

Article

Classification Method of 3D Pattern Film Images Using MLP Based on the Optimal Widths of Histogram

Jaeun Lee , Hongseok Choi and Jongnam Kim *

Department of Artificial Intelligence Convergence, Pukyong National University, 45, Yongso-ro, Nam-gu, Busan 48513, Republic of Korea; jaeun0413@pknu.ac.kr (J.L.); alice@pukyong.ac.kr (H.C.)

* Correspondence: jongnam@pknu.ac.kr

Abstract: 3D pattern film is a film that makes a 2D pattern appear 3D depending on the amount and angle of light. However, since the 3D pattern film image was developed recently, there is no established method for classifying and verifying defective products, and there is little research in this area, making it a necessary field of study. Additionally, 3D pattern film has blurred contours, making it difficult to detect the outlines and challenging to classify. Recently, many machine learning methods have been published for analyzing product quality. However, when there is a small amount of data and most images are similar, using deep learning can easily lead to overfitting. To overcome these limitations, this study proposes a method that uses an MLP (Multilayer Perceptron) model to classify 3D pattern films into genuine and defective products. This approach entails inputting the widths derived from specific points' heights in the image histogram of the 3D pattern film into the MLP, and then classifying the product as 'good' or 'bad' using optimal hyper-parameters found through the random search method. Although the contours of the 3D pattern film are blurred, this study can detect the characteristics of 'good' and 'bad' by using the image histogram. Moreover, the proposed method has the advantage of reducing the likelihood of overfitting and achieving high accuracy, as it reflects the characteristics of a limited number of similar images and builds a simple model. In the experiment, the accuracy of the proposed method was 98.809%, demonstrating superior performance compared to other models.

Keywords: 3D pattern film image; MLP; histogram; image processing; width of histogram



Citation: Lee, J.; Choi, H.; Kim, J. Classification Method of 3D Pattern Film Images Using MLP Based on the Optimal Widths of Histogram. *Electronics* **2024**, *13*, 1114. <https://doi.org/10.3390/electronics13061114>

Academic Editors: Fernando De la Prieta Pintado and Simeone Marino

Received: 29 December 2023

Revised: 21 February 2024

Accepted: 11 March 2024

Published: 18 March 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

3D pattern film is a 2D film that appears as a 3D pattern depending on the amount and angle of light. This film was created for marketing purposes, and it is used to attract consumer attention and make products look more luxurious by attaching the film to the exterior of a product. 3D pattern film is shown in Figure 1. Figure 2 shows an example of the film's application, where the left image is without the film attached, and the right image is with the film attached. Since this film was developed recently, the distinction between good and bad 3D pattern films is currently being made by visual inspection. However, to facilitate mass production, it is necessary to establish an inspection system capable of identifying defective products.

3D pattern film requires inspection to determine whether products are good or bad based on each pattern. For this purpose, research is being continuously published on methods that can be applied to determine if the shape of the products is defective. Among the rule-based methods, various approaches have been published, including those using segmentation to detect objects and assess images, methods employing image luminance or brightness values for inspection, and techniques based on the width at specific heights in the image histogram. In research utilizing machine learning, various methods have been continuously published, including techniques that apply Canny edge detection to detect objects followed by classifying images using Support Vector Machine (SVM), methods

employing Canny edge detection followed by inspection with a convolutional neural network (CNN), approaches using few-shot models for datasets with a small number of images, and studies that classify products using deep-layer models like VGG16 and ResNet-50. However, in this paper, since the images used have a small amount of data and are similar to each other, using deep and complex deep learning models can result in a high probability of overfitting. Moreover, when dealing with a small amount of data, it is common to use pre-trained models or data augmentation, but it is challenging to always obtain a good performance with these methods. Additionally, a challenge with 3D pattern films is that their contours are not distinct, making pattern detection difficult. Furthermore, recent machine learning research papers demonstrate high performance and utilize highly complex model structures, but the experimental data mostly consist of complex image data. However, the data used in this study consists of images that are very similar each other and are simpler compared to the data used in other papers. Therefore, using complex models for such data can lead to overfitting and result in significantly lower classification accuracy.

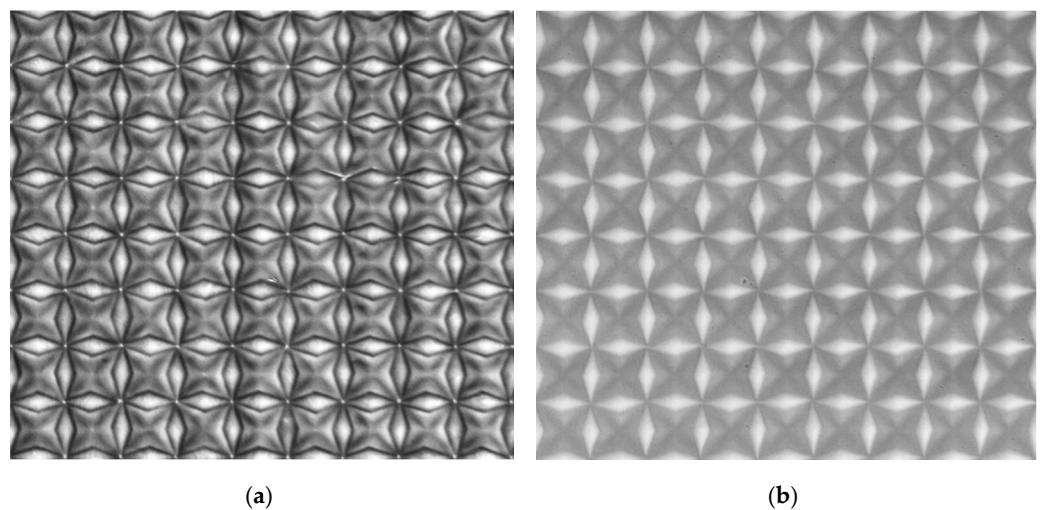


Figure 1. The 3D film pattern images. (a) Good pattern images; (b) bad pattern images.



Figure 2. Production goods without and with 3D pattern film (left: Product without 3D pattern film; right: Product with 3D pattern film).

To overcome these limitations, this paper proposes a method that includes a pre-processing step where widths at specific heights in the image histogram are calculated, as suggested by Lee et al. [1,2]. Following this, an MLP is used to classify the images. At this stage, the optimal hyper-parameters for the MLP are determined using the random search method. The proposed method, utilizing the width data obtained from the method proposed by Lee et al. [1,2], can solve the problem of detecting patterns in 3D films, which were previously difficult to segment, and can increase accuracy. Additionally, by reducing the complexity of the MLP model and finding the optimal hyper-parameters through the random search method, the probability of overfitting can be reduced.

In summary, the contributions of this study are as follows:

- 3D film data are image data for which object detection is challenging due to the faintness of the contours. To address this issue, we resolved the problem of object detection by preprocessing the 3D film image data. Specifically, we calculated the width of the bins for each interval of the image histogram, allowing us to effectively obtain information from the 3D film;
- If the method proposed in early work [1,2] is employed, the threshold values and accuracies for classifying products as ‘good’ and ‘bad’ will vary depending on the sampled data. To address these limitations, in this study, we propose a method of classifying images using deep learning instead of threshold values;
- Because of the high similarity between ‘good’ and ‘bad’ images in 3D film image data, complex structures of deep learning models can lead to overfitting and lower accuracy. In this study, we achieved high accuracy by employing a simple deep learning model that considers the characteristics of the data;
- The 3D film images used in this study are a recent development, and as there have not been many validation methods researched yet, quality verification in industrial settings still relies on manual inspection by workers. Likewise, in manufacturing and certain sectors, there is a need for data similar to those used in this study, and there is ongoing development and utilization of such data, necessitating validation methods. Therefore, this study will be helpful for quality research on similar products.

The remainder of this paper is organized as follows: Section 2 describes conventional and recently published methods for classifying good and bad images. Section 3 describes the method proposed in this paper, and Section 4 presents the experimental results obtained using the proposed method. Finally, Section 5 presents the conclusions of this study.

2. Related Works

The quality of 3D pattern film images can be inspected using rule-based methods or machine learning methods. Firstly, in the rule-based method, techniques have been proposed that classify images using the luminance or brightness of the image [1–4]. Among these methods, Lee and Kim published a study where they calculated the image histograms for 3D pattern film images, determined the width at the 10th percentile height, and then classified the images based on a threshold value [1]. This method was able to compensate for the issue of blurred contours in 3D pattern films, which makes pattern detection difficult, and, in experimental results, it demonstrated a high classification accuracy of up to 99.34%. And in the paper by [2], widths for all heights of the image histogram were calculated and compared to delineate the ranges of good and bad pattern images. However, these methods have limitations in that the classification threshold values at specific heights in the image histogram, ranging from 1/10 to 9/10, can all be different, and these threshold values can change with varying amounts of data. Michelson contrast differentiates images based on the maximum and minimum luminance values of the image [3]. The formula for this is as follows:

$$MC_i = (LV_{max} - LV_{min}) / (LV_{max} + LV_{min}), i = 1, 2, \dots, n, \quad (1)$$

where MC_i represents the Michelson contrast value of the i -th image, LV_{max} is the maximum luminance value of the image, and LV_{min} is the minimum luminance value of the image. However, as shown in Equation (1), Michelson contrast only considers the minimum and maximum luminance values of the image. Unlike the luminance values typically found in most good or bad images, there can be unusually large or small values in a minority of the luminance values. This variation can make it difficult to clearly distinguish between good and bad images.

In research on classifying the quality of images, segmentation, a method for detecting objects within images, is widely used. Segmentation refers to the process of differentiating objects in an image based on varying characteristics of pixels, such as color, texture, and brightness. Among the segmentation methods, there are techniques that allow for the

morphological detection of objects, with morphological geodesic active contour being a prominent example. Morphological geodesic active contour is a method that combines the morphology snake method [4] and the geodesic active contour method [5], and this method progressively evolves the segmented regions before morphologically segmenting objects. Recently, studies have been published utilizing morphological geodesic active contour and image processing techniques to segment and classify welding beads for evaluating the performance of robots used in welding bead manufacturing [6], as a deep learning model based on a non-parametric adaptive active contour method called fast morphological geodesic active contour (FGAC) for segmenting the left ventricle [7], for automatic segmentation of the aorta from CT images using morphological geodesic active contour [8], and for segmenting lung images based on ACM without prior training using FGAC [9]. However, morphological geodesic active contour has the limitation that it can only achieve high accuracy when the exact position of the segmentation target is accurately set.

As a traditional approach in segmentation, there is the method of edge detection, which detects objects by utilizing areas where the brightness values of pixels change abruptly. There are various types of edge detection, including Sobel edge detection, Canny edge detection, and Laplacian edge detection. Sobel edge detection detects objects by highlighting areas where the first derivative of a function exhibits significant changes. It is more robust in detecting noise compared to other edge detection methods and is more sensitive to diagonal edges than to vertical and horizontal components. Canny edge detection is a method that employs a Gaussian filter to remove noise from an image and applies thresholding twice to determine its edges. It provides sharp edges and is known for its relatively accurate detection, making it a widely used method by default. Laplacian edge detection, unlike other edge detection methods, utilizes second-order differential equations and excels at detecting edges between light and dark regions. Edge detection is typically used as part of preprocessing, and recently, research combining it with machine learning techniques has been consistently emerging [10–15]. Mlyahilu et al. proposed a method where they detected the edges of a 3D pattern film using Canny, Sobel, and Laplacian edge detection during the preprocessing stage and then classified 3D pattern images using a convolutional neural network (CNN) [10]. Salman et al. published a study in which they applied Canny edge detection to detect leaf contours and then used the Support Vector Machine (SVM) method for classification [11]. Furthermore, Jun and Jung proposed a method for inspecting the quality of Printed Circuit Board (PCB) products using a combination of the Laplacian filter and CNN methods [12]. The experimental results showed an improvement of 11.87% compared to the existing methods. Furthermore, in segmentation research, studies have proposed methods utilizing drones equipped with high-resolution proximity cameras for capturing images and then employing methods such as dual tree complex wavelet transform (DTCWT) and discrete wavelet transform (DWT) to segment and detect concrete cracks [13], detecting fires and extracting fire features using different image-processing techniques such as Canny, Sobel, and HSV transformations [14], and segmenting and detecting concrete cracks in images using edge detectors like Roberts, Prewitt, Sobel, and deep convolutional neural networks (DCNN) [15]. However, 3D pattern films pose a challenge for pattern detection due to their blurred contours. Furthermore, finding optimal hyper-parameters for data classification in deep learning models and SVM still presents a challenge.

Among machine learning models used for image data classification, popular and high-performance models include VGG16 and ResNet50 [16–21]. VGG16 is a model composed of a total of 16 convolutional layers, pooling layers, and fully connected layers, making it widely used in image recognition and classification research [16]. In their research using VGG16, Qu et al. proposed a method for detecting defects in paper using VGG16, particularly focusing on paper data with a small sample size [17]. To overcome the issue of overfitting, especially when dealing with a small dataset, the authors froze the first seven layers of VGG16 and fine-tuned the remaining convolutional layers using paper

defect images. Through this approach, they achieved a classification accuracy of 94.75% in their experimental results. Althubiti et al. presented a research study in which they developed a method for detecting defects in circuit manufacturing [18]. They converted images to the HSV color space, identified regions of interest (ROI), and used VGG16 to detect faulty products. ResNet50, another widely used model, consists of a total of 50 layers and addresses the problem of vanishing gradients in deep layers by employing a technique known as ‘residual connections’ [19]. Feng et al. published a research study in which they used ResNet50 to classify defects such as slag and scratches occurring on the surface of hot-rolled strip steel [20]. In this study, to reduce the risk of misclassification during defect classification, Feng et al. added FcaNet and Convolutional Block Attention Module (CBAM) methods to ResNet50, achieving an approximate classification accuracy of 94.85%. Additionally, Kumar and Bai presented a method using ResNet50 to detect and classify defects (cut, color, hole, thread, metal contamination) occurring during fabric production, achieving a high accuracy of 96.4% in their experimental results [21]. However, deep learning models with many layers can still experience overfitting, even when freezing some layers, especially when the dataset is small and the images are similar. Therefore, achieving high accuracy in such cases can be challenging.

Recently, research using few-shot learning has been published, using it as a method to address the challenges of using deep learning models when the dataset is small [22–24]. Few-shot learning is designed to extract and adapt as much information as possible from a small amount of data, often using transfer learning and meta-learning techniques [22]. Cao et al. proposed a method in which they fine-tuned only the parameters of the deep layers in a SqueezeNet-based model and integrated batch-size-independent Group Normalization (GN) for stable results. In their experimental results, they achieved accuracies of 97.69% and 82.92% on two different datasets, respectively [23]. Nagy and Czúni proposed a method that combines few-shot learning using the EfficientNet-B7 deep neural network with randomized classifiers [24]. In their experiments, they analyzed defect data from steel surfaces and achieved a high accuracy of over 99%. However, few-shot learning can be challenging to generalize since it uses a very small number of training samples, and obtaining a high performance in specific domains or complex problems can be difficult. Furthermore, recent deep learning models, while exhibiting high performance, tend to have highly complex structures. However, since the 3D film images used in this study are very similar to each other, using conventional deep learning models may result in a higher probability of obtaining a lower accuracy.

3. Proposed Method

The 3D pattern film images used in this study have the characteristics of blurred contours, similarity between images, and a small dataset, making them prone to overfitting when performing deep learning. However, as mentioned earlier, due to the characteristics of 3D pattern films, traditional methods such as segmentation, VGG, and ResNet still face limitations in image classification of these films. In addition, In the method proposed by Lee et al. [1], while achieving a very high classification accuracy, there was a limitation of having different classification thresholds for each 10th percentile height of the image histogram. This required the cumbersome process of finding the optimal threshold and height before performing the classification. Furthermore, for the method of Lee et al. [1], when classifying products into good and bad categories, the accuracy varies depending on the threshold value. In other words, while the threshold value may be suitable for the sampled data used in the experiment, the accuracy may vary when additional data are introduced. To address these challenges, in this study, we propose a method where we utilize the approach suggested by Lee et al. [1] as a preprocessing step. After that, we calculate the widths at specific heights for each image and then use deep learning to classify the 3D pattern film images. The proposed method’s procedure is depicted in Figure 3, and the details of step 1 are explained in Section 3.1, while step 2 is described in Section 3.2.

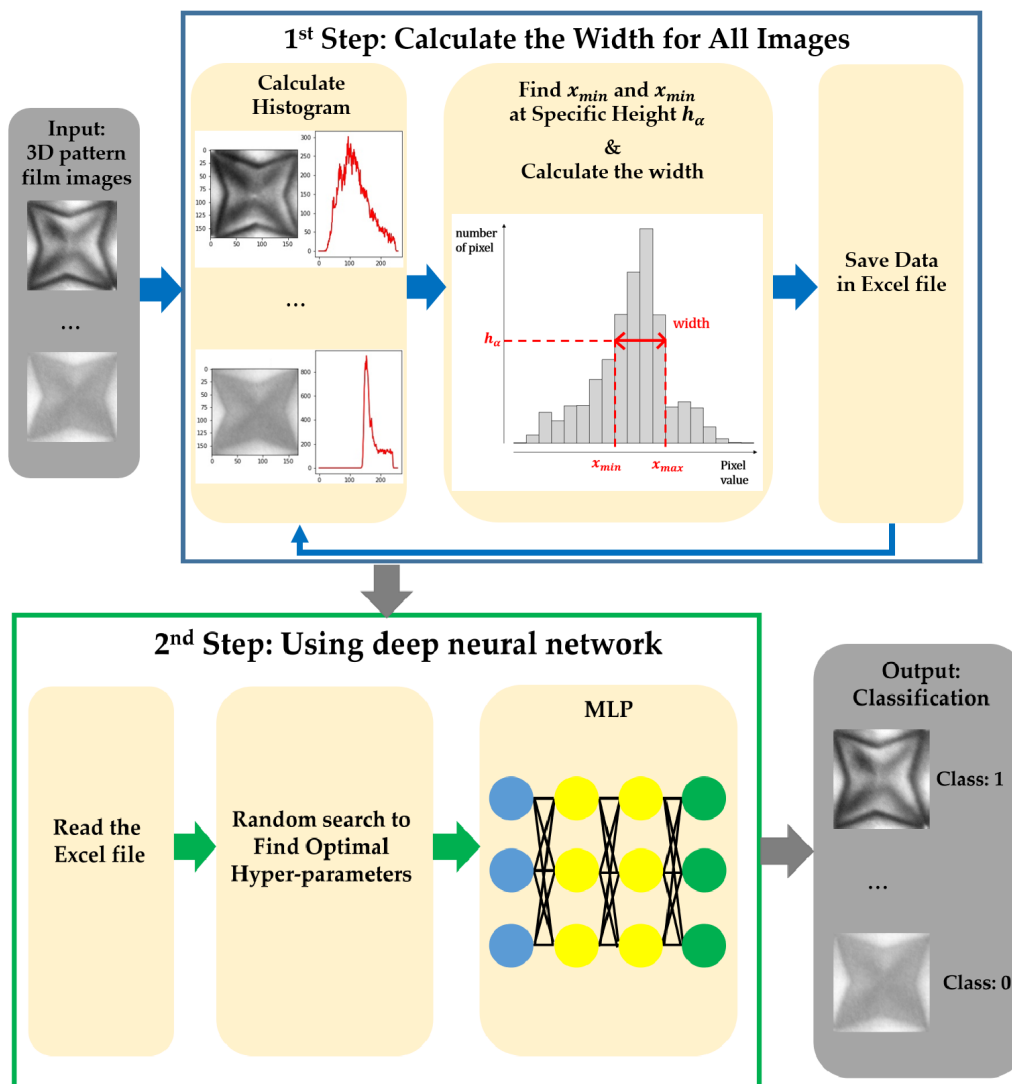


Figure 3. Procedures of the proposed method.

3.1. Calculating the Width at a Specific Height from the Image Histogram

We use the fast Fourier transform method proposed by Mlyahilu and Kim in [25] to cut images for each pattern in the 3D pattern film, as shown in Figure 2. Then, as shown in step 1 of Figure 3, we calculate the width at a specific height for each image using the method proposed by Lee et al. in [1]. In this context, the width, as described in Lee et al.'s [1] paper, carries information about whether the image is classified as 'good' or 'bad'. First, for each 3D pattern film image, we calculate the image histogram $h(b)$ representing the frequency of each pixel value. The formula is as follows:

$$h(b) = \sum_{i=1}^W \sum_{j=1}^H 1_{\{pixels|pixel_{ij}=b\}}, \quad b = 0, 1, \dots, 255, \quad (2)$$

where $h(b)$ represents the number of pixels with a brightness value b in the grayscale image, W and H are the width and height of the image, and $pixel_{ij}$ represents the value of the pixel at position (i, j) . Additionally, the term $1_{\{pixels|pixel_{ij}=b\}}$ represents an indicator function, which returns 1 if the $pixel_{ij}$ is equal to brightness value b , and 0 otherwise. After obtaining the image histogram, we calculate the heights h_α corresponding to the 10th percentile α of the image histogram as follows.

$$h_\alpha = \alpha \times \max(h(b)), \quad \alpha = \frac{1}{10}, \frac{2}{10}, \dots, \frac{9}{10}, \quad (3)$$

Then, in the image histogram, we calculate the minimum value x_{min} and maximum value x_{max} , which are the points of intersection with the x-axis at a specific height h_α , as follows.

$$x_{min} = \min(\{b|h(b) \geq h_\alpha, b = 0, 1, \dots, 255\}), \quad (4)$$

$$x_{max} = \max(\{b|h(b) \geq h_\alpha, b = 0, 1, \dots, 255\}), \quad (5)$$

Using the previously obtained minimum value x_{min} and maximum value x_{max} on the x-axis, we calculate the width w_α at a specific height h_α as given by the following equation.

$$w_\alpha = x_{max} - x_{min}, \quad (6)$$

We calculate and store the widths of the image histograms at heights ranging from 1/10 to 9/10 for each 3D pattern film image.

3.2. 3D Pattern Film Image Classification Using MLP

In the second step, we perform a classification using the MLP $f(w_\alpha)$ on the width values at specific heights h_α obtained from the previously calculated image histogram.

$$f(w_\alpha) = f_L(\dots f_2(f_1(v, w_\alpha))), \quad (7)$$

where v represents weights and L is the number of hidden layers. In deep learning, there is a necessary process of setting hyperparameters, which are parameters that determine the configuration of the method. Hyperparameters have a significant impact on the method's performance and learning capability, so finding the optimal values is important. However, since they are not automatically determined by the training data, users either manually set them or use hyperparameter optimization techniques to find the optimal values. In this paper, we use the random search optimization technique, which is well-known for efficiently exploring hyperparameter space within a given time frame [26]. Among the hyperparameters, we search for optimal values related to the number of hidden layers, the number of hidden nodes, and the learning rate. This is because the 3D pattern film images used in this study have a small amount of data and are similar to each other, which can lead to the problem of overfitting. Therefore, setting an appropriate model complexity is crucial, which necessitates finding the optimal number of hidden layers and hidden nodes. Additionally, in defect detection tasks, processing speed is important in addition to accuracy, which is why we search for the optimal learning rate value. For the random search method, we set the ranges for hyperparameters as follows: the number of hidden layers $L \sim U(2, 4)$, the number of hidden nodes $n \in \{64, 128, 256\}$, and the learning rate $\rho \in \{0.1, 0.01, 0.001\}$ to search for. The range for the number of hidden layers and the number of hidden nodes was set to values that allow the model to be sufficiently complex while still being able to learn effectively. The learning rate was chosen from commonly used values. After finding the optimal values for the number of hidden layers, number of hidden nodes, and learning rate through the random search method, these values are then applied to the MLP. The MLP is used to input 3D pattern film images and classify whether each image is 'good' or 'bad'.

4. Experimental Results

To evaluate the performance of the proposed method in this paper, we conducted an analysis using 3D pattern film, as shown in Figure 2. As shown in Figure 2, the 3D pattern film consists of multiple patterns printed on a large film. We used the fast Fourier transform method to cut images for each pattern, and the results are depicted in Figure 4. Therefore, the total number of image data is 2850, with 2136 being 'good' 3D pattern film images and 714 being 'bad' 3D pattern film images. To perform deep learning, we divided the data into training and test sets in an 8:2 ratio. In this split, the training data consisted of 1710 'good' images and 570 'bad' images, while the test data included 426 'good' images and 144 'bad' images.

images for experimentation. The PC specifications were as follows: Windows 10 Pro, Intel® Core™ i7010700k CPU@3.80GHz, NVIDIA GeForce RTX 2080 SUPER, 16GB, and Python 3.6.

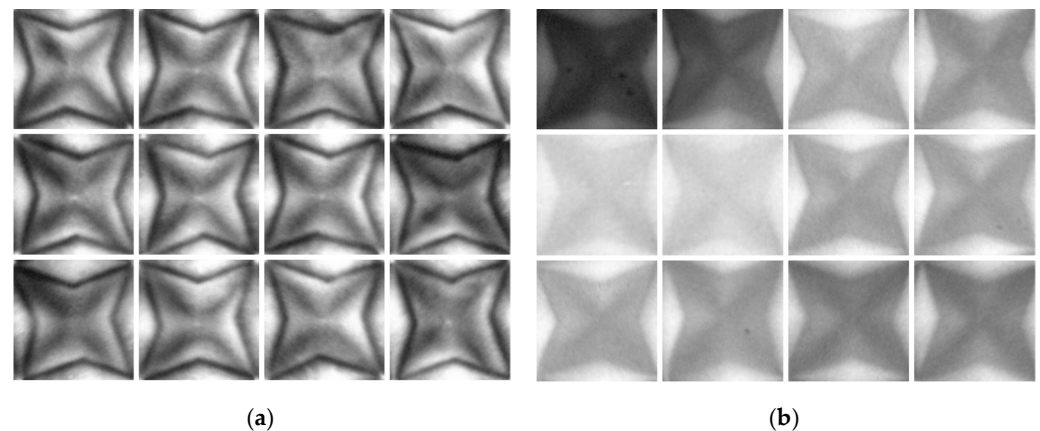


Figure 4. Cropped 3D film pattern images. (a) Good pattern images; (b) bad pattern images.

First, we performed preprocessing using the method proposed by Lee et al. [1]. In the preprocessing step, we calculated the image histograms for the cropped 3D pattern film images, as shown in Figure 5. Then, for each corresponding image histogram, we determined the minimum and maximum values on the x-axis that intersected with the height of 1/10 and used these values to calculate the width. In the same way, we calculated the widths of the image histograms for heights ranging from 2/10 to 9/10. After calculating the heights for all images from 1/10 to 9/10, we used deep learning to perform the classification. We utilized the MLP in deep learning, and to find the optimal hyperparameters used in the MLP method, we conducted a random search. The hyperparameters experimented with in the random search method included the number of hidden layers, the number of nodes in the hidden layers, and the learning rate. In this case, the ranges for each hyperparameter were: from two to four hidden layers, with each hidden layer having 64, 128, or 256 nodes, and learning rates of 0.001, 0.01, and 0.1. In addition to the mentioned hyperparameters, for MLP, the dropout was set to 0.1, the activation function for hidden layers was relu, the regularization was $L_2(\lambda = 0.01)$, the loss function was binary cross-entropy, epochs were set to 20, the batch size was 100, and five-fold cross-validation was used; all were kept constant for the experiments. As shown in Table 1, a total of 10 fittings were performed in the random search, and all but one achieved an accuracy of over 97%. The third experiment showed the highest accuracy of 99.3%, with the configuration of three hidden layers, with node counts of 64, 256, and 128 for each hidden layer, and a learning rate of 0.001. In addition, both the recall and precision were 99.5%. The total computation time was 3250.7 s, with 3240.2 s spent on preprocessing and 10.5 s spent on analysis using the MLP method.

Table 1. Classification results for methods with 3D film images (H.L: hidden layers, H.N: hidden nodes).

Num. of Exp.	Num. of H.L	Num. of H.N (H.L-1)	Num. of H.N (H.L-2)	Num. of H.N (H.L-3)	Num. of H.N (H.L-4)	Learning Rate	Accuracy	Recall	Precision
1	2	256	128			0.01	0.986	0.988	0.993
2	2	256	64			0.01	0.987	0.993	0.990
3	3	64	256	128		0.001	0.993	0.995	0.995
4	2	128	128			0.001	0.989	1.000	0.986
5	3	64	256	256		0.001	0.991	1.000	0.988
6	4	64	64	64	64	0.001	0.989	1.000	0.986
7	3	64	128	256		0.01	0.989	1.000	0.986

Table 1. Cont.

Num. of Exp.	Num. of H.L	Num. of H.N (H.L-1)	Num. of H.N (H.L-2)	Num. of H.N (H.L-3)	Num. of H.N (H.L-4)	Learning Rate	Accuracy	Recall	Precision
8	4	64	256	64	256	0.1	0.758	1.000	0.758
9	2	256	64			0.1	0.980	0.974	0.100
10	2	256	256			0.01	0.989	1.000	0.986

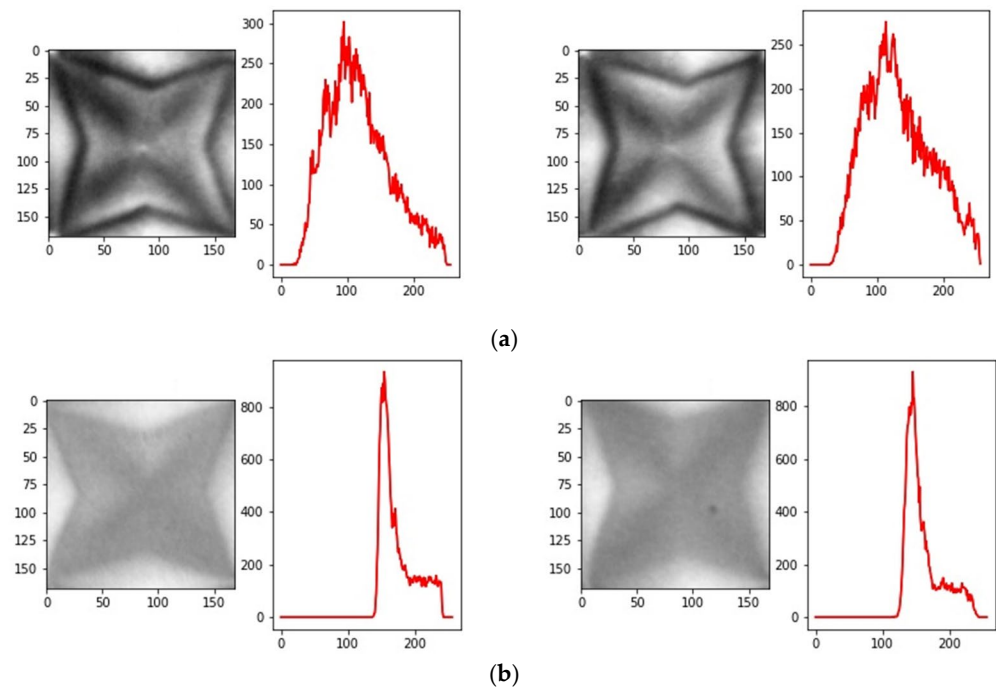


Figure 5. Image histogram (x -axis: pixel value; y -axis: frequency). (a) Good pattern images; (b) bad pattern images.

To evaluate the performance of the proposed method, comparative experiments were conducted. In the comparative experiments, the following methods were used: Michelson contrast [3], morphological geodesic active contour [6], CNN with Canny [10], SVM with Canny [11], few-shot (five-shot), VGG16, and ResNet50. In general, it is known that image data achieve a better performance when using a CNN compared to the MLP method. Therefore, CNN models were used in the comparison method. Furthermore, Michelson contrast is a preprocessing method that is similar to the one used in this study, and morphological geodesic active contour is capable of detecting the desired objects morphologically among segmentation methods. Therefore, we compared their performance in the experiments. Among these methods, the Michelson contrast and morphological geodesic active contour used the similarity index (SSIM) for classifying good and bad images after analysis [27]. The SSIM evaluates the similarity between two images using their structural information, enabling a comparison of the pixel structures that make up the images. The formula for the SSIM is as follows:

$$S(x, y) = s_1 \cdot s_2 \cdot s_3 = 4\mu_x \mu_y \sigma_{xy} / \left\{ \left(\mu_x^2 + \mu_y^2 \right) \left(\sigma_x^2 + \sigma_y^2 \right) \right\}, \quad (8)$$

where s_1 , s_2 , and s_3 represent the average brightness, contrast, and correlation of the two images, respectively. For Michelson contrast and morphological geodesic active contour, we set the SSIM thresholds at 0.5, respectively, to assess the similarity between two images. In this context, a higher SSIM value indicates greater similarity between the two images, while a lower SSIM value indicates lower similarity. For the CNN with Canny [10], we followed the same configuration as in the referenced paper, with two hidden layers and node counts

of 32 and 64 for each hidden layer. For few-shot, we conducted experiments using a five-shot approach. Additionally, in the case of the pre-trained models VGG16 and ResNet50, we augmented the data to increase its quantity before conducting the experiments. The reason for this is that both models have complex architectures, while the amount of data used in the experiments is relatively small, which could lead to overfitting.

The experiments were conducted based on defined criteria for comparison, and the results are presented in Table 2 as average accuracy, average recall, and average precision. The confidence intervals were calculated using 2000 bootstrap replicates. In the comparative experiment results, the proposed method achieved an average accuracy of 99.3%, followed by SVM with Canny at 99.0%, VGG16 at 97.3%, ResNet50 at 83.3%, CNN with Canny at 75.3%, Morphological geodesic active contour at 74.9%, few-shot at 72.8%, and Michelson contrast at 67.6%. While VGG16 and ResNet50 are known for their good performance, it was observed that their performance is relatively lower for datasets with few data and similar-shaped images, as in the experiments conducted in this study. Additionally, few-shot, which is known as a deep learning model for use with a small number of data, exhibited an accuracy of 72.8%, which is lower than the 99.3% achieved by the proposed method. The computational times for each method were as follows: CNN with Canny took 58.0 s, Michelson contrast took 107.2 s, morphological geodesic active contour took 1931.1 s, and ResNet50 took 2882.9 s, all faster than the proposed method. However, their accuracy values were all more than 10% lower than that of the proposed method. On the other hand, SVM with Canny had a similar accuracy to the proposed method but had a high computational cost of 5238.0 s. Additionally, few-shot and VGG16 had significantly longer computation times of 26,710.9 s and 27,264.2 s, respectively. Therefore, in the comparative experiments, the proposed method had the highest accuracy, and among models that achieved an accuracy of over 90%, the proposed method also had the fastest computational time.

Table 2. Classification results for methods with 3D film images.

Method	Accuracy (95% C.I)	Recall	Precision	Time (s)
		(95% C.I)	(95% C.I)	
Proposed method	0.993 (0.986–0.995)	0.995 (0.971–0.999)	0.995 (0.976–0.999)	3250.7
Michelson contrast [3]	0.676 (0.614–0.716)	0.575 (0.529–0.616)	0.987 (0.969–0.995)	107.2
Morphological geodesic active contour [6]	0.749 (0.697–0.801)	0.691 (0.647–0.749)	0.964 (0.945–0.986)	1931.1
CNN with Canny [10]	0.753 (0.698–0.796)	0.997 (0.959–1.000)	0.748 (0.704–0.771)	58.0
SVM with Canny [11]	0.990 (0.935–0.996)	0.988 (0.973–0.991)	0.995 (0.990–0.999)	5238.0
Few-shot(5shot)	0.728 (0.652–0.790)	0.676 (0.607–0.738)	0.719 (0.651–0.782)	26,710.9
VGG16	0.973 (0.958–0.987)	0.936 (0.906–0.967)	0.938 (0.910–0.967)	27,264.2
ResNet50	0.833 (0.734–0.891)	0.881 (0.818–0.943)	0.883 (0.820–0.945)	2882.9

5. Conclusions and Discussion

In this paper, we proposed a classification method for inspecting 3D pattern film images. We employed a preprocessing step, where we calculated histogram widths at specific heights in the image histograms. Then, we used an MLP model to analyze the width data. In this process, we employed the random search method to find the optimal hyperparameters for the number of hidden layers, the number of nodes in each hidden layer, and the learning rate, which were used to build the MLP model. In the proposed method in this paper, we addressed the limitation of blurry contours in 3D pattern film

images by using pixel histograms specific to each image in the preprocessing stage, thereby improving the accuracy of the analysis results. Furthermore, in the proposed method, we mitigated overfitting by constructing a simple MLP model for data with low sample sizes and similar image characteristics. In the experiments, the proposed method achieved an accuracy of 99.30%, which was the highest among the models tested. Comparatively, models with complex structures like VGG-16, ResNet50, CNN with Canny, and SVM with Canny had accuracies of 97.3%, 83.3%, 75.3%, and 99.0%, respectively, which were lower than that of the proposed method. The analysis results from the few-shot method, a method commonly used when data are scarce, also showed an accuracy of 72.8%, which was lower than the performance of the proposed method. Furthermore, the Michelson contrast method achieved an accuracy of 67.6%, while the remaining methods showed accuracies in the 70% range. Therefore, this paper demonstrated that, by using relatively simple deep learning models tailored to the characteristics of the data, it is possible to achieve a good performance.

The 3D film images used in this paper, as mentioned earlier, are a recently developed product that is continuously evolving. In addition, it is likely that other images currently under development will have similar characteristics to the images used in this study. Therefore, the results of this study could serve as foundational research for validation not only for newly developed 3D film images but also for similar products. In future research, we plan to conduct further studies to improve the preprocessing stage. The computation time for the MLP model used in the proposed method was approximately 10 s, but the preprocessing stage consumed most of the computation time. Therefore, we intend to conduct research which aims to improve the preprocessing stage to increase accuracy while reducing computation time.

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

Funding: This research was funded by the Small and Medium Business Technology Innovation Development Project from TIPA, grant number 00220304 and 00278083.

Data Availability Statement: Data is available at https://github.com/icml2410/3D_film_images (accessed on 1 March 2024).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Lee, J.; Kim, J. Three-Dimensional Film Image Classification Using an Optimal Width of Histogram. *Appl. Sci.* **2023**, *13*, 4949. [CrossRef]
2. Lee, J.; Choi, H.; Kim, J. Classification Algorithm of 3D Pattern Film Using the Optimal Widths of a Histogram. *Electronics* **2023**, *12*, 4139. [CrossRef]
3. Rao, B.S. Dynamic histogram equalization for contrast enhancement for digital images. *Appl. Soft Comput.* **2020**, *89*, 106114. [CrossRef]
4. Papolu, J.S.; Prasad, M.B.; Vasavi, S.; Geetha, G. A Framework for Sea Breeze Front Detection from Coastal Regions of India Using Morphological Snake Algorithm. *ECS Trans.* **2022**, *107*, 585. [CrossRef]
5. Wu, X.; Tan, G.; Pu, B.; Duan, M.; Cai, W. DH-GAC: Deep hierarchical context fusion network with modified geodesic active contour for multiple neurofibromatosis segmentation. *Neural Comput. Appl.* **2022**, 1–16. [CrossRef]
6. Mlyahilu, J.N.; Mlyahilu, J.N.; Lee, J.E.; Kim, Y.B.; Kim, J.N. Morphological geodesic active contour algorithm for the segmentation of the histogram-equalized welding bead image edges. *IET Image Process.* **2022**, *16*, 2680–2696. [CrossRef]
7. Medeiros, A.G.; Silva, F.H.; Ohata, E.F.; Peixoto, S.A.; Filho, P.P.R. An automatic left ventricle segmentation on echocardiogram exams via morphological geodesic active contour with adaptive external energy. *J. Artif. Intell.* **2019**, *1*, 77–95. [CrossRef]
8. Dasgupta, A.; Mukhopadhyay, S.; Mehre, S.A.; Bhattacharyya, P. Morphological Geodesic Active Contour Based Automatic Aorta Segmentation in Thoracic CT Images. In Proceedings of the International Conference on Computer Vision and Image Processing: CVIP 2016, Roorkee, India, 26–28 February 2016; pp. 187–195.

9. Medeiros, A.G.; Guimarães, M.T.; Peixoto, S.A.; Santos, L.D.O.; da Silva Barros, A.C.; De Souza Rebouças, E.; de Albuquerque, V.H.C.; Rebouças Filho, P.P. A new fast morphological geodesic active contour method for lung CT image segmentation. *Measurement* **2019**, *148*, 106687. [\[CrossRef\]](#)
10. Mlyahilu, J.; Kim, Y.; Kim, J. Classification of 3D Film Patterns with Deep Learning. *Comput. Commun.* **2019**, *7*, 158–165. [\[CrossRef\]](#)
11. Salman, A.; Semwal, A.; Bhatt, U.; Thakkar, V.M. Leaf Classification and Identification using Canny Edge Detector and SVM Classifier. In Proceedings of the 2017 International Conference on Inventive Systems and Control (ICISC), Coimbatore, India, 19–20 January 2017; pp. 1–4.
12. Jun, H.; Jung, I.Y. Enhancement of Product-Inspection Accuracy Using Convolutional Neural Network and Laplacian Filter to Automate Industrial Manufacturing Processes. *Electronics* **2023**, *12*, 3795. [\[CrossRef\]](#)
13. Dixit, A.; Wagatsuma, H. Investigating the effectiveness of the Sobel operator in the MCA-based automatic crack detection. In Proceedings of the 2018 4th International Conference on Optimization and Applications (ICOA), Mohammedia, Morocco, 26–27 April 2018; pp. 1–6.
14. Malbog, M.A.F.; Lacatan, L.L.; Dellosa, R.M.; Austria, Y.D.; Cunanan, C.F. Edge detection comparison of hybrid feature extraction for combustible fire segmentation: A Canny vs Sobel performance analysis. In Proceedings of the 2020 11th IEEE Control and System Graduate Research Colloquium (ICSGRC), Shah Alam, Malaysia, 8 August 2020; pp. 318–322.
15. Dorafshan, S.; Thomas, R.J.; Maguire, M. Comparison of deep convolutional neural networks and edge detectors for image-based crack detection in concrete. *Constr. Build. Mater.* **2018**, *186*, 1031–1045. [\[CrossRef\]](#)
16. Mascarenhas, S.; Agarwal, M. A comparison between VGG16, VGG19 and ResNet50 architecture frameworks for Image Classification. In Proceedings of the 2021 International conference on disruptive technologies for multi-disciplinary research and applications (CENTCON), Bengaluru, India, 19–21 November 2021; pp. 96–99.
17. Qu, Y.H.; Tang, W.; Feng, B. Paper defects classification based on VGG16 and transfer learning. *J. Korea TAPPI* **2021**, *53*, 5–14. [\[CrossRef\]](#)
18. Althubiti, S.A.; Alenezi, F.; Shitharth, S.K.S.; Reddy, C.V.S. Circuit manufacturing defect detection using VGG16 convolutional neural networks. *Wirel. Commun. Mob. Comput.* **2022**, *2022*, 1070405. [\[CrossRef\]](#)
19. Theckedath, D.; Sedamkar, R.R. Detecting affect states using VGG16, ResNet50 and SE-ResNet50 networks. *SN Comp. Sci.* **2020**, *1*, 79. [\[CrossRef\]](#)
20. Feng, X.; Gao, X.; Luo, L. A ResNet50-based method for classifying surface defects in hot-rolled strip steel. *Mathematics* **2021**, *9*, 2359. [\[CrossRef\]](#)
21. Kumar, K.S.; Bai, M.R. ResNet50: Automated Fabric Defect Detection and Classification based on a Deep Learning Approach. *Tuijin Jishu/J. Propuls.* **2023**, *44*, 2156–2165.
22. Wang, Y.; Yao, Q.; Kwok, J.T.; Ni, L.M. Generalizing from a few examples: A survey on few-shot learning. *ACM Comput. Surv.* **2020**, *53*, 1–34. [\[CrossRef\]](#)
23. Cao, Y.; Zhu, W.; Yang, J.; Fu, G.; Lin, D.; Cao, Y. An effective industrial defect classification method under the few-shot setting via two-stream training. *Opt. Lasers Eng.* **2023**, *161*, 107294. [\[CrossRef\]](#)
24. Nagy, A.M.; Czúni, L. Classification and fast few-shot learning of steel surface defects with randomized network. *Appl. Sci.* **2022**, *12*, 3967. [\[CrossRef\]](#)
25. Mlyahilu, J.; Kim, J. A Fast Fourier Transform with Brute Force Algorithm for Detection and Localization of White Points on 3D Film Pattern Images. *J. Imaging Sci. Technol.* **2021**, *66*, 030506. [\[CrossRef\]](#)
26. Bergstra, J.; Bengio, Y. Random search for hyper-parameter optimization. *J. Mach. Learn. Res.* **2012**, *13*, 281–305.
27. Setiadi, D.R.I.M. PSNR vs. SSIM: Imperceptibility quality assessment for image steganography. *Multimed. Tools Appl.* **2021**, *80*, 8423–8444. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.