Automated Fabric Defect Detection and Classification: A Deep Learning Approach

NC Sandhya, Nihal Mathew Sashikumar, M Priyanka, Sebastian Maria Wenisch, Kunaraj Kumarasamy

How to cite: Sandhya NC, Sashikumar NM, Priyanka M, Wenisch SM, Kumarasamy K. Automated Fabric Defect Detection and Classification: A Deep Learning Approach. Textile & Leather Review. 2021; 4:315-335. <https://doi.org/10.31881/TLR.2021.24>

How to link: <https://doi.org/10.31881/TLR.2021.24>

Published: 14 December 2021

Automated Fabric Defect Detection and Classification: A Deep Learning Approach

**NC SANDHYA, Nihal Mathew SASHIKUMAR, M PRIYANKA, Sebastian Maria WENISCH,
Kunaraj KUMARASAMY***

Loyola-ICAM College of Engineering and Technology (LICET), Loyola Campus, Chennai-600034, India

*k.kunaraj@gmail.com

Article

UDC 677.019:004

DOI 10.31881/TLR.2021.24

Received 26 July 2021; Accepted 28 October 2021; Published 14 December 2021

ABSTRACT

A computer-based intelligent visual inspection system plays a major role in evaluating the quality of textile fabrics and its demand is continuously increasing in the textile industry, especially when the quality of textile is to be considered. In this paper, we propose an AI-based automated fabric defect detection algorithm which utilizes pre-trained deep neural network models for classifying possible fabric defects. The fabric images are enhanced by pre-processing at various levels using conventional image processing techniques and they are used to train the networks. The Deep Convolutional Neural Network (DCNN) and a pre-trained network, AlexNet, are used to train and classify various fabric defects. With the exiting textile dataset, a maximum classification accuracy of 92.60% is achieved in the conducted simulations. With this accuracy, the detection and classification system based on this classifier model can aid the human to find faults in the fabric manufacturing unit.

KEYWORDS

fabric defects, artificial intelligence, defect classifier, AlexNet, deep neural network

INTRODUCTION

In the textile industry, fabrics are prone to various defects and deformities which have an obstructive effect on the quality of the product. The defects are caused by the misuse of the materials and carelessness while manufacturing. It is challenging to inspect the real fabric defects manually, due to the huge number and various categories of defects which are characterized by uncertainty. A low percentage of the defects are being detected by a manual inspection due to human fatigue, which decreases the efficiency, whereas in a real-time automatic system defect detection could be more efficient.

The textile industry, like any other industry, is concerned with quality. As we all know, industries generally aim to manufacture the highest standard of goods in the stipulated time. Fabric faults make-up approximately 85% of the defects in the textile industry and perhaps the manufacturers obtain only

45% to 65% of their profits from reselling or from off-quality goods [1]. Statistics (as of January 2020) published by NITI Aayog revealed that India is the second-largest manufacturer of textiles and clothing and the second-largest exporter of textiles and apparel [2]. The major concerns are the contamination and the poor quality of the fibre. India uses old technology even though it has the largest installed production bases. Certainly, by adopting state-of-the-art technologies related to the textile industry, competitive manufacturing can be done.

Currently, fabric defects are being detected manually by trained inspectors, which has its own share of issues. Firstly, inspectors who have been deployed must be skilled and experienced. Also, staring at the fabric for prolonged periods of time strains one's eyes, which in turn takes a toll on the accuracy with which faults can be identified. Hence, in order to detect, identify and prevent these defects from reoccurring there is a growing demand for an automated fabric fault detection system in the industry. Common fabric defects include holes, scratches, stretching, loose yarn, dirty spots, cracked points, misprints, colour bleeding etc. [3]. Apparently, using computer vision (CV) to address this problem is a good solution and we employ an artificial intelligence (AI) - based model to classify possible defects such as colour, cut, holes, metal contamination and thread. A machine learning/deep learning model is trained with a dataset consisting of at least 500,000 images.

Section 1 introduces the current scenario, section 2 highlights the application of AI in the textile industry and section 3 reviews broadly the works related to textile defect detection and classification. Section 4 discusses the dataset and the methodology is discussed in section 5. Various image pre-processing techniques used in this experiment are discussed in section 6, while the network architecture for training and classification is discussed in section 7. The experimental results and its evaluation is discussed in section 8 and section 9 provides the concluding remarks with the outlook for extension of this work.

AI IN TEXTILE

In the textile industry, AI is revolutionizing the total production process and it is the need of the hour since there is an inflated demand for quality textiles. In the past decade there has been a significant leap in the number of industries using AI because both production costs and the number of faults are kept low without compromising speed and accuracy. Fabric defects deteriorate the value of textile products. The final product with a single minuscule defect can be easily rejected. Neural network (NN) with deep learning plays a vital role in inspecting and identifying defects at a much faster rate and with better accuracy. This new era of textile industry leveraged with AI brings cutting-edge revolution and has a great future ahead.

RELATED WORKS

A variety of works has been carried out in detecting fabric defects since the 1980s. Several approaches used to classify fabric defects generally fall into one of the following categories i.e., spectral, structural, model-based, statistical, learning, hybrid or comparative studies [4]. The following sections, 3.1 to 3.3, present a brief overview of these commonly used approaches.

Spectral-based approaches

The spectral-based approaches exploit the fact that faultless fabrics exhibit periodic property and the occurrence of faults leads to aperiodicity. Upon translating the images into frequencies, faults become clearly visible as high frequency components. Chi-Ho Chan and G. K. H. Pang proposed the central spatial frequency spectrum-based approach that relies on the Fourier analysis to understand the fabric structure by mapping the image space and frequency space representations [5]. A fast Fourier transform was used to find the faults in the fabric images consisting of double yarn, missing yarn and web. A. Bodnarova et al. treated fabric defect detection like a semi-supervised segmentation problem and the Gabor filter was applied twice: for obtaining the feature matrix for the first time and for smoothening the images [6]. The optimal Gabor filter is obtained by tuning the fisher function and the results were said to have low false alarm rates. Le Tong et al. proposed a novel algorithm for Gabor filter optimization using composite differential evolution (CoDE) together with fusion and threshold processes [7]. In the optimization phase, decision vectors are encoded, followed by population initialization and finally the mutation and crossover processes. In spite of having the advantage of being less sensitive to noise, the spectral-based approaches are outdated owing to the lack of prominence in the results obtained

Deep learning-based approaches

With the emergence of AI, the application of machine learning and deep learning-based approaches for anomaly detection has been increasing drastically and fabric defect detection is no exception. Jun-Feng Jing et al. presented the DCNN model which has initially been trained by using the MNIST dataset and applied transfer learning by using LeNet-5, AlexNet and VGG16 models [8]. The model was trained by using local patches of the images and whole images were used in the testing phase. The trained model is said to have been evaluated using two public datasets: TILDA and Guangdong Esquel Textiles and a self-made dataset fabric database derived from Henry Y. T. Ngan (Industrial Automation Research Laboratory) and the average accuracy obtained was 97.31%. Xiang Jun et al. proposed a learning-based framework that employs local defect prediction and global defect recognition [9]. The images in the TILDA dataset were first cropped to appropriate the dimensions and enhanced by the

histogram equalization. The inception V1 model was then incorporated to detect defects locally and the LeNet5 model was used to recognize the feature map and thus predict the defect.

Zhoufeng Liu et al. optimized deep neural networks and used the Xiamen Face++ Company dataset consisting of 8000 fabric images for training the VGG16 model [10]. In order to reduce the memory consumption and the number of parameters involved, an additional layer, called deconvolutional network layer, was introduced. This layer served the purpose of projecting the feature activation back to the input pixel space, so that the input pattern that caused a specific activation in the feature maps can be identified.

Hybrid approaches

Guangzhong Cao et al. focused on detecting large and complex surface defects [11]. Here, the texture of the material under study, in test and reference product images, is divided into background and special pattern areas. Defect in the background is detected by using threshold segmentation, whereas the defects in the special pattern area are detected by applying image registration and image differentiation techniques.

Yundong Li and Cheng Zhang used a hybrid approach for detecting defects in warp-knitted fabrics that had two phases, involving image enhancement and image segmentation [12]. The former was carried out using Gabor filters with the aim of making the defects clearly visible, whereas the latter was implemented using an adaptive pulse coded neural network (PCNN). The results from these two stages underwent morphology filtering before yielding the final output.

Hermanus Vermaak et al. proposed a hybrid method for fabric defect detection using dual-tree complex wavelet transform (DTCWT) [13]. The DTCWT was applied for wavelet decomposition of the fabric images and computing features like mean energy, mean magnitude and variance of magnitudes of the wavelet coefficients. The features obtained for defective and non-defective in the train set from the previous steps went through an Euclidean Distance Classifier, which compares the test and train sets to predict whether the given sample is defective or not.

DATASET

The machine learning model presented in this paper has been trained with a dataset taken from the public dataset - MVTec Anomaly Detection [14]. Our dataset consists of a whopping 540,000 images having five defect classes i.e., colour, cut, hole, metal contamination, thread and the non-defective class (here, good) and is divided into test and train sets having 180,000 and 360,000 images respectively.

Figure 1 illustrates the sample image patches used in the training set that has 72,000 images and is equally divided into 6 classes, with each class having 12,000 images.

Figure 1. Sample image patches of each class

Figure 2 shows the real time samples of various defects.

Figure 2. Real time image samples of defects

METHODOLOGY

The collected fabric dataset is initially checked for any corrupted pixels and it is further pre-processed [15,16]. Figure 3 shows the block diagram of the proposed fabric defect detection system. The pre-processing techniques, like image scaling, pixel normalization and dimension reduction, are used for the training dataset. Feature extraction by the CNN model aids in characterizing and analysing the defective and non-defective texture of the fabric images. Datasets consisting of defective and non-defective fabric images are used to train the system for defect detection and classification. If the fabric image is clean, it is classified as good else it is defective. The defective images are further classified into colour, cut, hole, thread and metal contamination. Thus, image processing algorithms in DNN are used and the defects in fabrics are detected and classified.

Figure 3. Fabric defect detection system

IMAGE PRE-PROCESSING TECHNIQUES

Image pre-processing involves operation and manipulation of images at the lowest level of abstraction for all the input images in order to improve the effectiveness of image processing algorithms and the image quality. It reduces unwanted distortions and enhances some other image features for further processing so that the ML model can benefit from this improved data. In this method, the grayscale image is normalized to have a stable learning process and to reduce the number of training epochs. Once the image pre-processing is done, it moves to the next stage (feature extraction).

The commonly used steps include:

- I. Resizing the image
- II. Denoising
- III. Segmentation
- IV. Morphing

The following sub-sections elaborate on the pre-processing and data augmentation techniques that were carried out on our fabric dataset.

Image scaling

The raw image (of size 1024x1024) was resized to 32x32 by retaining the features necessary for testing and training, in order to eliminate the unnecessary details in the image.

Mathematically the algorithm for image scaling is interpreted as follows [17]:

Let $w \times h$ be the image size, t be the threshold and the ratio be $r = w/h$. If $t > \max(w, h)$ no operation needs to be performed.

Now, let $w \geq h$; if we want to find a linear mapping $f: [0, w] \rightarrow [0, t]$, that is

$$f(x) = ax + b, f(0) = 0, f(w) = t \quad (1)$$

then from the first condition $b = 0$ and from the second $a = t/w$.

The new width is $w' = t$ since we have to preserve the ratio: $r = w'/h'$, and hence $h' = w'/r$. Similarly,

the mapping for y can be computed as: $g(y) = h'y/h$.

Let the new image have dimensions $w' \times h'$; then for every point in the image (x, y) , the transformed coordinates are $(x', y') = (w'x/w, h'y/h)$. If $w < h$ then $h' = t$ and $w' = rh'$.

Image Normalization

Image normalization makes sure that all the pixels are uniformly distributed. We implemented this with the help of the transform, which is a part of the pandas library with the parameters (mean and standard deviation) being 0.5. The normalized images are made to undergo pixel transformations in accordance with the formula whose simplified form is as follows:

$$\text{Normalized image} = \frac{(\text{image pixel} - \text{mean})}{\text{standard deviation}} \quad (2)$$

Let the pixels from the image be $\{x_i\}$ and the data matrix be $= [x_0, x_1, x_2, \dots, x_n]$. The normalized image data x' is computed as [18]:

$$x' = \frac{(x_i - \bar{x})}{\sigma} \quad (3)$$

where $\bar{x}$ is the mean value of $\{x_i\}$ and σ is the standard deviation of $\{x_i\}$ of the entire dataset.

Dimension Reduction

Dimension reduction technique reduces the number of channels used, in order to avoid complexity during training. The original RGB image (3 channels / 3D) will be transformed to grayscale image (1 channel / 1D), thus reducing the number of channels used from three to one. It involves two components viz: feature selection and feature extraction. In literature, principal component analysis (PCA), linear discriminant analysis (LDA), and generalized discriminant analysis (GDA) techniques can be seen quoted frequently [19-21]. Figure 4 depicts a set of sample images from each class after the pre-processing.

Figure 4. Image patches after the pre-processing

Data Augmentation

Data augmentation expands the dataset by making varied versions of the image of interest. The purpose of data augmentation is to make a huge dataset from the scanty available dataset, so that it is feasible to train the model using the desired network architecture. Some of the processes carried out to transform the images are: cropping, rotating to various angles, shifting, and flipping. The dataset made use of the augmentation techniques that are likely to replicate the scenarios occurring during processing in the Indian apparel industry. The following are the five augmentation techniques that have been carried out on our dataset: rotation, flipping - horizontal and vertical, blurring and noise addition, after which we obtained 108,000 images for each of the five augmentation techniques.

Rotation

Our dataset comprises augmented images of each class of defect and the non-defective class images from the previous stage which have been rotated through angles 20°, 40°, 60°, 80°, 100°, 120° and 140°. This kind of augmentation replicates the scenario wherein the fabric might accidentally be fed upside down into the automatic defect detection system.

The rotation operation performs a geometric transformation by mapping the source pixel (x1, y1) to the destination pixel (x2, y2) by rotating through angle θ around the centre (x0, y0) [22].

$$x_2 = \cos(\theta) * (x_1 - x_0) - \sin(\theta) * (y_1 - y_0) + x_0 \quad (4)$$

$$y_2 = \sin(\theta) * (x_1 - x_0) + \cos(\theta) * (y_1 - y_0) + y_0 \quad (5)$$

Yet another simple mathematical representation for rotation can be in the form of the following matrix [23]:

$$\begin{bmatrix} \cos(q) & \sin(q) & 0 \\ -\sin(q) & \cos(q) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (6)$$

where q is the angle of rotation.

Flipping

Flipping refers to mirroring any given image either parallel or perpendicular to the plane being considered. In horizontal flipping, the images are flipped from side to side, whereas in vertical flipping the images are flipped in a bottom-up manner. This technique is carried out to ensure that there would not be any discrepancies while identifying such an image, by the trained model, when encountered in real-time.

Blurring

Blurring is an augmentation technique wherein the quality of the image is deteriorated deliberately, to ensure that it is better equipped for dealing with real-time scenarios. Relating to the context of our project, it is very likely that a camera might capture a hazy sample image of fabric in an industrial set-up, given large rolls of fabrics would be moving in a conveyor.

Here, Gaussian blur is used to blur the image, which inherently replaces a pixel by a weighted average of the connected pixels. It is given by [24];

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\left(\frac{x^2 + y^2}{2\sigma^2}\right)} \quad (7)$$

where σ is the standard deviation and (x, y) are the coordinates of the pixels in the Gaussian Blur Kernel.

Convolving, the $G(x,y)$ with the actual image results in a blurred image

Noise Addition

In this augmentation technique, random noise is added to the images in the dataset so that each image is different. This would mimic a real-time situation wherein sometimes the sample image captured might be corrupted by noise [25]. To the images in our dataset, Gaussian noise has been injected while augmenting. Figure 5 depicts the sample images using the mentioned augmentation techniques from the “Hole” class.

Figure 5. Image patches after data augmentation

Image Enhancement

Histogram equalization is generally employed for adjusting image intensities and hence reinforcing contrast [26]. Its primary goal is to make sure that the intensity is uniformly distributed on the histogram thus improving the visibility of the image. The conventional equalization described in is used here to ensure all the pixels are defined [27]. Figure 6 represents the histogram of the various classes before and after equalization.

Figure 6. Histogram of image patches before and after equalization

NETWORK ARCHITECTURE

AlexNet

AlexNet is a pre-trained network that has the ability to classify images into 1000 object categories. The network was initially introduced by A. Krizhevsky [28]. Figure 7 shows the architecture of AlexNet which consists of eight layers. In the eight layers of Alexnet, the initial five layers are convolutional and the pending 3 are fully connected layers. There are 4096 neurons in each of the fully connected layers. The various convolutional filters (kernels) extract interesting features in a fabric image. Each single convolutional layer has several filters of the same size. After the five convolutional layers, the output

of the remaining overlapping max-pooling layers is fed into two fully connected layers. The down sampling of the height and width of the tensors are performed by the max-pooling layers.

Figure 7. Architecture of AlexNet

Overlapping max pool layers are the same except for the fact that the maximum of the adjacent windows overlaps. The overlapping nature of pooling reduces the top-1 error rate by 0.4% and top-5 error rate by 0.3% when compared to non-overlapping pooling windows of size 2×2 with a stride of 2 that gives the same output dimensions. The second fully connected layer feeds into a softmax classification layer with a 1000 class labels. After the fully connected layers, the one, are followed by a max pooling operation and a rectified linear unit (ReLU) nonlinearity function is used not only to enable the NN to train at a much faster rate than the saturated activation function, like tan h or sigmoid, but also the networks with ReLU consistently learn several times faster than the equivalents with saturating neurons.

In the proposed AlexNet architecture, there are six convolution layer operations in sequence. Except the third and fourth, all the convolution layers have 3×3 kernels, with a stride and the padding of activation function. Each max pooling operation performs zero padding with a stride of 2. It keeps the depth unchanged and down samples the height and width of the tensors. The ReLU activation function helps in overcoming the vanishing gradient problem. This allows the model to learn faster and perform better. Three fully connected layers come after the sixth layer. They multiply their inputs by trainable weight vectors and with trainable bias.

Optimization Algorithm

The foremost goal of incessant training in a NN is to reduce the error and increase the accuracy, for which adjusting the weights in each consecutive iteration is indispensable. In order to ensure that the learning rate is perfect, optimizers are used. In our model we made use of the adaptive moment

estimation (Adam) algorithm, which is often used interchangeably with the vanilla stochastic gradient descent (SGD) algorithm.

Adam optimizer inherits the AdaGrad and RMSProp algorithms and merges its best properties. Adam stores an exponentially decaying average of past squared gradients and past gradients v_t and m_t , respectively, given by [29]:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \quad (8)$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \quad (9)$$

where β_1 , β_2 are the decay rates and g_t is the current gradient.

EXPERIMENTAL RESULTS

Performance Evaluation

AlexNet provides better performance than the other networks with an accuracy of 92.60%. The test accuracy of various architectures is given in Table 1.

Table 1. Accuracy of Various Architectures

Model	Accuracy, %
AlexNet after data augmentation	92.60
Differential evolution [7]	93.40
Co-occurrence matrix [30]	90.78
Mathematical morphology [31]	90.41
CNN with AlexNet	81.20
CNN	73.80
MLP	74.13

Learning Curve

Figure 8 indicates the decrease in training loss and validation loss and Figure 9 indicates the increase in the training accuracy and validation accuracy with subsequent epochs. After training the AlexNet model for 25 epochs, the training and validation losses were 0.368 and 0.243, respectively, and the training and validation accuracies were 88.10% and 92.67%, respectively.

Figure 8. Training and Validation loss

Figure 9. Training and Validation Accuracy

Error Matrix

The confusion matrix shown in Table 2 summarizes the number of correct and incorrect predictions obtained after training with AlexNet architecture.

Among the various commonly used evaluation metrics, the sensitivity, specificity, positive predictive value (PPV), negative predictive value (NPV) and their denotations are discussed in the succeeding section [32].

Depending on the actual class and the class predicted while testing the model, the prediction is classified into one of the following [33]:

- True positive (TP) indicates the outcome, where the model correctly predicts the positive class.
- True negative (TN) indicates the outcome where the model correctly predicts the negative class.
- False positive (FP) indicates the outcome where the model incorrectly predicts the positive class when it is actually negative. It is also called a type 1 error.
- False negative (FN) indicates the outcome where the model incorrectly predicts the negative class when it is actually positive. It is also called a type 2 error.

Table 2. Confusion matrix of the trained AlexNet

		Colour	Cut	Good	Hole	Metal	Thread
A C T U A L	Colour	92.40	1.80	2.50	1.80	0.82	0.70
	Cut	1.98	90.00	1.73	3.80	1.30	1.20
	Good	2.10	4.20	90.60	1.60	0.48	1.10
	Hole	1.81	2.80	3.10	91.70	0.40	0.20
	Metal	0.84	0.37	1.68	0.36	95.3	1.45
	Thread	0.86	0.65	0.41	0.81	2.00	95.30
		PREDICTED CLASS					

Specificity can be understood as the correct identification of the non-defective samples.

$$\text{Specificity} = \frac{TN}{TN + FP} \quad (10)$$

Sensitivity can be interpreted as the correct identification of defective samples and is sometimes referred to as recall.

$$\text{Sensitivity} = \frac{TP}{TP + FN} \quad (11)$$

PPV is the proportion of positive results that are true positive and is also known as precision. NPV is the proportion of negative results that are true negative. PPV and NPV are not intrinsic to the test (i.e.) they depend on the prevalence.

$$\text{Negative predictive value (NPV)} = \frac{TN}{FN + TN} \quad (12)$$

$$\text{Positive predictive value (PPV)} = \frac{TP}{TP + FP} \quad (13)$$

The proposed model turned out to obtain 95% (approx.) specificity, sensitivity, NPV and PPV.

Testing and Cross-Validation

An image of a colour defect is given as a sample for testing the network and the network was able to classify it accurately as shown in Figure 10.

Prediction in progress
color defect

Figure 10. Output from testing the trained network

CONCLUSION

The deep learning model was trained for five defect classes: cut, colour, hole, thread, metal contamination. The dataset used is a part of the public dataset, MVTec Anomaly Detection. The trained model was limited only to these five defect classes not only due to the number of images available in the open source database, but also the fact that in India these five defect classes remain as the most prevalent faults found in today's textile industry. The developed Deep Convolutional Neural Network (DCNN) classifier, which employs transfer learning using the AlexNet pretrained network yielded maximum accuracy of 92.60% after continuous testing and training. The developed DL-based classifier model could be deployed in textile production which will have a vital role in improving the overall

efficiency of the fabric in quality and cost. This will not only enable accurate and rapid defect identification but also help differentiate minute differences between the defect classes. Particularly, it aids the industry in functioning without manual inspectors, which plays a crucial role concerning efficiency and net profit, especially in times like the current pandemic where remote operation is the new normal.

Future scope of this work includes:

- Incorporating many more defect classes and making it more suitable for real-time scenarios.
- Integrating possible hardware components, preferably those often used in textile inspection sites.

Author Contribution

Conceptualization – Sashikumar NM and Kumarasamy K; methodology – Sashikumar NM, Wenisch SM, Sandhya NC and Priyanka M; formal analysis – Sashikumar NM and Kumarasamy K; investigation – Sandhya NC and Priyanka M; resources – Sandhya NC and Priyanka M; writing-original draft preparation – Sashikumar NM, Sandhya NC and Priyanka M; writing-review and editing – Wenisch SM; visualization – Wenisch SM; supervision – Kumarasamy K. All authors have read and agreed to the published version of the manuscript.

Conflicts of Interest

The authors declare no conflict of interest.

Acknowledgements

The authors greatly appreciate the hardware GPU support provided by Nvidia Corporation.

REFERENCES

- [1] Sengottuvelan P, Wahi A, Shanmugam A. Automatic Fault Analysis of Textile Fabric Using Imaging Systems. *Research Journal of Applied Sciences*. 2008; 3(1):26-31.
- [2] National Portal of India. Available from: <http://niti.gov.in/weaving-way-indian-textile-industry>
- [3] Patel J, Jain M, Dutta P. Detection of Faults Using Digital Image Processing Technique. *Asian Journal of Engineering and Applied Technology*. 2013; 2(1):36-40. <https://doi.org/10.51983/ajeat-2013.2.1.644>
- [4] Hanbay K, Talu MF, Özgüven ÖF. Fabric defect detection systems and methods—A systematic literature review. *Optik*. 2016; 127(24):11960-11973. <https://doi.org/10.1016/j.ijleo.2016.09.110>

- [5] Chan CH, Pang GKH. Fabric defect detection by Fourier analysis. IEEE Transactions on Industry Applications. 2000; 36(5):1267-1276. <https://doi.org/10.1109/28.871274>
- [6] Bodnarova A, Bennamoun M, Latham SJ. Optimal Gabor filters for textile flaw detection. Pattern Recognition. 2002; 35(12):2973-2991. [https://doi.org/10.1016/S0031-3203\(02\)00017-1](https://doi.org/10.1016/S0031-3203(02)00017-1)
- [7] Tong L, Wong WK, Kwong CK. Differential evolution-based optimal Gabor filter model for fabric inspection. Neurocomputing. 2016; 173:1386-1401. <https://doi.org/10.1016/j.neucom.2015.09.011>
- [8] Jing JF, Hao Ma H, Zhang HH. Automatic fabric defect detection using a deep convolutional neural network. Coloration Technology. 2019; 135(3):213-223. <https://doi.org/10.1111/cote.12394>
- [9] Jun X, Wang J, Zhou J, Meng S, Pan R, Gao W. Fabric defect detection based on a deep convolutional neural network using a two-stage strategy. Textile Research Journal. 2021; 91(1-2):130-142. <https://doi.org/10.1177/0040517520935984>
- [10] Liu Z, Zhang C, Li C, Ding S, Dong Y, Huang Y. Fabric defect recognition using optimized neural networks. Journal of Engineered Fibers and Fabrics. 2019; 14:1-10. <https://doi.org/10.1177/1558925019897396>
- [11] Cao G, Ruan S, Peng Y, Huang S, Kwok N. Large-Complex-Surface Defect Detection by Hybrid Gradient Threshold Segmentation and Image Registration. IEEE Access. 2018, 6:36235-36246. <https://doi.org/10.1109/ACCESS.2018.2842028>
- [12] Li Y, Zhang C. Automated vision system for fabric defect inspection using Gabor filters and PCNN. SpringerPlus 5. 2016, 765. <https://doi.org/10.1186/s40064-016-2452-6>
- [13] Vermaak H, Nsengiyumva P, Luwes N. Using the Dual-Tree Complex Wavelet Transform for Improved Fabric Defect Detection. Journal of Sensors. 2016; 9794723. <https://doi.org/10.1155/2016/9794723>
- [14] Bergmann P, Fauser M, Sattlegger D, Steger C. MVTec AD — A Comprehensive Real-World Dataset for Unsupervised Anomaly Detection. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019 June 16–20 Long Beach, California; Computer Vision Foundation; 2019:9584-9592. <https://doi.org/10.1109/CVPR.2019.00982>
- [15] Kunaraj K, Wenisch SM, Balaji S, Don Bosco MFP. Impulse Noise Classification Using Machine Learning Classifier and Robust Statistical Features. In: Smys S, Tavares J, Balas V, Iliyasu A (eds) Computational Vision and Bio-Inspired Computing. ICCVBIC 2019. Advances in Intelligent Systems and Computing. 2020; vol 1108. Springer, Cham. p.631-644. https://doi.org/10.1007/978-3-030-37218-7_72
- [16] Kumarasamy K, Wenisch SM, Balaji S, Suriya LJJ, Jerlin A, Rajkumar R. Improving Impulse Noise Classification Using Ensemble Learning Methods. Series: Advances in Intelligent Systems and Computing. 2021; 1299:187-199. https://doi.org/10.1007/978-981-33-4299-6_16

- [17] Mathematics. Resizing and scaling image. Available from: <https://math.stackexchange.com/questions/3078121/resizing-and-scaling-image>
- [18] Han YJ, Yu HJ. Fabric Defect Detection System Using Stacked Convolutional Denoising Auto-Encoders Trained with Synthetic Defect Data. *Applied Sciences*. 2020; 10(7):2511. <https://doi.org/10.3390/app10072511>
- [19] Ma J, Yuan Y. Dimension reduction of image deep feature using PCA. *Journal of Visual Communication and Image Representation*. 2019; 63, 102578. <https://doi.org/10.1016/j.jvcir.2019.102578>
- [20] Kim H, Drake BL, Park H. Multiclass classifiers based on dimension reduction with generalized LDA. *Pattern Recognition*. 2007; 40(11):2939-2945. <https://doi.org/10.1016/j.patcog.2007.03.002>
- [21] Dogantekin E, Dogantekin A, Avci D. An expert system based on Generalized Discriminant Analysis and Wavelet Support Vector Machine for diagnosis of thyroid diseases. *Expert Systems with Applications*. 2011; 38(1):146-150. <https://doi.org/10.1016/j.eswa.2010.06.029>
- [22] Image Processing Learning Resources. Rotate. Available from: <https://homepages.inf.ed.ac.uk/rbf/HIPR2/rotate.htm>
- [23] Medium. Affine Transformation – Image Processing in TensorFlow – Part 1. Available from: <https://medium.com/mlait/affine-transformation-image-processing-in-tensorflow-part-1-df96256928a>
- [24] Song C, Xu W, Wang Z, Yu S, Zeng P, Ju Z. Analysis on the Impact of Data Augmentation on Target Recognition for UAV-Base Transmission Line Inspection. *Complexity*. 2020; 3107450:1-11. <https://doi.org/10.1155/2020/3107450>
- [25] Medium. Mathematical foundation for Noise, Bias and Variance in #NeuralNetworks. Available from: <https://medium.com/autonomous-agents/mathematical-foundation-for-noise-bias-and-variance-in-neuralnetworks-4f79ee801850>
- [26] Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. *Communications of the ACM*. 2017; 60(6):84–90. <https://doi.org/10.1145/3065386>
- [27] Xiong J, Yu D, Wang Q, Shu L, Cen J, Liang Q, Chen H, Sun B. Application of Histogram Equalization for Image Enhancement in Corrosion Areas. *Shock and Vibration*. 2021; 8883571:1-13. <https://doi.org/10.1155/2021/8883571>
- [28] Zhu H, Chan FHY, Lam FK. Image Contrast Enhancement by Constrained Local Histogram Equalization. *Computer Vision and Image Understanding*. 1999; 73(2):281-290. <https://doi.org/10.1006/cviu.1998.0723>
- [29] Ruder S. An overview of gradient descent optimization algorithms. *ArXiv*. 2016; [1609.04747](https://arxiv.org/abs/1609.04747)
- [30] Latif-Amet A, Ertuzun A, Ercil A. An efficient method for texture defect detection: sub-band domain

- co-occurrence matrices. Image and Vision Computing, 2000; 18(6-7):543–553. [https://doi.org/10.1016/S0262-8856\(99\)00062-1](https://doi.org/10.1016/S0262-8856(99)00062-1)
- [31] Zhang YF, Bresee RR. Fabric defect detection and classification using image analysis. Textile Research Journal. 1995; 65(1):1–9. <https://doi.org/10.1177/004051759506500101>
- [32] Rasheed A, Zafar B, Rasheed A, Ali N, Sajid M, Dar SH, Habib U, Shehryar T, Mahnood MT. Fabric Defect Detection Using Computer Vision Techniques:A Comprehensive Review. Mathematical Problems in Engineering. 2020; 8189403. <https://doi.org/10.1155/2020/8189403>
- [33] Liong ST, Gan YS, Huang YC, Yuan CA, Chang HC. Automatic Defect Segmentation on Leather with Deep Learning. ArXiv. 2019; [1903.12139](https://arxiv.org/abs/1903.12139)