IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received May 7, 2021, accepted May 24, 2021, date of publication June 3, 2021, date of current version June 14, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3086028

Fabric Defect Detection With Deep Learning and False Negative Reduction

TOMÁS ALMEIDA[1], FILIPE MOUTINHO[1,2], AND J. P. MATOS-CARVALHO[3]
[1]NOVA School of Science and Technology, 2829-516 Caparica, Portugal
[2]Centre of Technology and Systems, UNINOVA, 2829-516 Caparica, Portugal
[3]Cognitive and People-Centric Computing Laboratories (COPELABS), Universidade Lusófona de Humanidades e Tecnologias, 1749-024 Lisbon, Portugal

Corresponding author: Filipe Moutinho (fcm@fct.unl.pt)

This work was supported in part by the Portuguese Fundação para a Ciência e a Tecnologia (FCT) under Project UIDB/04111/2020 and Project UIDB/00066/2020.

ABSTRACT Quality control is an area of utmost importance for fabric production companies. By not detecting the defects present in the fabrics, companies are at risk of losing money and reputation with a damaged product. In a traditional system, an inspection accuracy of 60-75% is observed. In order to reduce these costs, a fast and automatic defect detection system, which can be complemented with the operator decision, is proposed in this paper. To perform the task of defect detection, a custom Convolutional Neural Network (CNN) was used in this work. To obtain a well-generalized system, in the training process, more than 50 defect types were used. Additionally, as an undetected defect (False Negative - FN) usually has a higher cost to the company than a non-defective fabric being classified as a defective one (false positive), FN reduction methods were used in the proposed system. In testing, when the system was in automatic mode, an average accuracy of 75% was attained; however, if the FN reduction method was applied, with intervention of the operator, an average of 95% accuracy can be achieved. These results demonstrate the ability of the system to detect many different types of defects with good accuracy whilst being faster and computationally simple.

INDEX TERMS CNN, deep learning, fabric defect detection, false negative reduction.

I. INTRODUCTION

Fabric defect detection is a quality control process that has to ensure the identification of defects present in the textile fabric. These defects can reduce the textile fabric price as much as 45% to 65% [1]. A traditional inspection system is composed of manual workers/operators. Their job is to detect the defects while the fabric is being moved by a machine. Traditionally, this motorized machines unroll the fabric rolls, so that the fabric is stretched and presented to the worker without folds and thickness differences. As this process relies on human visual ability and concentration, this task can be very tedious and time-consuming, which can lead to fatigue and consequently human error. Therefore, traditional systems can only achieve a 60% to 75% accuracy, even though they have very slow speed compared to production rate. As a result, automatic visual inspection systems to ensure the high quality of products in production lines are in increasing demand.

The associate editor coordinating the review of this manuscript and approving it for publication was Corrado Mencar.

The main advantages of the automatic defect detection systems, compared to human inspection, are high efficiency, reliability and consistency [2]. Nickoloy and Schmalfub [3] have shown that the investment in automated fabric inspection systems is economically attractive when reduction in personnel cost and associated benefits are considered. Although automated visual inspection systems have many advantages, there are still some obstacles to overcome. The large number of features, interclass similarity and intraclass diversity of fabric defects form major obstacles to perform this task [4].

To overcome the mentioned difficulties, a fast fabric defect detection system that was trained in more than 50 defect types, which was not observed in literature, is proposed in this paper. This system, presented in Figure 1, is an automatic system, which can be complemented with the operator decision. In this work, three main contributions can be highlighted:

1) A new Convolutional Neural Network (CNN) defect detection system, which can be operator-assisted, is proposed;

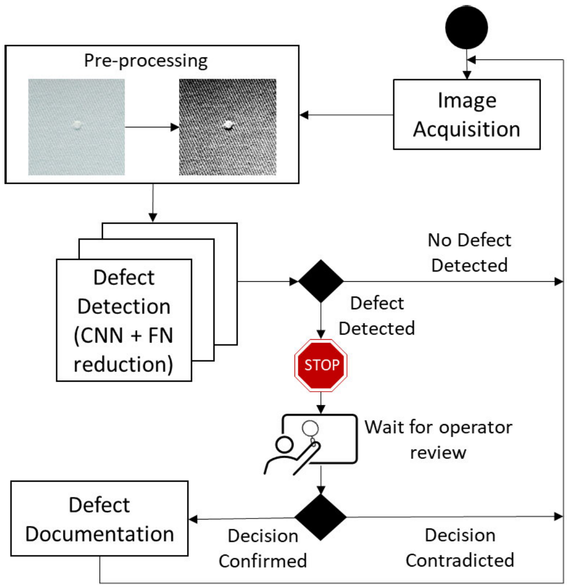


FIGURE 1. Fabric defect detection system operation flowchart.

2) As the images classified as no-defect by the system are not reviewed by the operator, two False Negative (FN) reduction methods were studied;
3) In order to solve the problem of underrepresented defect types on existing datasets, a new and more general dataset was created.

The CNN architecture was created from scratch, with Convolutional, Max-pooling, ReLU and Dense layers. No pretrained networks or weights were used. In order to obtain the best architecture/configuration various training and tests were performed with the chosen datasets. The selected datasets were collected from various sources, in order to compare and test this model against existing works. Layer feature visualization and exhaustive testing on four different datasets, also constitute contributions that are not usually seen in similar works. The proposed system presents high accuracy and lower execution time, in comparison with related works. The interaction between the system and the operator, complemented by the use of FN reduction methods, constitutes a novel approach to this problem.

Further structure of this paper is the following. Section II provides a review of several fabric defect detection approaches observed in literature, whereas section III identifies the datasets used in this paper. Section IV describes the novel fabric defect detection system as well as the false negative reduction methods that were explored. Section V presents and discusses the experimental results. Finally, section VI provides some conclusions and future improvements to this system.

II. RELATED WORK

This section provides a literature review of some fabric defect detection techniques. These techniques can be divided into the following categories: statistical, spectral, model-based and learning.

A. STATISTICAL APPROACHES

Statistical approaches mainly include autocorrelation function, eigenfilters, histogram, Gray Level Co-occurrence Matrix (GLCM), Local Binary Patterns (LBP), mathematical morphology and fractal dimension. These methods study the statistical properties of the relationships between the gray levels of an image. The statistical properties of the defective fabric stand out from the defect-free regions that remain statistically constant throughout a significant portion of the inspection images [4]. Most of these methods require prior knowledge of the defect-free characteristics when they are used single-handed and are only effective in a few types of defects. Wood [5] used autocorrelation function to describe and analyze the symmetry of carpet fabric patterns. Unser and Ade [6] used eigenfilters information for defect detection in textured materials, obtaining good results and low rate of false alarms. Monadjemi [7], [8] proposed the application of structurally matched eigenfilters generated by rotation, negation and mirroring for texture defect detection. In 2010, Thilepa [9] used histogram equalization with noise filtering and thresholding to detect fabric defect and 85% accuracy was attained.

Haralick *et al.* [10] defined 14 different characteristics extracted from the GLCM to analyze texture features and tested the system in various known textures like wood patterns. Rosler [11] used these characteristics to develop a defect detection system that obtained 95% accuracy in some defect types. In 2011, Ben Salem and Nasri [12] combined the GLCM characteristics with a Support Vector Machine (SVM) to classify different defect types and an average accuracy of 80% was obtained.

Tajeripour *et al.* [13] used LBP to extract the characteristics of defect-free fabric samples, comparing them with the test samples and then find the defective ones. Ben Salem and Nasri [12] combined this technique with an SVM to classify defective samples and obtained an average of 86,7% accuracy. In 2001, Kwak *et al.* [14] used morphologic operations and thresholding, combined with a decision tree, to detect and classify the defects, obtaining 91,25% accuracy. Mak *et al.* [15], [16] created morphological filters to detect fabric defects and achieved 96,7% accuracy. Conci and Proença [17] suggested an inspection system based on the variation of fractal dimension to detect the fabric defects. The authors found this approach very simple but also experimentally limited, with poor defect localization accuracy and high false alarm, resulting in 96% detection but only on eight defect types.

B. SPECTRAL APPROACHES

Many of the statistical approaches discussed break down when used on fabric defects that present very smooth transitions. Therefore, other more robust and efficient methods have been created. Spectral methods analyze the textured images in the spatial frequency domain and require a high degree of periodicity. These methods include variants of Fourier transform, Gabor filters and wavelet transform. Hoffer *et al.* [18] used the optical Fourier transform combined with a neural network to detect defects in cloth fabric. Tsai and Hsieh [19] tested a combination of a discrete Fourier transform and Hough transform to enhance the defective region of the image. Bovik *et al.* [20] used Gabor filters to classify different types of defective pattern fabrics that had been previously studied. Sari-Sarraf and Goddard [21] developed a low cost and easy to implement inspection system based on wavelet transform, that obtained 89% accuracy and a 2.5% false alarm rate. Dorrity and Vachtsevanos [22] built a defect detection system that combined wavelet transform and fuzzy analysis. Yildiz and Buldu [23] achieved 95% classification accuracy combining wavelet transform and principal component analysis with the help of thermal imaging.

C. MODEL-BASED APPROACHES

The textures present in an image can be defined by predetermined parameters that constitute an aleatory or deterministic model. Although these methods are very complex and computationally expensive, they may be adequate for images with non-uniform patterns. Autoregressive model and Gaussian Markov random field are some of the methods that constitute this type of approaches. These models can sometimes become very complex and computationally expensive. McCormick and Jayaramamurthy [24] used autoregressive models do synthesize different fabric textures and compare them to predefined textures. Cohen *et al.* [25] modeled the defect-free fabric with a Gaussian Markov random field model and compared it to the test image fabrics to detect the fabric defects.

D. LEARNING APPROACHES

The latest technological developments in processing power and volume of data facilitate the adoption of learning approaches. These methods try to find patterns in the extracted characteristics of the fabric image and therefore can be used standalone or as a complement to other methods like LBP [26] and GLCM [27]. SVMs [28], Feed-forward Networks (FFNs) [29], [30] and CNNs [31]–[33] are some of the methods included in this type of approaches. In 2011, Ghosh [34] developed an SVM-based inspection system, to detect three different types of fabric defects. Kumar [35] used principal component analysis to reduce the dimension of the characteristics vector, complemented with a two-layer FFN, and obtained a robust low-cost defect detection system. Mei *et al.* [36] proposed an unsupervised learning-based automated approach by using a multi-scale convolutional denoising auto-encoder network and Gaussian pyramid to detect and segment fabric defects. Their overall inspection accuracy reached over 80.0% on all datasets. In 2019, Ouyang *et al.* [37], used a CNN to detect defects in a dataset created by the authors and a 98.82% accuracy was attained.

Although the mentioned methods may present high accuracy rates in defect detection, some are computationally inefficient. Moreover, their accuracy may not be very high when many fabric defect types are considered. It is important to note that these methods were used with different purposes (defect detection, defect segmentation, and defect classification); however, in this work we are just focused on defect detection. Even though learning approach-based systems provide high-level information and are considered state-of-the-art, they also have some limitations. These methods have a black-box character, and require high computational resources and large training sets to obtain accurate models. Additionally, although the defect classification as False Positive (FP) or FN has different consequences on the company production, with impact on the production speed and product quality, most defect detection systems do not consider this problem.

III. DATA GATHERING

For this work, four different datasets were used. In addition to the three existing datasets (TILDA, MVTec and Stains dataset) and to have a more general dataset that could better represent the many different defect types observed in a production environment, a new dataset was created (Fabric-Net-Dataset). The Fabric-Net-Dataset was divided into three parts that were used in train, validation and test phases. The other three datasets were only used in the test phase. It is important to note that all the images correspond to pattern less fabric and were resized to 150×150 dimension (using the "inter_area" interpolation function), in order to shorten the execution times and thus make the network even faster. Several tests were performed to find an image size and the input size of the proposed CNN (see subsection IV-C) that would obtain the best balance between the amount of image information and the model's effectiveness. In the case of TILDA and MVTec datasets only a portion of the images were randomly selected for the testing phase, in order to produce balanced datasets for this binary defect detection problem, required to ensure the correctness of the defined metrics results (accuracy, recall, sensitivity and others).

TILDA (FD-TL) is a dataset developed by the group "Texture Analysis of the DFG's" [38]. This dataset is divided into 8 different fabric types, each one containing seven defect classes, making a total of 3200 images. Since some defect types are not related to the fabric structure, three of the defect types were not considered.

MVTec Anomaly Detection Dataset (FD-MV) was created by the MVTec company [39] and contains many different object classes from different industries. For this work, only the "carpet" class was used. Given the different nature of carpet fabric structure in comparison with other textiles, this dataset can demonstrate the robustness and generalization of the fabric defect detection system created.

The Fabric Stains Dataset (FD-ST) was created by the Intellisense lab of Moratuwa University, Sri Lanka [40]. This dataset contains images of two different fabric types with stain defects.

Fabric-Net-Dataset (FD-NT) was created with images from two different sources, namely the fabric defect images from Cotton Incorporated [41] and other fabric images found

TABLE 1. Used datasets, in a total of 1394 image samples.

Dataset	No. of images
FD-TL	28.70%
FD-MV	12.76%
FD-ST	9.76%
FD-NT (train)	36.58%
FD-NT (validation)	7.32%
FD-NT (test)	4.88%

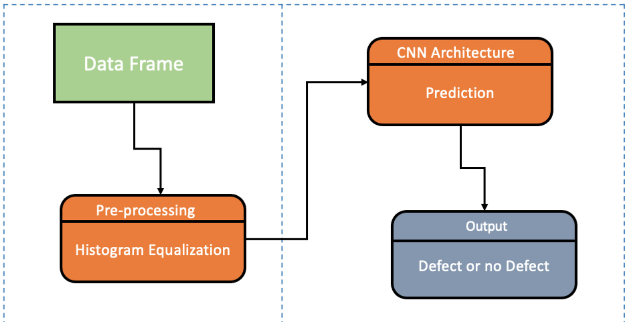


FIGURE 2. Fabric defect detection system block diagram.

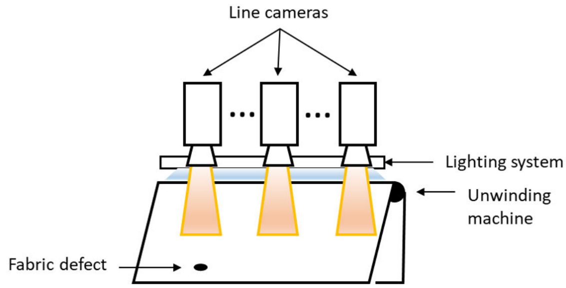


FIGURE 3. Structure of the visual inspection system.

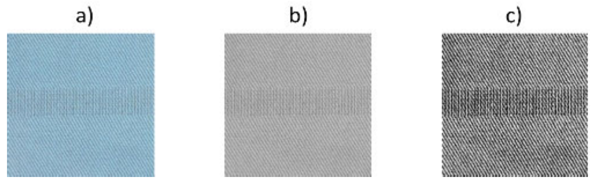


FIGURE 4. Pre-processing Fabric-Net-Dataset example: a) original image; b) grayscale transformation; c) histogram equalization.

on the web. The dataset from Cotton Incorporated contains 194 images with more than 50 different defect types which allows for a broader representation of fabric defect classes. To complement and increase the number of examples in this dataset, other images were extracted from the Web.

Summary: In total, there are 1394 images in the sum of the four datasets. Each dataset is balanced between the two classes and the weight of each one is represented in Table 1.

IV. PROPOSED SYSTEM

The proposed system is divided into several processes, as presented in Figure 2. Firstly, the image acquisition system captures the fabric image, which is then pre-processed, with histogram equalization, to enhance the defective region. Then, image analysis is performed to detect the fabric defect. A customized CNN architecture was used for the defect detection process, in order to create a robust and computationally effective defect detection system. If this system detects a possible defect in the fabric roll, the system's activity is stopped, and the nearby operator is informed. After that the operator confirms or reverses the system evaluation and gives the authorization for the system to proceed the operation and continue to the next fabric image. Furthermore, the defect type, location and other characteristics can be documented for later research. In the following subsections, the system processes as well as the CNN architecture and false negative reduction methods are described.

A. IMAGE ACQUISITION

In order to have a fast RGB (Red, Green, Blue) image acquisition system with low noise and high resolution images, a series of line scan CMOS (Complementary Metal-Oxide-Semiconductor) sensors cameras can be used. These cameras capture less blurry images and have more speed and range of motion then other scan cameras [42]. To capture the entire width of the inspected fabric with good resolution, this system uses a row of line cameras, jointly with the lighting system, while the fabric roll is unwinded by a machine, as illustrated in Figure 3.

B. PRE-PROCESSING

In addition to image resizing, described in section III, grayscale transformation and histogram equalization were performed in order to increase the contrasts between the objects present in the images. This can emphasize the differences between the uniform fabric structure and the defect region as shown in Figure 4.

C. CNN ARCHITECTURE

This section provides the configuration of the custom CNN architecture used in this defect detection system. As this system must be able to fast process the images from several cameras, to allow the roll to be inspected as quickly as possible, a simpler architecture was used instead of other more complex state-of-the-art deep learning models. To further improve system speed, in this stage, only defect detection is performed, leaving defect classification for a later stage. Figure 5 shows an overview of the CNN layers and its dimensions. This architecture, is composed of four convolutional and max-pool layers, followed by two fully-connected layers. ReLU was used for all the activations. Table 2 contains all the layers and its hyperparameters, where F is the number of feature maps, K corresponds to the kernel size and S is the stride parameter. As mentioned, this CNN was trained and validated on Fabric-Net-Dataset. Each layer activations were also visualized in order to confirm that the network could recognize the fabric defect features (see subsection V-C).

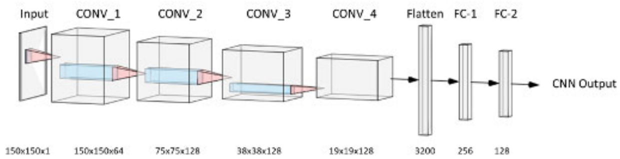


FIGURE 5. The proposed custom CNN layers dimension.

TABLE 2. The proposed custom CNN architecture hyperparameters.

Layer	Output	Hyperparameters
INPUT	150x150x1	
CONV_1	150x150x64	F=64; K=5x5; S=1;
MAX-POOL_1	75x75x64	pool=2x2; S=2;
CONV_2	75x75x64	F=64; K=5x5; S=1;
MAX-POOL_2	38x38x64	pool=2x2; S=2;
CONV_3	38x38x128	F=128; K=3x3; S=1;
MAX-POOL_3	13x13x128	pool=2x2; S=3;
CONV_4	13x13x128	F=128; K=3x3; S=1;
MAX-POOL_4	5x5x128	pool=2x2; S=3;
Flatten	3200	
FC-1	256	Neurons=256
FC-2	128	Neurons=128
CNN OUTPUT	1	optm=ADAM loss=binary_crossentropy

D. FALSE NEGATIVE REDUCTION

The lesser the number of FN and FP examples, the better the system performance. Even though in most literature work both are treated equally, usually, one of them has a higher cost. As in this system specifically, the FN cost will be higher than FP because all the FN examples will not be reviewed by the operator. Taking this into account, two FN reduction methods were studied and later evaluated. Both methods modify the standard classification threshold value. It is also important to note that there is always a trade-off between FN and FP, so reducing one will ultimately increase the other. As the percentage of FP examples will also be reviewed by the operator, increasing the number of FP will also increase the operators working time. Thus, in the end other variables like fabric cost and operator salary rate, must be considered. This improvement can offer flexibility to the system and can be used as a system variable that can be adjusted depending on the environment.

The **Classification Threshold Reduction (CTR)** method uses a lower value for the threshold to reduce the examples classified as defect-free by the detection system. This is because, if the detection system is well trained, most of the FN examples will be positioned close to the standard threshold value, therefore reducing the number of FN.

The **Rejection Region (RR)** method uses two different threshold values, therefore creating a rejection region. Every example classified as defect inside this region will be later reviewed by the operator. The second threshold allows the number of examples that must be reviewed by the operator to be decreased. However, with this method some FP examples won't be reviewed by the operator, therefore decreasing the system accuracy.

V. EXPERIMENTS

A. IMPLEMENTATION PLATFORM

The models were implemented using Tensorflow [43] and Keras API [44] and trained on a laptop with NVIDIA GeForce GTX 1650 Max-Q, 16GB RAM and Intel Core i7-9750H. This vindicates that the system proposed is simple and cheap to implement on a real-world environment.

B. EVALUATION METRICS

The method proposed in this work was evaluated and compared against state-of-the-art using standard metrics based on confusion matrix results. These metrics can be defined as:

- Accuracy (eq. 1);
- Recall (eq. 2);
- Precision (eq. 3);
- AUC - Area Under Curve;
- F2-score (eq. 4);
- Mathews Correlation Coeficient (MCC - eq. 5).

$$Accuracy = \frac{TP+TN}{TP+TN+FN+FP} \tag{1}$$

$$Recall = \frac{TP}{TP+FN} \tag{2}$$

$$Precision = \frac{TP}{TP+FP} \tag{3}$$

$$F2\text{-}score = ((1+\beta^2)\frac{recall \times precision}{(\beta^2 \times precision)+recall)}, \text{ with } \beta = 2 \tag{4}$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \tag{5}$$

C. RESULTS

1) CNN DETECTS THE DEFECTIVE FEATURES

In order to confirm that our CNN can in fact identify the fabric defect features present in an image, the neurons activations in each layer were visualized. This method allows to verify if the network is well trained and behaves accordingly to the expected outcome. It can also show the characteristics behind false positive and false negative cases which can be used as a debug tool for these models. The following is an example of application of this method that distinguishes a well-classified defect example (see Figure 6) from a poorly classified one (see Figure 7). The heat maps represent the activation levels of some feature maps (not all) in each CNN layer. The more yellow it gets on the defective regions the greater it is the defect probability in that area. Comparing the two results, in the well-classified example (Figure 6) the CNN can better identify the defective features of both top and bottom defective regions throughout the convolutional layers, so the color difference between those regions and the rest of the image is very high. As a result, in the last layer the two defective regions are isolated and segmented from the rest of the image.

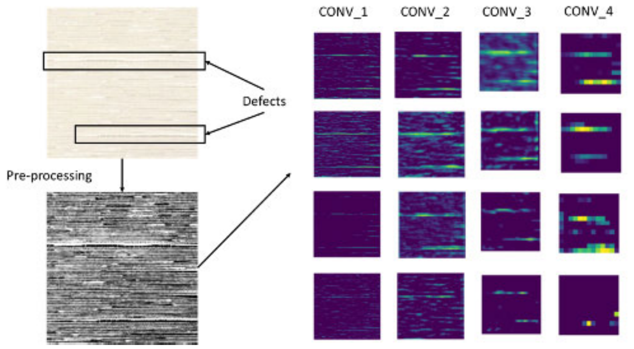


FIGURE 6. Visualization of the layers activation for a well-classified fabric defect sample image.

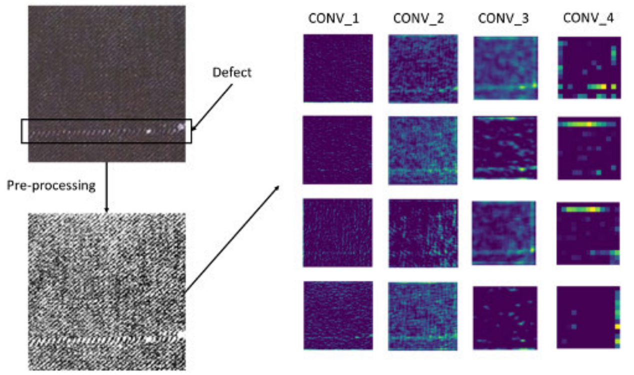


FIGURE 7. Visualization of the layers activation for a poorly classified fabric defect sample image.

This leads the network to correctly classify this example as a defective fabric with very high confidence (value = 0.9999). In contrast, in the second example (Figure 7), the CNN cannot clearly identify the defective features, even in the second layer. So, in the last layer the defective features are not well defined, and this leads the network to classify it as a FN (value = 0.1928).

2) CNN SURPASSES STATE-OF-THE-ART MODELS

As mentioned in subsection IV-C, a customized CNN was used to perform the task of defect detection. The training of both the CNN and state-of-the-art algorithms were performed using the FD-NT dataset together with data augmentation transformations to increase accuracy and reduce overfitting. They were then tested on four different test datasets. The state-of-the-art deep learning models were loaded with pretrained weights from ImageNet and the two final fully-connected layers were substituted by two 256 neuron layers. Only these two final layers were fine-tuned using FD-NT dataset. Then the whole architecture was trained with a smaller learning rate to fine-tune the networks. In Tables 3, 4 the performance of the proposed CNN is compared to the state-of-the-art fabric defect detection algorithms, in which GLCM and LBP methods were selected to perform feature extraction, SVMs and FFNs were used as classifiers. The results in Tables 3, 4 show that the proposed CNN outperforms the others defect detection algorithms across all test metrics. For example, the proposed CNN achieves an average accuracy of 0.7545 (see last column of Table 3), which is 0.07 more in average accuracy compared to the second best performing algorithm, and obtains less false negatives (an average of 20.25, as presented in Table 4). Some of

TABLE 3. Accuracy, loss, recall and precision of the proposed model and state-of-the-art defect detection techniques in different datasets with the respective average (between 0 and 1).

	Accuracy				
Model	FD-NT	FD-TL	FD-MV	FD-ST	Average
Proposed	0.8824	0.7450	0.7584	0.6324	0.7545
LBP+SVM	0.6912	0.5450	1.0000	0.4779	0.6785
GLCM+SVM	0.8529	0.6000	0.5169	0.6397	0.6524
LBP+FFN	0.6912	0.5525	1.0000	0.4706	0.6786
GLCM+FFN	0.7059	0.5000	0.2022	0.3824	0.4476
	Loss				
Model	FD-NT	FD-TL	FD-MV	FD-ST	Average
Proposed	0.3375	0.7167	0.5243	1.0006	0.6448
LBP+SVM	---------	---------	---------	---------	---------
GLCM+SVM	---------	---------	---------	---------	---------
LBP+FFN	1.0394	1.9577	0.0074	2.1801	1.2961
GLCM+FFN	0.5907	0.9479	1.5294	1.3261	1.0985
	Recall				
Model	FD-NT	FD-TL	FD-MV	FD-ST	Average
Proposed	0.9706	0.6200	0.9775	0.9706	0.8847
LBP+SVM	0.6765	0.4450	1.0000	0.5735	0.6738
GLCM+SVM	0.8529	0.4100	1.0000	0.8382	0.7753
LBP+FFN	0.7059	0.6050	1.0000	0.6029	0.7285
GLCM+FFN	0.6765	0.5000	0.4045	0.3529	0.4835
	Precision				
Model	FD-NT	FD-TL	FD-MV	FD-ST	Average
Proposed	0.8250	0.8267	0.6797	0.5789	0.7276
LBP+SVM	0.6970	0.5563	1.0000	0.4815	0.6837
GLCM+SVM	0.8529	0.6613	0.5086	0.6000	0.6557
LBP+FFN	0.6857	0.5475	1.0000	0.4767	0.6775
GLCM+FFN	0.7188	0.5000	0.2880	0.3750	0.4704

TABLE 4. AUC, FN, f2-score and MCC of the proposed model and state-of-the-art defect detection techniques in different datasets with the respective average (between 0 and 1).

	AUC				
Model	FD-NT	FD-TL	FD-MV	FD-ST	Average
Proposed	0.9191	0.8197	0.8597	0.7098	0.8271
LBP+SVM	---------	---------	---------	---------	---------
GLCM+SVM	---------	---------	---------	---------	---------
LBP+FFN	0.7098	0.5645	1.0000	0.4229	0.6743
GLCM+FFN	0.7651	0.6429	0.0110	0.3409	0.4400
	FN				
Model	FD-NT	FD-TL	FD-MV	FD-ST	Average
Proposed	1	76	2	2	20.25
LBP+SVM	11	111	0	29	37.75
GLCM+SVM	5	118	0	11	33.5
LBP+FFN	10	79	0	27	29
GLCM+FFN	11	100	53	44	52
	F2-Score				
Model	FD-NT	FD-TL	FD-MV	FD-ST	Average
Proposed	0.9375	0.6526	0.8988	0.8549	0.8360
LBP+SVM	0.6805	0.4635	1.0000	0.5524	0.6741
GLCM+SVM	0.8529	0.4437	0.8380	0.7766	0.7278
LBP+FFN	0.7018	0.5926	1.0000	0.5726	0.7167
GLCM+FFN	0.6845	0.5000	0.3742	0.3571	0.4790
	Mathews Correlation Coeficient				
Model	FD-NT	FD-TL	FD-MV	FD-ST	Average
Proposed	0.7769	0.5061	0.5750	0.3594	0.5543
LBP+SVM	0.3825	0.0919	1.0000	-0.0449	0.3574
GLCM+SVM	0.7059	0.2162	0.1309	0.3044	0.3394
LBP+FFN	0.3825	0.1056	1.0000	-0.0610	0.3568
GLCM+FFN	0.4125	0.0000	-0.6512	-0.2357	-0.1186

TABLE 5. Accuracy, loss, recall and precision of the proposed model and state-of-the-art deep learning architectures in different datasets with the respective average (between 0 and 1).

	Accuracy				
Model	FD-NT	FD-TL	FD-MV	FD-ST	Average
Proposed	0.8824	0.7450	0.7584	0.6324	0.7545
VGG16	0.8529	0.7300	0.5337	0.6691	0.6964
InceptionV3	0.8088	0.6300	0.6517	0.6985	0.6973
Xception	0.8382	0.5425	0.7528	0.6471	0.6952
MobileNetV2	0.8235	0.7925	0.6404	0.7132	0.7424
	Loss				
Model	FD-NT	FD-TL	FD-MV	FD-ST	Average
Proposed	0.3375	0.7167	0.5243	1.0006	0.6448
VGG16	0.4352	1.2383	2.8711	1.0478	1.3981
InceptionV3	0.4509	1.9270	2.6300	1.0756	1.5209
Xception	0.4444	2.3983	1.5487	1.2664	1.4145
MobileNetV2	0.4359	0.4460	1.0297	0.6990	0.6527
	Recall				
Model	FD-NT	FD-TL	FD-MV	FD-ST	Average
Proposed	0.9706	0.6200	0.9775	0.9706	0.8847
VGG16	0.8529	0.6900	1.0000	0.8382	0.8453
InceptionV3	0.7941	0.2950	0.9775	0.7206	0.6968
Xception	0.8235	0.1050	0.9888	0.6765	0.6484
MobileNetV2	0.8235	0.7750	0.9775	0.7206	0.8242
	Precision				
Model	FD-NT	FD-TL	FD-MV	FD-ST	Average
Proposed	0.8250	0.8267	0.6797	0.5789	0.7276
VGG16	0.8529	0.7500	0.5174	0.6264	0.6867
InceptionV3	0.8182	0.8939	0.5918	0.6901	0.7485
Xception	0.8485	0.8400	0.6718	0.6389	0.7498
MobileNetV2	0.8235	0.8031	0.5839	0.7101	0.7302

the algorithms are not even very robust. For example, the two LBP algorithms have a perfect classification in the FD-MV dataset, but in the FD-ST dataset they cannot distinguish the differences between a defect and non-defective example. This is shown by the Mathews Correlation Coefficient metric, which says that if a value is close to zero, the system is almost a random classifier. In Tables 5, 6 the proposed CNN is compared to the deep learning state-of-the-art architectures. VGG16, InceptionV3, Xception and MobileNetV2 were the ones used for comparison. The results in Tables 5, 6 show that the proposed CNN outperforms other deep learning architectures in almost all test metrics except precision and FN. From all the deep learning architectures, the simpler one (MobileNetV2) is the one that comes closer in terms of average accuracy (0.7424 in comparison with the 0.7545 of the proposed CNN). Finally, the defect detection execution time of a sample image, for each of the studied models, is presented in Table 7, which shows the good performance of the proposed model.

3) FALSE NEGATIVE REDUCTION METHODS IMPROVE SYSTEMS PERFORMANCE

As previously mentioned, two false negative reduction methods were studied, namely CTR and RR. Both methods are based on classification threshold value modification. The FD-NT dataset (validation part) was used to obtain the best threshold value for both studies. After obtaining the threshold values, both methods were tested on the test datasets and the results are presented in Tables 8, 9. It is important to note that these results consider the operator analysis. This means

TABLE 6. AUC, FN, f2-score and MCC of the proposed model and state-of-the-art deep learning architectures in different datasets with the respective average (between 0 and 1).

	AUC				
Model	FD-NT	FD-TL	FD-MV	FD-ST	Average
Proposed	0.9191	0.8197	0.8597	0.7098	0.8271
VGG16	0.9040	0.7492	0.6511	0.6530	0.7478
InceptionV3	0.9217	0.7933	0.7157	0.7980	0.7948
Xception	0.9079	0.7902	0.7674	0.6756	0.8038
MobileNetV2	0.9061	0.8804	0.4977	0.7888	0.7536
	FN				
Model	FD-NT	FD-TL	FD-MV	FD-ST	Average
Proposed	1	76	2	2	20.25
VGG16	5	62	0	11	19.5
InceptionV3	7	141	2	19	42.25
Xception	6	179	1	22	52
MobileNetV2	6	45	2	19	18
	F2-Score				
Model	FD-NT	FD-TL	FD-MV	FD-ST	Average
Proposed	0.9375	0.6526	0.8988	0.8549	0.8360
VGG16	0.8529	0.7012	0.8428	0.7851	0.7955
InceptionV3	0.7988	0.3406	0.8648	0.7143	0.6796
Xception	0.8284	0.1273	0.9035	0.6686	0.6319
MobileNetV2	0.8235	0.7805	0.8614	0.7185	0.7960
	Mathews Correlation Coeficient				
Model	FD-NT	FD-TL	FD-MV	FD-ST	Average
Proposed	0.7769	0.5061	0.5750	0.3594	0.5543
VGG16	0.7059	0.4615	0.1868	0.3594	0.4284
InceptionV3	0.6179	0.3502	0.4000	0.3974	0.4414
Xception	0.6768	0.1756	0.5735	0.2946	0.4301
MobileNetV2	0.6471	0.5854	0.3803	0.4265	0.5098

TABLE 7. Defect detection execution time of each model for one sample image.

Model	Time (ms)
LBP+SVM	7.24
GCLM+SVM	9.53
LBP+FFN	7.23
GLCM+FFN	9.52
VGG16	27.79
InceptionV3	30.90
Xception	15.68
MobileNetV2	13.78
Proposed	**5.69**

that in theory every example reviewed by the operator is correctly classified. This is because in this system, unlike the traditional system, the operator is reviewing the detected defect while the system is stationary, and he knows the location of that defect. The results from Tables 8, 9 show that the CTR method is the most effective one, as it was expected, because of the greater cooperation from the operator. In comparison, the RR method obtains less accuracy given the decreased number of examples reviewed by the operator. Nevertheless, there is a sharp decrease in the operator's work, which in the case of inspecting cheaper fabric might be a better solution. This approach allows the operator to focus only on a smaller and important portion of the inspected fabric. Overall, from a total of 782 test examples, when CTR is applied, the system achieves an average of 95% accuracy with the operator intervention in just 42% of the inspected fabric. When RR is applied, with only 30% of operator's intervention, an average accuracy of 80%

TABLE 8. Number of TP, TN, FP, FN and examples reviewed by the operator for false negative reduction methods in different datasets with the respective average.

	TP				
Method	FD-NT	FD-TL	FD-MV	FD-ST	Average
Standart	33	124	87	66	77.5
CTR	33	141	88	66	82
RR	33	141	88	66	82
	TN				
Method	FD-NT	FD-TL	FD-MV	FD-ST	Average
Standart	34	200	89	68	97.75
CTR	34	200	89	68	97.75
RR	30	183	57	27	74.25
	FP				
Method	FD-NT	FD-TL	FD-MV	FD-ST	Average
Standart	0	0	0	0	0
CTR	0	0	0	0	0
RR	4	17	32	41	23.5
	FN				
Method	FD-NT	FD-TL	FD-MV	FD-ST	Average
Standart	1	76	2	2	20.25
CTR	1	59	1	2	15.75
RR	1	59	1	2	15.75
	Examples Reviewed by the Operator				
Method	FD-NT	FD-TL	FD-MV	FD-ST	Average
Standart	33	124	87	66	77.5
CTR	33	141	88	66	82
RR	29	124	56	25	58.5

TABLE 9. Accuracy, recall, precision, f2-score and MCC of the system with false negative reduction methods in different datasets with the respective average (between 0 and 1).

	Accuracy				
Method	FD-NT	FD-TL	FD-MV	FD-ST	Average
Standart	0.9853	0.8100	0.9888	0.9853	0.9423
CTR	0.9853	0.8525	0.9944	0.9853	0.9544
RR	0.9265	0.8100	0.8146	0.6838	0.8087
	Recall				
Method	FD-NT	FD-TL	FD-MV	FD-ST	Average
Standart	0.9706	0.6200	0.9775	0.9706	0.8847
CTR	0.9706	0.7050	0.9888	0.9706	0.9087
RR	0.9706	0.7050	0.9888	0.9706	0.9087
	Precision				
Method	FD-NT	FD-TL	FD-MV	FD-ST	Average
Standart	1.0000	1.0000	1.0000	1.0000	1.0000
CTR	1.0000	1.0000	1.0000	1.0000	1.0000
RR	0.8919	0.8924	0.7333	0.6168	0.7836
	F2-Score				
Method	FD-NT	FD-TL	FD-MV	FD-ST	Average
Standart	0.9763	0.6710	0.9819	0.9763	0.9014
CTR	0.9763	0.7492	0.9910	0.9763	0.9232
RR	0.9538	0.7359	0.9244	0.8707	0.8712
	Mathews Correlation Coeficient				
Method	FD-NT	FD-TL	FD-MV	FD-ST	Average
Standart	0.9710	0.6703	0.9778	0.9710	0.8975
CTR	0.9710	0.7378	0.9888	0.9710	0.9172
RR	0.8563	0.6341	0.6712	0.4488	0.6526

can be attained. However, considering that in a production environment the number of defects is much lower than the number of defect-free regions, the percentage of examples reviewed by the operator will be much lower. In some fabric rolls there may not even exist defects. In this case only FP cases will be reviewed by the operator and given the low FP rate this can lead to no intervention from the operator at all. Therefore, the time spent by the operator in this system will always be much shorter than in a traditional inspection system.

D. DISCUSSION

The state-of-the-art study compares the proposed model with four different fabric defect detection approaches and four different deep learning architectures. The comparisons presented in Tables 3, 4, 5 and 6 show that the proposed CNN model not only outperforms human inspection performance, but it is also able to generalize better and outperform all other tested approaches. Even though for specific datasets, mainly in the FD-MV dataset, the proposed method does not always have the best results, in the average of all datasets (with a wide variety of defects) it reaches the best results. Furthermore, the proposed approach is also much faster in comparison with other deep learning approaches (Table 7), due to its simple architecture. The results also show that some of these algorithms cannot generalize very well, for example the LBP approach obtains perfect accuracy in FD-MV dataset even though it cannot distinguish between the two classes in the FD-ST dataset. Even though the deep learning state-of-the-art models were built with transfer learning from ImageNet, it is possible to assume that with more training examples a better performance would be obtained. So, as this problem does not have a lot of prepared data, the proposed CNN is simpler and can outperform even the less complex MobileNetV2.

VI. CONCLUSION AND FUTURE WORK

Overall this system shows the potential of operator-assisted systems, which may be easier to implement, low-cost and better to tackle realistic scenarios. In this work, a new CNN-based fabric defect detection system, suited for a realistic scenario, was proposed. The CNN method provides a good feature detection, as it can be confirmed in Section V-C. This system has the possibility of being operator-assisted whom may confirm or reverse the system evaluation when a possible defect is detected. This increases the system accuracy by reducing the number of FP examples. Therefore, two FN reduction methods were studied. To obtain a reliable dataset that could represent most of the fabric defect types found in the literature, a new dataset was created. In total four different datasets were used to train and test the proposed methods. An average of 75% accuracy was observed in the test datasets when the system was in automatic mode. Therefore, this system outperforms human inspection systems even in automatic mode. When the false negative reduction method was applied, together with the operator intervention, an average of 95% accuracy was attained in the same datasets. These promising results show that the proposed system achieves better performance when compared to traditional systems and others found in literature, in addition to being much faster, cheaper and easier to maintain. Plus, the time spent by the operator in this system will always be much shorter than in a traditional inspection system. Furthermore, this thorough study using four different datasets, with distinct characteristics, is uncommon in known literature and shows the generalization capacity of this model.

As future work, it would be of interest to develop a larger dataset, with more real-world examples that would allow to train and implement a more complex defect detection system. A larger dataset would allow the use of more complex models, such as Long Short-Term Memory (LSTM) and Transformer networks. To improve the FN reduction functionality, more robust methods that can be executed during training, such as a custom loss function, could be tested. As a final objective, the developed system should be implemented in a real-world environment, to test and compare it to a traditional inspection system.

ACKNOWLEDGMENT

J. P. Matos-Carvalho would like to thank COPELABS for the support and commitment.

REFERENCES

[1] K. Srinivasan, P. H. Dastoor, P. Radhakrishnaiah, and S. Jayaraman, "FDAS: A knowledge-based framework for analysis of defects in woven textile structures," *J. Textile Inst.*, vol. 83, no. 3, pp. 431–448, Jan. 1992, doi: 10.1080/00405009208631217.

[2] Y. F. Zhang and R. R. Bresee, "Fabric defect detection and classification using image analysis," *Textile Res. J.*, vol. 65, no. 1, pp. 1–9, Jan. 1995, doi: 10.1177/004051759506500101.

[3] B. N. Nickolay and H. Schmalfub, "Automatic fabric inspection—Utopia or reality?" *Mellind Textilberichte*, vol. 73, pp. 33–37, Jan. 1993.

[4] A. Kumar, "Computer-vision-based fabric defect detection: A survey," *IEEE Trans. Ind. Electron.*, vol. 55, no. 1, pp. 348–363, Jan. 2008.

[5] E. J. Wood, "Applying Fourier and associated transforms to pattern characterization in textiles," *Textile Res. J.*, vol. 60, no. 4, pp. 212–220, Apr. 1990.

[6] M. Unser and F. Ade, "Feature extraction and decision procedure for automated inspection of textured materials," *Pattern Recognit. Lett.*, vol. 2, no. 3, pp. 185–191, Mar. 1984.

[7] A. Monadjemi, M. Mirmehdi, and B. Thomas, "Restructured eigenfilter matching for novelty detection in random textures," in *Proc. Brit. Mach. Vis. Conf.*, Sep. 2004, pp. 637–646, doi: 10.5244/C.18.66.

[8] A. Monadjemi, "Towards efficient texture classification and abnormality detection," Ph.D. dissertation, Univ. Bristol, Bristol, U.K., Oct. 2004. [Online]. Available: https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwjLhY7JzYDxAhVhyoUKHRWbCvQQFjACegQIChAD&url=https%3A%2F%2Fciteseerx.ist.psu.edu%2Fviewdoc%2Fdownload %3Fdoi%3D10.1.1.60.6140%26rep%3Drep1%26type%3Dpdf&usg=AOvVaw0TglZayx_2kdOzAdyzTvTv

[9] R. Thilepa and M. Thanikachalam, "A paper on automatic fabrics fault processing using image processing technique in MATLAB," *Signal Image Process., Int. J.*, vol. 1, no. 2, pp. 88–99, Dec. 2010.

[10] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-3, no. 6, pp. 610–621, Nov. 1973.

[11] U. Roesler, "Defect detection of fabrics by image processing," *Melliand Textilberichte, Int. Textile Rep. (German Ed.)*, vol. 73, no. 8, pp. 635–639, 1992.

[12] Y. B. Salem and S. Nasri, "Woven fabric defects detection based on texture classification algorithm," in *Proc. 8th Int. Multi-Conf. Syst. Signals Devices (SSD)*, 2011, pp. 1–5.

[13] F. Tajeripour, E. Kabir, and A. Sheikhi, "Fabric defect detection using modified local binary patterns," *EURASIP J. Adv. Signal Process.*, vol. 2008, no. 1, pp. 1–12, Dec. 2007, doi: 10.1155/2008/783898.

[14] C. Kwak, J. A. Ventura, and K. Tofang-Sazi, "Automated defect inspection and classification of leather fabric," *Intell. Data Anal.*, vol. 5, no. 4, pp. 355–370, Nov. 2001, doi: 10.3233/IDA-2001-5406.

[15] K. L. Mak, P. Peng, and K. F. C. Yiu, "Fabric defect detection using morphological filters," *Image Vis. Comput.*, vol. 27, no. 10, pp. 1585–1592, Sep. 2009, doi: 10.1016/j.imavis.2009.03.007.

[16] K. L. Mak, P. Peng, and H. Y. K. Lau, "Optimal morphological filter design for fabric defect detection," in *Proc. IEEE Int. Conf. Ind. Technol.*, Dec. 2005, pp. 799–804.

[17] A. Conci and C. B. Proença, "A fractal image analysis system for fabric inspection based on a box-counting method," *Comput. Netw. ISDN Syst.*, vol. 30, nos. 20–21, pp. 1887–1895, Nov. 1998.

[18] L. M. Hoffer, F. Francini, B. Tiribilli, and G. Longobardi, "Neural networks for the optical recognition of defects in cloth," *Opt. Eng.*, vol. 35, pp. 3183–3190, Nov. 1996, doi: 10.1117/1.601057.

[19] D.-M. Tsai and C.-Y. Hsieh, "Automated surface inspection for directional textures," *Image Vis. Comput.*, vol. 18, no. 1, pp. 49–62, Dec. 1999.

[20] A. C. Bovik, M. Clark, and W. S. Geisler, "Multichannel texture analysis using localized spatial filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 1, pp. 55–73, Jan. 1990.

[21] H. Sari-Sarraf and J. S. Goddard, "Vision system for on-loom fabric inspection," *IEEE Trans. Ind. Appl.*, vol. 35, no. 6, pp. 1252–1259, Nov. 1999.

[22] J. L. Dorrity and G. Vachtsevanos, "On-line defect detection for weaving systems," in *Proc. IEEE Annu. Textile, Fiber Film Ind. Tech. Conf.*, May 1996, pp. 1–6.

[23] K. A. Yildiz and Buldu, "Wavelet transform and principal component analysis in fabric defect detection and classification," *Pamukkale Univ. J. Eng. Sci.-Pamukkale Universitesi Muhendislik Bilimleri Dergisi*, vol. 23, no. 5, pp. 622–627, 2017.

[24] B. H. McCormick and S. N. Jayaramamurthy, "Time series model for texture synthesis," *Int. J. Comput. Inf. Sci.*, vol. 3, no. 4, pp. 329–343, 1974.

[25] F. S. Cohen, Z. Fan, and S. Attali, "Automated inspection of textile fabrics using textural models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 8, pp. 803–808, Aug. 1991.

[26] Q. Luo, X. Fang, Y. Sun, L. Liu, J. Ai, C. Yang, and O. Simpson, "Surface defect classification for hot-rolled steel strips by selectively dominant local binary patterns," *IEEE Access*, vol. 7, pp. 23488–23499, 2019, doi: 10.1109/ACCESS.2019.2898215.

[27] J. P. Matos-Carvalho, F. Moutinho, A. B. Salvado, T. Carrasqueira, R. Campos-Rebelo, D. Pedro, L. M. Campos, J. M. Fonseca, and A. Mora, "Static and dynamic algorithms for terrain classification in UAV aerial imagery," *Remote Sens.*, vol. 11, no. 21, p. 2501, Oct. 2019, doi: 10.3390/rs11212501.

[28] D. Wang, L. Xie, S. Yang, and F. Tian, "Support vector machine optimized by genetic algorithm for data analysis of near-infrared spectroscopy sensors," *Sensors*, vol. 18, no. 10, p. 3222, Sep. 2018, doi: 10.3390/s18103222.

[29] M. Atwya and G. Panoutsos, "Transient thermography for flaw detection in friction stir welding: A machine learning approach," *IEEE Trans. Ind. Informat.*, vol. 16, no. 7, pp. 4423–4435, Jul. 2020, doi: 10.1109/TII.2019.2948023.

[30] R. Wang, Q. Guo, S. Lu, and C. Zhang, "Tire defect detection using fully convolutional network," *IEEE Access*, vol. 7, pp. 43502–43510, 2019, doi: 10.1109/ACCESS.2019.2908483.

[31] K. Muhammad, J. Ahmad, I. Mehmood, S. Rho, and S. W. Baik, "Convolutional neural networks based fire detection in surveillance videos," *IEEE Access*, vol. 6, pp. 18174–18183, 2018, doi: 10.1109/ACCESS.2018.2812835.

[32] P. Jiang, Y. Chen, B. Liu, D. He, and C. Liang, "Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks," *IEEE Access*, vol. 7, pp. 59069–59080, 2019, doi: 10.1109/ACCESS.2019.2914929.

[33] L. M. Azizah, S. F. Umayah, S. Riyadi, C. Damarjati, and N. A. Utama, "Deep learning implementation using convolutional neural network in mangosteen surface defect detection," in *Proc. 7th IEEE Int. Conf. Control Syst., Comput. Eng. (ICCSCE)*, Penang, Malaysia, Nov. 2017, pp. 242–246, doi: 10.1109/ICCSCE.2017.8284412.

[34] A. Ghosh, T. Guha, R. B. Bhar, and S. Das, "Pattern classification of fabric defects using support vector machines," *Int. J. Clothing Sci. Technol.*, vol. 23, nos. 2–3, pp. 142–151, Jun. 2011, doi: 10.1108/09556221111107333.

[35] A. Kumar, "Neural network based detection of local textile defects," *Pattern Recognit.*, vol. 36, no. 7, pp. 1645–1659, Jul. 2003, doi: 10.1016/S0031-3203(03)00005-0.

[36] S. Mei, Y. Wang, and G. Wen, "Automatic fabric defect detection with a multi-scale convolutional denoising autoencoder network model," *Sensors*, vol. 18, no. 4, p. 1064, Apr. 2018, doi: 10.3390/s18041064.

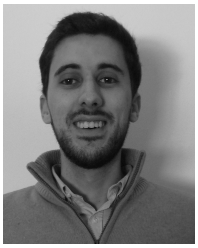
[37] W. Ouyang, B. Xu, J. Hou, and X. Yuan, "Fabric defect detection using activation layer embedded convolutional neural network," *IEEE Access*, vol. 7, pp. 70130–70140, 2019, doi: 10.1109/ACCESS.2019.2913620.

[38] (1996). *Texture Analysis of the DFG's TILDA Textile Texture-Database*. [Online]. Available: https://lmb.informatik.uni-freiburg.de/resources/datasets/tilda.en.html

[39] MVTec Software Company. *MVTec Anomaly Detection Dataset (MVTec AD)*. Accessed: Nov. 30, 2020. [Online]. Available: https://www.mvtec.com/company/research/datasets/mvtec-ad/

[40] P. Pathirana. *Fabric Stain Dataset*. Accessed: Nov. 30, 2020. [Online]. Available: https://www.kaggle.com/priemshpathirana/fabric-stain-dataset.

[41] Cottoninc. *Standard Fabric Defect Glossary*. Accessed: Nov. 30, 2020. [Online]. Available: https://www.cottoninc.com/quality-products/textile-resources/fabric-defect-glossary/
[42] C.-S. Cho, B.-M. Chung, and M.-J. Park, "Development of real-time vision-based fabric inspection system," *IEEE Trans. Ind. Electron.*, vol. 52, no. 4, pp. 1073–1079, Aug. 2005.
[43] M. Abadi *et al.* (2015). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. [Online]. Available: http://tensorflow.org/. Software available from tensorflow.org
[44] F. Chollet. (2015). *Keras*. [Online]. Available: https://github.com/fchollet/keras



TOMÁS ALMEIDA was born in Lisbon, Portugal, in 1996. He received the M.S. degree in electrical and computer engineering from the NOVA School of Science and Technology, in 2021. He is a Software Engineer and holds more than three years of experience in the field. His main research interests include machine learning applications and software development.

FILIPE MOUTINHO received the Engineering, M.Sc., and Ph.D. degrees in electrical and computer engineering from the NOVA School of Science and Technology (FCT NOVA), Portugal, in 2003, 2009, and 2014, respectively. He was a Software Engineer with NewHotel Software, Lisbon. He was a researcher in several research projects, among which are the Portuguese project with the designation Petri-Rig (Ref. PTDC/EEI-AUT/2641/2012) and the European Projects with the designations Arrowhead (Grant Agreement 332987), Productive 4.0 (Grant Agreement 737459), and Arrowhead Tools (Grant Agreement 826452). He is currently an Assistant Professor with the Department of Electrical and Computer Engineering, FCT NOVA, and a Researcher at the Centre of Technology and Systems (CTS), UNINOVA.

J. P. MATOS-CARVALHO was born in Almada, Portugal, in 1993. He received the M.Sc. degree (Hons.) in electrical and computer engineer from the NOVA School of Science and Technology (FCT NOVA), Portugal, in 2017, and the Ph.D. degree in electrical and computer engineering, in 2021. He has been working in the aerial robotics research and development field, since 2016. He is currently an Assistant Professor with the Universidade Lusófona de Humanidades e Tecnologias, Lisbon. He received the Best Paper Award "UAV Downwash Dynamic Texture Features for Terrain Classification on Autonomous Navigation" from the prestigious IEEE Conference, in 2018.