parameshachari, R. Banud and P. Rashmi, PCB Fault Detection Using Image Processing, *IOP Conference Series: Materials Science and Engineering*, CMAEM-2017, 3–4 July 2017, Narsimha Reddy Engineering College, India, 2017, Vol. 225, pp. 1-5, DOI:10.1088/1757-899X/225/1/012244.

[22] S. Guan, F. Guo, A New Image Enhancement Algorithm for PCB Defect Detection, *International Conference: Intelligence Science and Information Engineering (ISIE)*, 20-21 August, Wuhan, China, 2011, pp. 454-456, DOI: 10.1109/ISIE.2011.54.

[23] Sanli Tang, Fan He, Xiaolin Huang, Jie Yang, Online PCB Defect Detector On A New PCB Defect Dataset. arXiv: Computer Vision and Pattern Recognition, [Online] https://arxiv.org/abs/1902.06197 (Accessed Date: April 27, 2024).

[24] Basak Aldemir, Elif Guner, Halis Aygun, Linear Diophantine Fuzzy Sets: Image Edge Detection Techniques based on Similarity Measures, *WSEAS Transactions on Signal Processing*, 2023, Vol. 19, pp. 205-214, https://doi.org/10.37394/232014.2023.19.22 .

[25] Melnyk, R., Korotyeyeva,T., Levus,Y., Chains Defects Detection in a Printed Circuit Board Image by the Plane Partition and Flood-filling of Traces. *International Journal of Computing*, 2023, 22(1), pp. 35–42, DOI: 10.47839/ijc.22.1.2877.

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

The authors contributed in the ratio of 70 (Melnyk) to 30 (Vorobii) in the present research, at all stages from the formulation of the problem to the final findings and solution.

Sources of Funding for Research Presented in a Scientific Article or Scientific Article Itself

No funding was received for conducting this study.

Conflict of Interest

The authors have no conflicts of interest to declare.