Concatenated Network Fusion Algorithm (CNFA) Based on Deep Learning: Improving the Detection Accuracy of Surface Defects for Ceramic Tile

Kan Wang 1,*, Zeren Li 1,2 and Xu Wang 1

1 Institute of Fluid Physics, China Academy of Engineering Physics, 64 Mianshan Road, Mianyang 621900, China; zeren109@yeah.net (Z.L.); wangxu.289525@163.com (X.W.)
2 Center for Advanced Material Diagnostic Technology, College of Engineering Physics, Shenzhen Technology University, 3002 Lantian Road, Shenzhen 518118, China
* Correspondence: wangkan83@126.com

Abstract: The low accuracy of detection algorithms is one impediment in detecting ceramic tile’s surface defects online utilizing intelligent detection instead of human inspection. The purpose of this paper is to present a CNFA for resolving the obstacle. Firstly, a negative sample set is generated online by non-defective images of ceramic tiles, and a comparator based on a modified VGG16 extracts a reference image from it. Disguised rectangle boxes, including defective and non-defective, are acquired from the image to be inspected by a detector. A reference rectangle box most similar to the disguised rectangle box is extracted from the reference image. A discriminator is constituted with a modified MobileNetV3 network serving as the backbone and a metric learning loss function strengthening feature recognition, distinguishing the true and false of disguised and reference rectangle boxes. Results exhibit that the discriminator appears to have an accuracy of 98.02%, 13% more than other algorithms. Furthermore, the CNFA performs an average accuracy of 98.19%, and the consumption time of a single image extends by only 64.35 ms, which has little influence on production efficiency. It provides a theoretical and practical reference for surface defect detection of products with complex and changeable textures in industrial environments.

Keywords: ceramic tiles; surface defects; accuracy; deep metric learning; rectangle box; discriminator

1. Introduction

With the rapid development of infrastructure construction in the past two decades, ceramic tile (CT) production capacity has increased significantly and has gradually become an essential industry for national economic growth [1]. CT production mainly includes raw materials mixing and grinding, dehydration, firing, inkjet, and polishing. During this period, mechanical or glaze defects such as corner/edge cracks, pinholes, delaminations, glaze bubbles, and scratches, etc., will inevitably appear on the surface. Defect detection is a key link to prevent defective products from entering the market. Due to the lack of high accuracy and robustness of surface quality automatic detection technology, manual inspection, which is manipulated mainly by specially trained human operators who visually inspect every CT and grade the quality of CTs, still dominates in CT surface defect detection [2]. Workers who observe fast-moving CTs on the assembly line under bright light for a long time are prone to visual fatigue, which induces many problems, such as missed defect inspections, defect types and dimensions that cannot be quantified, and statistical deviations in product qualification rates. At the same time, colossal labor intensity has brought higher labor costs to enterprises [3].

In recent years, machine vision technology to detect defects on the surface of CTs has gradually become a trend. The effectiveness and practicability of this technology depend critically on the performance of image processing algorithms, which has also
become a research hotspot for scholars all over the world. Andrzej Sioma [4] used line laser scanning to obtain three-dimensional images of CT and mainly adopted morphological transformation for crack-type defects to realize the extraction and location of the edge features of the defects. Li D et al. [5] proposed a method for CT surface defect detection based on texture clustering. Gabor wavelet transform was used to extract defect features, and they designed custom coefficients to enhance the gap between the defect and the background. Defect classification and segmentation were performed through fuzzy K-means clustering and a priori threshold. Hanzaei SH et al. [6] combined the invariant rotation measure of local variance with morphological operators to extract defect features and then used support vector machines (SVM) for classification. It has achieved good results in the detection of single-color CT defects. Casagrande L et al. [7] extracted defect features based on a combination of segmented fractal texture analysis (SFTA) and discrete wavelet transform, then a genetic algorithm was applied to optimize classification parameters and feature classification relied on SVM. The detection effect of scratches, spots and pits on 20 cm CT was better than that of Local Binary Pattern (LBP) and Gray-Level Co-occurrence Matrix (GLCM). Zhang et al. [8] proposed a detection method for CT surface defects with complex textures. Based on the HSV color space, after the traditional saliency detection method was used to segment the defects, the color histogram features of the segmented defect rectangles were extracted, then the defect discriminator was constructed by SVM for two classifications. The algorithm finally achieved the effect of improving the accuracy of defect detection. Zorić B et al. [9] proposed to use the 2D Fourier transform to obtain the power spectrum of the CT images as a feature vector and construct a classifier based on K nearest neighbors and random forest for defect classification. Prasetyo MD et al. [10] obtained the LBP feature from the CT images and then trained an SVM classifier. The results indicated that the defect detection accuracy of the linear SVM classifier reaches 87.5%. Zhang Jun et al. [11] proposed an edge defect detection algorithm for CTs based on morphology and wavelet transform. The separated high-frequency and low-frequency components were obtained through the second wavelet transform. The low-frequency components were combined with morphological transformation to remove the background, the defect information of the high-frequency components was extracted with the help of filters, and finally the defect contours were reconstructed. Li Junhua et al. [12] proposed a multi-feature fusion detection algorithm for CT surface defects. The improved the scale-invariant feature transform (SIFT), the color moments were combined to extract image features, and the combined integrated feature vectors were sent to the SVM classifier for final classification.

As mentioned earlier, traditional image processing methods that rely on artificial design defect features and utilize machine learning classification have made some achievements in defect detection for CTs. Many scholars who directly adopt deep convolutional neural networks (DCNN) for feature extraction and classification have also achieved good defect detection accuracy. Liu Chang et al. [13] used the improved U-Net model to extract the characteristics of magnetic tile defects and trained the neural network for defect classification. The defect detection accuracy rate reached 94%. Kan Wang et al. [14] proposed a detection algorithm that synchronously compares the CT defect image and the normal image, they utilized a deep neural network to detect the defect image and the corresponding normal image at the same time, and judged the detection result through a comparator to reduce the false detection rate. Li Zehui et al. [15] presented a detection method for CT surface defects based on a lightweight convolutional auto-encoding network. The auto-encoder was trained with non-defective CT images, and defects were gained by defective image reconstruction. B Mariyadi et al. [1] extracted GLCM features of CT to characterize image defects, CNN was applied to the classification, and the accuracy rate reached 83% on 20 cm tiles. Junior GS et al. [16] combined traditional image processing methods with deep learning, and traditional methods such as contrast enhancement and equalization were used as preprocessing units to improve the CT cracks and background contrast, and then trained the U-Net segmentation model for defect segmentation. Huang W et al. [17]
applied the YOLOv3-tiny model to the mobile phone ceramic substrate to detect scratches with an accuracy of 89.9%. We summarized the results of previous researchers, as shown in Table 1.

| No. | Type of Defects Detected | Type of Used Feature | Background Texture of CTs | Defects Discrimination | Evaluation Metrics |
|-----|--------------------------|----------------------|---------------------------|------------------------|-------------------|
| [1] | Corner or edge, glaze, pin-hole, cracks | GLCM | Simple textures | Without | Accuracy: 83% |
| [4] | Cracks | Morphological transformation | | Without | Not indicated |
| [5] | Cracks, pit and damage defects | Gabor energy spectra | | Without | Not indicated |
| [6] | Pin-holes, holes, cracks | Invariant rotation measure of local variance | | Without | Accuracy: 93.4% |
| [7] | Cracks, holes and spots | GLCM, HOG, DWT, SFTA | Without and monochromatic | Without | Accuracy: 99.01%, 93.2% |
| [8] | Spots, breakage, melt holes, wear scars, and cracks | Color spatial distribution variance | One texture; two textures, three textures | Color spot area weight features added as a discriminator | Accuracy: 98.4%, 96.2% and 91.21% respectively |
| [9] | Cracks and spots | Fourier transform | Black Random Stripes, brown light stripes without | Without | F1-score: 92.36% and 88.66% |
| [10] | Cracks | LBP | Without | Without | Accuracy: 87.5% |
| [11] | Cracks | Morphology and wavelet transform | Without | Without | Not indicated |
| [12] | Cracks and spots | SIFT and color moments | Gray and colored textures | Without | Highest accuracy: 92.6% |
| [13] | Pits, notches, cracks, grinding | U-Net model | Without | Without | Accuracy: 94% |
| [14] | Cracks, dirt | Deep neural network | Complex textures | With a comparator | The average false detection rate: 5.8% |
| [15] | Cracks, holes | A lightweight convolutional auto-encoding network | Simple textures | Without | F1-score: 94.8% and 87.6% |
| [16] | Cracks | U-Net | Simple textures | Without | Precision: 99.9%; Recall: 78.7% |
| [17] | Cracks | YOLOv3-tiny | Simple textures | Without | Accuracy: 89.9% |

Although traditional and deep learning algorithms have accomplished substantial success in identifying and classifying surface defects of CT, they are limited to monochromatic or simple texture CTs. We can conclude that existing algorithms failed to better detect surface defects for CTs with complex textures resulting in low accuracy, and employing the discriminator to recheck the detection results can improve the defect detection accuracy from Table 1. Perhaps the main reason lies in that non-defective CTs with complex textures are more likely to be identified as defects [18]. Meanwhile, obtaining various patterns in advance is unrealistic due to the variation of CT styles, and it belongs to the open-set problem of pattern recognition. This paper proposes a CNFA for improving the detection accuracy of surface defects for CT, which was shown to have stable performances on both the validation and test sets, including uncomplicated and complex textures. It provides a theoretical and practical reference for surface defect detection of products with complex and changeable textures in industrial environments such as steel, fabrics, and leather.

The remaining part of this paper proceeds as follows: a detailed description of the proposed methodology is provided in Section 2. We first introduce the structure and flow of the entire algorithm and then explore how the comparator, detector, and discriminator...
implement their respective functions separately. Experience with the proposed methodology is described in Section 3. The results and analysis from the experiment are revealed in Section 4. A conclusion is presented in Section 5.

2. Methodology

2.1. Architecture and Workflow

In order to improve the accuracy of surface defect detection for CT, this paper proposes an algorithm that integrates comparison, detection, and discrimination, as shown in Figure 1. The detector dictates the defect detection performance of the CNFA. Improving the detector’s sensitivity to defects will bring a high false detection rate, which is unbearable in practical applications. Therefore, the accuracy of our proposed algorithm is determined by the comparator and the discriminator together. The comparator is used to extract the characteristics and compare the image to be detected with the negative sample set, and ultimately the corresponding reference image is obtained. The detector realizes the defect target detection on the detected image to acquire rectangle boxes; the discriminator simultaneously accomplishes feature extraction and similarity measurements of the derived rectangular boxes from the reference image and the one to be detected, finally evaluating whether the rectangle boxes of the image to be detected are actual defects. The entire approach process consists of the following steps:

![Figure 1. Block diagram of our algorithm. The gray line specifies that an image to be detected is directly inferred, and det_boxes surrounded by the light orange background box are obtained simultaneously by the detector. The blue line indicates the comparator extracts feature vectors of non-defective and defective images, then the most similar non-defective image retrieved is the reference image, ref_boxes surrounded by a light green background box are acquired based on the reference image and det_boxes.](image)

Step 1: Collecting non-defective tile images of different textures with the same texture template to construct a negative sample set and obtaining the feature vectors of each image in real-time based on the comparator, which is defined as \( V_{re}^j(\alpha_i) \), assume that \( \alpha_i \) is the feature dimension of the reference image, \( j \) represents the number of reference images.

Step 2: The comparator is the first used to extract the feature vectors of the detection image, named \( V_{det}(\beta_i) \), where \( \beta_i \) indicates the dimension of feature vectors is \( i \). Then, measure its distance from each feature vector in the negative sample set; the feature vector with the smallest distance corresponds to the image as the reference image.

Step 3: The category and rectangle boxes of the image to be detected(det_boxes) are obtained by the defect detector, and the rectangle boxes image are cropped. Based on the det_boxes coordinates, the length and width of the rectangle boxes are enlarged several times on the reference image, and the enlarged rectangle boxes image is obtained by cropping, the reference rectangle boxes(ref_boxes) is retrieved from the enlarged rectangle boxes image, which is the most recognizable to det_boxes.
Figure 2. Schematic diagram of a modified VGG16 network structure.

The traditional image feature extraction method is mainly based on spatial domain [6] or frequency domain transformation [7,11] and artificially designed targeted feature descriptors. Many CTs with different textured templates include dark gray, light gray, gold-plated lines, and white, etc. At the same time, each texture template will be randomly cut into different textures according to the production size specifications. Therefore, the robustness of artificially designed feature operators is very poor, and it is almost challenging to adapt to various texture detection environments. In contrast, the automatic feature extraction method based on DCNN has been proven to have good results in complex and changeable situations [19,20]. VGG16 [21] is the most representative network structure in DCNN, and it has achieved good results in characterizing abstract features in a complex background. In order to balance the calculation efficiency and accuracy, this paper adopts the improved VGG16 structure to extract the features of images for CT, as shown in Figure 2. The five convolutional pooling modules in the original VGG16 structure are retained, and the last three FC layers and soft-max layer are removed and replaced with FC1-25088 and FC2-1024. Considering the cost of model training time and feature extraction capabilities, this paper uses the pre-trained model weights based on the ImageNet-Ik data set without retraining the modified VGG16 network.

Step 4: The det_boxes and the corresponding ref_boxes are pushed into the discriminator; feature extraction and distance measurement capabilities are pretty beneficial to distinguish the degree of mutual acquaintance.

Step 5: The judgment of the true/false defect of the det_boxes is accomplished according to the prior threshold.

Retrieving the correct reference image is a prerequisite for improving the detection accuracy of the entire algorithm. In addition, another essential procedure is to extract the characteristics of different scales and shapes of the det_boxes and the corresponding ref_boxes and measure the gap to strengthen the ability to distinguish true/false defects.

2.2. Comparator

The image of CT is scale-transformed to a resolution of 224 × 224 and sent to the modified VGG16 network to obtain a 1024-dimensional feature vector. When feature vectors of the negative sample set and the image to be detected are obtained, the problem of retrieving the negative sample with the most similar texture to the detection image is transformed into two feature vector distance measures. Commonly, distance metrics include Euclidean distance, Mahalanobis distance, Bach distance, cosine distance, and Hamming distance. Considering the comparison based on multi-dimensional feature vectors, the angle between the vectors is a good representation of the difference between them. At the same time, it is necessary to ensure the invariance of spatial translation, so this paper adopts the Pearson correlation distance to measure.
Suppose that the images to be detected and the negative sample set are feature extraction and the vectors obtained are $V_{\text{det}}(\beta_i)$ and $V_{\text{re}}(\alpha_i)$, respectively, then the Pearson correlation coefficient between the two can be expressed as \[22\]:

$$\text{Corr}^{i=1, \cdots, k}(V_{\text{re}}(\alpha_i), V_{\text{det}}(\beta_i)) = \frac{\sum_{i=1}^{n} (V_{\text{re}}(\alpha_i) - \bar{V}_{\text{re}}) \cdot (V_{\text{det}}(\beta_i) - \bar{V}_{\text{det}})}{\sqrt{\sum_{i=1}^{n} (V_{\text{re}}(\alpha_i) - \bar{V}_{\text{re}})^2} \cdot \sqrt{\sum_{i=1}^{n} (V_{\text{det}}(\beta_i) - \bar{V}_{\text{det}})^2}}$$

(1)

where $n$ represents the dimension of the feature vector, $k$ represents the number of negative sample sets, $\bar{V}_{\text{re}}$ and $\bar{V}_{\text{det}}$, respectively, represent the mean value of the feature vector of the reference image and the image to be detected.

$$\bar{V}_{\text{re}} = \frac{1}{n} \sum_{i=1}^{n} \alpha_i$$

$$\bar{V}_{\text{det}} = \frac{1}{n} \sum_{i=1}^{n} \beta_i$$

From Equation (1), the distance between vectors can be obtained as:

$$\text{Dis}^{i=1, \cdots, k}(V_{\text{re}}(\alpha_i), V_{\text{det}}(\beta_i)) = 1 - \text{Corr}^{i=1, \cdots, k}(V_{\text{re}}(\alpha_i), V_{\text{det}}(\beta_i))$$

(2)

Obviously: $\text{Dis} \in [0, 2]$, and the smaller the distance between vectors, the higher the similarity; the greater the distance values, the lower the similarity. Therefore, according to Equation (2), it can be known that the corresponding image obtained by taking $\min_k(\text{Dis})$ from the set of $k$ negative samples is the reference image.

2.3. Detector

The surface defects of CT mainly include corner cracks, edge cracks, delamination, cracks, scratches, lack of glaze, pinholes, soiling, and dripping ink, etc. \[23\], which have the characteristics of diverse types, large scale spans, and significant morphological differences. Therefore, it is challenging to characterize defects based on the targeted design characteristics of traditional image processing methods, and the method of DCNN has advantages in terms of effectiveness and robustness. The detector network structure in this paper is shown in Figure 3, in which the backbone adopts the residual block structure Resnet101 \[24\], which has more vital feature characterization ability, and then a variability convolution is added to adapt to the shape and size of defects. With the help of FPN \[25\] to achieve the multi-scale fusion of the defect space, Roi-align is used to reduce the quantization error caused by channel standardization. In order to improve the positioning and classification accuracy of defects, the Cascade R-CNN \[26\] inspection head is used for the second step of regression and classification. The detector trained on the labeled dataset can detect 14 types of surface defects for CTs. The defects in the area enclosed by the red rectangle box in the detection result image are the lack of glaze. Usually, DCNN-based defect detectors have high accuracy on the training set. If applied to an unknown texture image (not trained), especially when it has a texture that is very similar to the defects of the training set, the accuracy rate of the detector will be significantly reduced, which is also the actual problem that the algorithm of this paper needs to solve.
Figure 3. Network structure diagram of the defect detector. Two red rectangle boxes produced by the detector indicates the location of defects, which is defined as lack of glaze.

At the same time, to obtain the reference rectangle, it is necessary to search on the corresponding reference image according to the coordinates of the resulting rectangle produced by the detector. Due to the deviation of angle and displacement between the image to be detected and the reference image, the first method is to find the affine transformation matrix of the two images through feature point matching and obtain the coordinates of the reference rectangle frame after the images are aligned. However, considering that the deviation of angle and displacement is relatively small and the accuracy and efficiency of feature matching must be balanced, this paper adopts a time-saving and effective method, as shown in Figure 4.

Figure 4. Schematic diagram of the reference rectangle box matching. (a,b) respectively represent the image to be detected and the reference image. In (b), det_box is shown in the red rectangle box produced by the detector, it indicates the location of the defect, which is defined as lack of glaze. In (a), the yellow false_ref_box represents the rectangle box on the reference image with the coordinates of the det_box, obviously, and false_ref_box is not the negative sample area corresponding to det_box. The green box denotes the enlarged reference box according to det_box, the blue ref_box is searched from the ref_enlarge_box.

Assume that the upper left corner of the red det_box is A, and the lower right corner is B. Obviously, if \( A = A' \) and \( B = B' \), the corresponding yellow false_ref_box obtained on the reference image is not the negative sample area corresponding to the defect. Using the midpoint of the line segment \( A'B' \) as the benchmark, enlarge the length and width by \( \beta \) times to obtain a green ref_enlarged_box with a diagonal \( A''B'' \). Use det_box as a template to search for the best matching area on ref_enlarged_box. Based on the standard correlation coefficient matching method in OpenCV3, the blue ref_box has the most significant correlation coefficient among all matches.
2.4. Discriminator

Since the detector may obtain incorrect results, removing the wrong results and keeping the correct results is the key to improving the detection accuracy of the entire algorithm, which is exactly what the discriminator can achieve. The det_boxes produced by the detector contain defects and textures, which have the characteristics of a large scale span, diverse forms, and many categories. The essence of the discriminator is to distinguish between defects and textures so that there are significant differences between them. It is necessary to solve feature mapping of defects and textures to obtain a longer feature distance from each other. Therefore, building a discriminant model based on deep metric learning [27] can achieve good results to deal with such problems. The discriminator network structure is shown in Figure 5, including the backbone, neck, and the head. The det_boxes and ref_boxes are linearly interpolated to a resolution of 224 × 224, combined with simple preprocessing, and then sent to the network simultaneously. From the comprehensive analysis of the speed and feature representation effect, the backbone is partly modified based on MobileNetV3-Large [28], the layer behind conv2d-1 × 1 is removed, and the output is a tensor of 1 × 1 × 1280. Due to the relatively small area of det_boxes and ref_boxes, too many channels will not significantly improve the feature representation ability. On the contrary, it will reduce the subsequent calculation speed. Therefore, the FC layer is used as a Neck module to map features to 512 dimensions. In order to improve the feature discrimination ability of the discriminator, head is added immediately after the neck, and the Arc-margin [29] function in metric learning is used to maximize the classification boundary in the angular space to achieve the effect of increasing the outer distance of the class and reducing the inner distance.

At the same time, the cross-entropy loss function (CE_Loss) is integrated to improve the feature separation ability to prevent the Arc-margin gradient from disappearing. Therefore, the total loss function can be expressed as:

$$L_{tot} = \alpha L_{ce} + \beta L_{arc}$$  \hspace{1cm} (3)

where $\alpha$ and $\beta$ are the weights of Arc-margin and CE_Loss, $L_{ce}$ and $L_{arc}$ [28] can be expressed as:

$$L_{ce} = -\frac{1}{N} \sum_{i=1}^{N} \log \frac{e^{W_j^T x_i + b_j}}{\sum_{j=1}^{N} e^{W_j^T x_i + b_j}}$$  \hspace{1cm} (4)

$$L_{arc} = -\frac{1}{N} \sum_{i=1}^{N} \log \frac{e^{\delta (\cos(\theta_{ij} + m))}}{e^{\delta (\cos(\theta_{ij} + m))} + \sum_{j=1, j \neq i}^{N} e^{\delta \cos \theta_j}}$$  \hspace{1cm} (5)

![Figure 5. Network structure diagram of a discriminator.](image-url)
Combined Equations (3)–(5) can be obtained:

\[
L_{\text{tot}} = -\frac{1}{N} \left( a \sum_{i=1}^{N} \log \frac{e^{W_{yi}^T x_i + b_{yi}}}{\sum_{j=1}^{n} e^{W_{yi}^T x_i + b_{yi}}} + \beta \sum_{i=1}^{N} \log \frac{e^{\alpha \cos(\theta_{yi} + m)}}{e^{\alpha \cos(\theta_{yi} + m)}} \right)
\]

where batch size and the class number are \( N \) and \( n \). \( y_i \) and \( x_i \), respectively, denote the category label of the \( i \)-th sample and the feature vector, \( x_i \) is set to 512 in this paper following. \( b_{yi} \) and \( b_{yj} \), respectively, represent the bias term. \( w_{yi} \) and \( w_{yj} \) indicate the \( y_i \)-th column and the \( j \)-th column of the weight \( W \), respectively. \( S \) is the radius of the hypersphere, \( \theta_{yi} \) and \( \theta_{yj} \) are the angles between the weights \( w_{yi} / w_{yj} \) and the feature vector \( x_i \), and \( m \) is the additional angle penalty factor between \( \theta_{yi} \) and \( x_i \). In the inference, the output of the neck part is used as the model feature output.

After \( \text{det\_boxes} \) and \( \text{ref\_boxes} \) pass through the discriminator, if it is a proper defect box, the margin of the two feature vectors in the angle space is bigger than zero; otherwise, the margin equals zero. Therefore, the angle cosine distance metric is adopted. The cosine distance of \( \text{det\_boxes} \) and \( \text{ref\_boxes} \) feature vectors \( V_{\text{det}}(\beta_i) \) and \( V_{\text{ref}}(\alpha_i) \) can be expressed as:

\[
D_{\text{cos}}(V_{\text{ref}}(\alpha_i), V_{\text{det}}(\beta_i)) = 1 - \frac{\sum_{i=1}^{n} V_{\text{ref}}(\alpha_i) \ast \sum_{i=1}^{n} V_{\text{det}}(\beta_i)}{\sqrt{\sum_{i=1}^{n} (V_{\text{ref}}(\alpha_i))^2} \ast \sqrt{\sum_{i=1}^{n} (V_{\text{det}}(\beta_i))^2}}
\]

(7)

It can be seen from Equation (7) that the smaller the distance, the higher the similarity is, and the greater the possibility of texture; the greater the distance, the lower the similarity is, and the greater the possibility of defects. Combined with threshold segmentation, the result of whether \( \text{det\_boxes} \) is a defect can be obtained.

3. Experiments

3.1. Datasets

In this paper, the comparator and the detector do not need to be trained, and the current model weights are used for inferring; only the discriminator is trained. The datasets are shown in Table 2. In order to verify the effectiveness of our algorithm, images that did not participate in the detector training were selected to form a test set to compare our algorithm with others, as shown in Table 2.

Table 2. Distribution configuration of the datasets.

| Category Item                  | Non-Defective Images | Defective Images | Reference Boxes | Defective Boxes |
|--------------------------------|----------------------|------------------|-----------------|-----------------|
| Discriminator training set     | 6000                 | 6000             | 14,852          | 14,852          |
| Discriminator validation set   | 322                  | 322              | 781             | 781             |
| Test set                       | 524                  | 524              |                 |                 |

The defect type distribution of the discriminator verification data set is shown in Table 3, including the 14 most common types of surface defects. The morphology of each type of defect is slightly different, but the scale is quite different; Table 3 lists only one form of each defect. There are a total of 50 different texture images in the training set and the validation set. The reference rectangle is marked on the standard image with the same texture as the defect. The test set comprises three images with different textures to assess our algorithm, as shown in Table 4.
Table 3. Defects distribution in the datasets of the discriminator.

| Type of Defects   | Corner Crack | Edge Crack | Collapsed Surface | White Border | Pinhole     |
|-------------------|--------------|------------|-------------------|--------------|-------------|
| legend            | ![Legend](image1) | ![Legend](image2) | ![Legend](image3) | ![Legend](image4) | ![Legend](image5) |
| Type of defects   | Crack        | Pull line  | Soiling           | Glaze bubble | Lack of glaze |
| legend            | ![Legend](image6) | ![Legend](image7) | ![Legend](image8) | ![Legend](image9) | ![Legend](image10) |
| Type of defects   | Delamination | Scratch    | dripping ink      | Cave         |
| legend            | ![Legend](image11) | ![Legend](image12) | ![Legend](image13) | ![Legend](image14) |

Table 4. Distribution configuration of the test set.

| Item SN Number | A         | B         | C         |
|----------------|-----------|-----------|-----------|
| Texture type   | Light color | Medium gray | Dark gray |
| Category Number | Defective 170 | Non-defective 170 | Defective 167 | Non-defective 167 | Defective 187 | Non-defective 187 |

3.2. Experimental Configuration

According to the data format requirements of the discriminator, the reference and defect rectangles were uniformly transformed to $224 \times 224$ resolution by linear interpolation. The defect boxes were not classified during training, and all types of defect boxes and corresponding reference ones were classified into one category. The hardware environment was configured with an Nvidia GeForce GTX 1080 8 GB graphics card, Intel Core i5-9500F CPU, and 16G memory. Paddlepaddle2.1 was employed as the deep learning framework, batch_size was set to 16, each category selected eight images, a total of 100 epochs were trained, and the initial learning rate was set to 0.0025. The learning rate decreased by 0.5 at 40, 60, and 80 epochs, respectively. The momentum optimizer was selected, and the momentum factor was set to 0.9. Based on Equation (6), we set the hyperparameters $s = 30$, $m = 0.15$, $\alpha = \beta = 1$; then loaded the pre-training parameters of MobileNetV3_large_x1_0 on ImageNet as the initial weight of the backbone. During verification and testing, the 512-D feature vector was taken from the neck part of the discriminator, calculating the cosine distance according to Equation (7) and we set a threshold equal to 0.5 to determine whether det_boxes was a defect. The comparator, detector, and discriminator were integrated to detect three kinds of CTs with different textures on the test set. The threshold of the detector ref_boxes is set to 0.95, and the threshold of defect is 0.5.

3.3. Evaluation Methodology

In the classification task, the calculation basis of each indicator comes from the classification results of positive and negative samples, which is represented by a confusion matrix, as shown in Table 5.
Table 5. Confusion matrix of evaluating indicators.

| True Results | Prediction Results |   |
|--------------|--------------------|---|
| Positive     | Positive (TP)      | Negative (FP) |
| Negative     | False Positive (FN) | True Positive (TN) |

TP: defective CTs are correctly identified as defective; FP: non-defective CTs are incorrectly identified as defective; FN: defective CTs are incorrectly identified as non-defective; TN: non-defective CTs are correctly identified as non-defective. In order to evaluate the detection effect of the discriminator and our algorithm, this article intends to use accuracy, precision, recall, and F1-score as the evaluation indicators of model performance.

4. Results and Discussion

Reference 10 utilized LBP as feature description operators for defective CT and then trained an SVM classification. HOG (Histogram of Oriented Gradients) [7] is also often employed to describe features of defects. Therefore, LBP + SVM, HOG + SVM and the defect discriminator of this article are compared on the verification set simultaneously. The discriminator parameters are configured according to Section 3.2. LBP sets eight sampling points in a radius of 2, HOG sets 32 pixels per cell unit, four cells per block unit, and the number of histogram channels is 9. The following conclusions can be drawn from Table 6: the number of defect TP is lower than the reference TP, and the number of defect FN is greater than the reference FN, indicating that the probability of misrecognizing a defect as a texture is greater than that of misrecognizing a texture as a defect. In addition, the TP value of the three methods corresponding to the defect and the reference rectangle increases with the increase in the SN, while the FN and FP decrease accordingly. The conclusion that can be drawn is that our method has the highest detection accuracy rate for the defective boxes and the lowest false detection rate.

Table 6. Validated results from three methods.

| SN | Method       | Items            | True Number | TP   | FN   | FP   | TN   |
|----|--------------|------------------|--------------|------|------|------|------|
| 1  | LBP + SVM    | Defective boxes  | 781          | 557  | 224  | 219  | 562  |
| 2  | HOG + SVM    | Defective boxes  | 781          | 639  | 142  | 81   | 700  |
| 3  | This paper   | Defective boxes  | 781          | 761  | 20   | 11   | 770  |

The evaluation indicators of the three methods can be obtained as shown in Figure 6.

(a) (b)

Figure 6. Comparison diagram of the evaluation indexes (a) and cost time (b) from the three methods.
It is evident from Figure 6a that our algorithm is superior to the LBP + SVM and HOG + SVM methods in four evaluation indicators, and its accuracy rate reaches 98.02%, which is more than 13% higher than the other two methods. In ref. [8], re-judging the detection results by the discriminator can significantly improve the defect detection accuracy of the algorithm, which shows that the detection accuracy of CTs with three different textures is increased by 6.38%, 5.24% and 1.65%, respectively. Our discriminator has a better-detailed feature representation than these three methods and achieves a good defect discrimination effect. With such a significant improvement in our algorithm’s accuracy, the misidentification of non-defective CTs as defects will be significantly reduced, saving enterprises production costs and allowing automated visual inspection technology to be used practically. In combination with the actual production situation, there may be multiple detection rectangles on each image, which takes too much time to discriminate and impedes the enterprise’s production capacity. On the verification set, the comparison results of the detection time are obtained after five rounds of testing with three methods, as shown in Figure 6b. The first method takes an average time of 10.8 ms to distinguish each rectangular box, the second method is 8.1 ms, and ours is 12 ms. Nevertheless, the increased inspection time for the production of a single tile has minimal impact on the company’s production capacity. Therefore, our method has significant advantages based on the comprehensive evaluation of the detection accuracy and efficiency of the rectangular frame discriminator.

In addition to the comparison and verification of the discriminator, the effectiveness evaluation of the entire algorithm is more valuable for practical applications. The complete algorithm integrating the comparator, detector, and discriminator is verified on the test set and compared with the results of the independent detector. The results in Table 7 indicate: as the texture number increases (complex textures), the number of FP in Method 1 increases significantly, while the number of FPs in Method 2 remains relatively stable. The two methods have the same number of TP and FN and are not affected by texture changes, which shows that these two indicators are only related to the detector. Meanwhile, the evaluation indicators of the two methods can be obtained, as shown in Figure 7. It is evident from Figure 7a that the accuracy, F1-score and accuracy of Method 2 are significantly higher than those of Method 1. In addition, the three indicators of Method 2 are not affected by texture changes and have stable results. The average accuracy rate on the three different textures has reached 98.19%; since the recall rate is only related to the detector’s performance, the recall rate indicators of the two methods are consistent.

Furthermore, a comparative experiment was carried out on time spent by the comparator, detector, and discriminator to run a single image on the test set. The results after five rounds of testing are shown in Figure 7b. The discriminator takes the least time in the whole algorithm, and the detector takes the most time. It takes 230.29 ms to complete an image detection, of which the detection takes 165.94 ms, and the comparator and discriminator take a total of 64.35 ms, which is equivalent to 38.8% of the detector time. The extra time will not affect the production capacity of the CT production line. On the contrary, this method can significantly improve the accuracy of ceramic tile defect detection.

| SN | Method               | SN | Defective | Non-Defective | TP  | FN | FP  | TN  |
|----|----------------------|----|-----------|---------------|-----|----|-----|-----|
| 1  | Independent detector | A  | 170       | 170           | 166 | 4  | 30  | 140 |
|    |                      | B  | 167       | 167           | 160 | 7  | 37  | 130 |
|    |                      | C  | 187       | 187           | 181 | 6  | 51  | 136 |
| 2  | This paper           | A  | 170       | 170           | 166 | 4  | 0   | 170 |
|    |                      | B  | 167       | 167           | 160 | 7  | 1   | 166 |
|    |                      | C  | 187       | 187           | 181 | 6  | 1   | 186 |

Table 7. The results from two methods.
In addition, the CNFA was compared with the algorithms of previous researchers, and the results are shown in Table 8. The following conclusion can be outlined: the more complex the background texture of the CTs, the lower the accuracy of the detection result. After employing the discriminator to recheck the detection results, the accuracy of the detection results is higher than that without the discriminator, regardless of whether the background texture is complex or straightforward. Among the methods that double-check the detection results with a discriminator, our CNFA achieves the best results in complex background texture, the number of defect types, and defect detection accuracy.

Table 8. Comparison with previous researchers’ results.

| No. | Number of Defects Detected | Background Texture of CTs | Defect Discrimination | Evaluation Index |
|-----|----------------------------|--------------------------|----------------------|------------------|
| [1] | 5                          | Simple textures          | Without              | Accuracy: 83%    |
| [6] | 3                          | Without                  | Without              | Accuracy: 93.4%  |
| [7] | 3                          | Without and monochromatic| Without              | Accuracy: 99.01% |
| [8] | 6                          | One texture, two textures, three textures | With | Accuracy: 98.4%, 96.2% and 91.21% |
| [9] | 2                          | Black Random Stripes, brown light stripes | Without | F1-score: 92.36% and 88.66% |
| [10]| 1                          | Without                  | Without              | Accuracy: 87.5%  |
| [12]| 2                          | Gray and colored textures| Without              | Highest accuracy: 92.6% |
| [13]| 4                          | Without                  | Without              | Accuracy: 94%    |
| [14]| 2                          | Complex textures         | With                 | Accuracy: 94.2%  |
| [15]| 2                          | Simple textures          | Without              | F1-score: 94.8% and 87.6% |
| [16]| 1                          | Simple textures          | Without              | Precision: 99.9%; Recall:78.7%, F1-score:88% |
| [17]| 1                          | Simple textures          | Without              | Accuracy: 89.9%  |
| [18]| 1                          | Simple textures          | With                 | Accuracy: 99.43% |
| The CNFA | 14                        | Complex textures        | With                 | The average accuracy: 98.19%, The average F1-score: 98.16% |

5. Conclusions

In order to improve the detection accuracy of surface defects for CT, it is necessary to deal with the feature expression and measurement of complex background textures and defects. It is an open-set detection problem because of the variety of tile textures, complex patterns, and similar features to certain defects. We have proposed an effective algorithm to improve the accuracy of ceramic tile surface defect detection. It can solve this open set problem and obtain the following conclusions:
• A CNFA for surface defect detection combining a comparator, detector, and discriminator is proposed. Collect the non-defective image to construct the negative sample, retrieve the reference image with the help of the comparator; obtain the result rectangular frame of the detection image through the detector, and obtain the corresponding reference rectangular frame. Finally, the discriminator judges the true and false defects of the resulting rectangle to improve the detection accuracy;

• Based on the modified VGG16 network, the 1024-dimensional feature vector of the detection image and the negative sample set is extracted, and the Pearson correlation coefficient is used to measure the distance between each other, and the corresponding image with the smallest search distance is the reference image;

• The detector is composed of Resnet101 + FPN + Cascade R-CNN network structure. According to the coordinates of det_boxes obtained through the detector, the corresponding box on the reference image is magnified by β times. Then, the coordinate box of the maximum correlation coefficient is obtained by the correlation coefficient matching method, which is the ref_box;

• The discriminator is composed of the modified MobileNetV3-Large as the backbone, combined with neck and head parts. The loss function based on the combination of Arc-margin and CE-Loss improves the network’s feature discrimination ability of defects and background textures. Then calculate the cosine distance of the feature vectors of det_boxes and ref_boxes; the smaller the result, the greater the texture probability, otherwise it is a defect;

• After comparing with other methods on the verification set and test set, the accuracy and efficiency of our algorithm are proven. On the verification set, the accuracy of the discriminator reached 98.02%; on the test set, the average accuracy of the entire algorithm reached 98.19%, and the detection time of a single tile only increased by 64.35 ms.

The detection algorithm proposed in this paper solves the problem of the decrease in detection accuracy caused by the change of CT’s texture in the actual scene. Distinctly, the comparator and the discriminator are the keys to ensuring accuracy. Although the algorithm has achieved good results on the verification and test sets, the FP value still has not decreased to zero (texture misidentified as a defect). The reasons may include: the comparator does not match the correct reference image, or the threshold set by the discriminator does not apply to certain types of defects; this will also be our follow-up research direction.

Author Contributions: Conceptualization, K.W. and Z.L.; methodology, K.W.; software, K.W.; validation, K.W. and X.W.; formal analysis, X.W.; investigation, X.W.; resources, K.W.; data curation, X.W.; writing—original draft preparation, K.W.; writing—review and editing, K.W., Z.L. and X.W.; visualization, K.W.; supervision, Z.L.; project administration, X.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Pretrained parameters of the modified VGG16 are available at: https://paddle-imagenet-models-name.bj.bcebos.com/ dygraph/legendary_models/VGG16_pretrained.tar (accessed on 4 December 2021). Codes of our proposed algorithm are available at: https://github.com/jackie8310/tiles-defect-detection (accessed on 4 December 2021). Pretrained parameters of the detector, the test set and discriminator validation set are available at: https://pan.baidu.com/s/1w7 1h5skj_mZBjx-21VqRA (accessed on 4 December 2021), extraction code: pvw7.

Conflicts of Interest: The authors declare no conflict of interest.
References

1. Mariyadi, B.; Fitriyani, N.; Sahroni, T.R. 2D Detection Model of Defect on the Surface of Ceramic Tile by an Artificial Neural Network. J. Phys. Conf. Ser. 2021, 1764, 012176. [CrossRef]

2. Matić, T.; Aleksić, I.; Hocenski, Ž. CPU, GPU and FPGA Implementations of MALD: Ceramic Tile Surface Defects Detection Algorithm. Automatika 2014, 55, 9–21. [CrossRef]

3. Zhao, Z.K. Review of non-destructive testing methods for defect detection of ceramics. Ceram. Int. 2021, 47, 4389–4397. [CrossRef]

4. Sioma, A. Automated Control of Surface Defects on Ceramic Tiles Using 3D Image Analysis. Materials 2020, 13, 1250. [CrossRef] [PubMed]

5. Li, D.; Niu, Z.; Peng, D. Magnetic Tile Surface Defect Detection Based on Texture Feature Clustering. J. Shanghai Jiaotong Univ. (Sci.) 2019, 24, 663–670. [CrossRef]

6. Hanzaei, S.H.; Afshar, A.; Barazandeh, F. Automatic detection and classification of the ceramic tiles’ surface defects. Pattern Recognit. 2017, 66, 174–189. [CrossRef]

7. Casagrande, L.; Macarini, L.A.B.; Bitencourt, D.; Fröhlich, A.A.; de Araujo, G.M. A new feature extraction process based on SFTA and DWT to enhance classification of ceramic tiles quality. Mach. Vis. Appl. 2020, 31, 1–15. [CrossRef]

8. Zhang, H.; Peng, L.; Yu, S.; Qu, W. Detection of Surface Defects in Ceramic Tiles with Complex Texture. IEEE Access 2021, 9, 92788–92797. [CrossRef]

9. Zorić, B.; Matić, T.; Hocenski, Ž. Classification of biscuit tiles for defect detection using Fourier transform features. ISA Trans. 2021. [CrossRef]

10. Praseto, M.D.; Rifai, M.H.; Xavieurullah, R.Y. Design of Defect Classification on Clay Tiles using Support Vector Machine (SVM). In Proceedings of the 2020 6th International Conference on Interactive Digital Media (ICIDM), Bandung, Indonesia, 14–15 December 2020; pp. 1–6. [CrossRef]

11. Zhang, J.; Zhang, H.Y.; Zhao, Y.G.; Sima, Z.W. Tile Defects Detection Based on Morphology and Wavelet Transformation. Comput. Simul. 2019, 36, 462–465. [CrossRef]

12. Li, J.H.; Quan, X.X.; Wang, Y.L. Research on Defect Detection Algorithm of Ceramic Tile Surface with Multi-feature Fusion. Comput. Eng. Appl. 2020, 56, 191–198.

13. Liu, C.; Zhang, J.; Lin, J.P. Detection and Identification of Surface Defects of Magnetic Tile Based on Neural Network. Surf. Technol. 2019, 48, 330–339.

14. Wang, K.; Li, Z. A new method to reduce the false detection rate of ceramic tile surface defects online inspection system. In Proceedings of the Tenth International Symposium on Ultrafast Phenomena and Terahertz Waves (ISUPTW 2021), Chengdu, China, 16–19 June 2021; Volume 11909, pp. 5–13. [CrossRef]

15. Li, Z.H.; Chen, X.D.; Lian, T.Q. Surface Defect Detection of Ceramic Tile Based on Convolutional AutoEncoder Network. Mod. Comput. 2021, 27, 109–114.

16. Junior, G.S.; Ferreira, J.; Millán-Arias, C.; Daniel, R.; Junior, A.C.; Fernandes, B.J. Ceramic cracks segmentation with deep learning. Appl. Sci. 2021, 11, 6017. [CrossRef]

17. Huang, W.; Zhang, C.; Wu, X.; Shen, J.; Li, Y. The detection of defects in ceramic cell phone backplane with embedded system. Measurement 2021, 181, 109598. [CrossRef]

18. Stephen, O.; Maduh, U.J.; Sain, M. A Machine Learning Method for Detection of Surface Defects on Ceramic Tiles Using Convolutional Neural Networks. Electronics 2022, 11, 55. [CrossRef]

19. Bhatt, P.M.; Malhan, R.K.; Rajendra, P.; Shah, B.C.; Thakar, S.; Yoon, Y.J.; Gupta, S.K. Image-based surface defect detection using deep learning: A review. J. Comput. Inf. Sci. Eng. 2021, 21, 040801. [CrossRef]

20. Yang, J.; Li, S.; Wang, Z.; Dong, H.; Wang, J.; Tang, S. Using deep learning to detect defects in manufacturing: A comprehensive survey and current challenges. Materials 2020, 13, 5755. [CrossRef]

21. Hammad, I.; El-Sankary, K. Impact of approximate multipliers on VGG deep learning network. IEEE Access 2018, 6, 60438–60444. [CrossRef]

22. Jebli, I.; Belouadh, E.Z.; Kabbaj, M.I.; Tilioua, A. Prediction of solar energy guided by pearson correlation using machine learning. Energy 2021, 224, 120109. [CrossRef]

23. Karimi, M.H.; Asemani, D. Surface defect detection in tiling Industries using digital image processing methods: Analysis and evaluation. ISA Trans. 2014, 53, 834–844. [CrossRef] [PubMed]

24. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [CrossRef]

25. He, K.; Girshick, R.; Dollar, P.; Girshick, S. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2961–2969. [CrossRef]

26. Cai, Z.; Vasconcelos, N. Cascade r-cnn: Delving into high quality object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 6154–6162. [CrossRef]

27. Yu, C.C.; Sa, L.B.; Ma, X.Q. Few-shot parts surface defect detection based on the metric learning. Chin. J. Sci. Instrum. 2020, 41, 214–223.
28. Howard, A.; Sandler, M.; Chu, G.; Chen, L.C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for mobilenetv3. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27–28 October 2019; pp. 1314–1324. [CrossRef]

29. Deng, J.; Guo, J.; Xue, N.; Zafeiriou, S. Arcface: Additive angular margin loss for deep face recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4690–4699. [CrossRef]