Multilevel Deep Learning Model for Fabric Classification and Defect Detection

Pranshu Goyal[1], Abhiroop Agarwal[1], Basavraj Chinagundi[1], and Prashant Singh Rana[1]

Thapar Institute of Engineering and Technology, Patiala, Punjab, India

Abstract. Detecting defects in fabrics is a difficult task as there are a lot of variations in the type of fabric and the defect itself. Many methods have been proposed to solve this problem, but their detection speed and accuracy were very low depending on the model being tested. To eliminate the variations and to improve the performance, we implemented multilevel modelling in our approach. This paper proposes an improved approach with higher accuracy for fabric defect detection, in which we compare the performance of various state-of-the-art deep learning models such as MobileNetV2, Xception, VGG19, and InceptionV3 and how their performance changes with the type of fabric. First, a Convolutional Neural Network model is used to classify the fabric into different types with an accuracy of 97.6%, and then on the basis of the type of fabric, the best model is used to detect defects in the fabric. This has a significant advantage in improving the overall performance of fabric defect detection. Further, K-Fold cross-validation has been performed to check the consistency of the proposed model.

Keywords: Fabric Defect Detection · Deep Learning · MaxPool · Multilevel Modelling · TensorFlow · ImageNet

1 Introduction

Texture assessment is exceptionally critical in the Textile Industry [1], including the detection of defects in fabrics. The nature of texture depends on indispensable processes of texture review to distinguish the deformities of fabric. The benefits of industrialism have been diminished because of texture imperfections causing repulsive losses. Conventional deformity discovery strategies are directed in numerous ventures by proficient human auditors who physically curate the fabric. In any case, such recognition strategies have a few weaknesses, such as fatigue, dreariness, carelessness, mistake, confusion, and tediousness which cause to diminish the finding of shortcomings. To address these shortcomings, various image processing techniques have been executed to naturally and productively distinguish and recognise texture defects.

Development and advancement of the sector normally bring to build the going through huge investment [2]. Be that as it may, the textile industry, like any other sector, experienced various issues. These include some insurance to diminish the effect of misfortunes that are budgetary, client disappointment, time squandering, and so on. Fabric defects are probably the greatest test confronting the textile business. Fabric is made in a day-by-day life utilizing fibers and a usually utilized material. Most fabrics are delivered after passing through a series of making stages. Various machines and methods are utilized during the making stages.

Fabrics are subjected to pressures and stresses along these lines that cause defects. As indicated by their structures and directions, defects take various names. The textile business has distinguished in more than 70 types of defects such as laddering, end-out, holes, and oil spots. Unexpected tasks might be the reason for various defects on the fabric surface during the manufacturing of fabric. Fabric manufacturing is one of the largest traditional businesses where fabric inspection systems can play a vital role in growing the manufacturing rate. The process of inspection is really important in any manufacturing procedure, especially from the viewpoint of an industrialist. The idea of the inspection process is to recognize the errors or defects, on the off chance that any exist, and at that point to give an alert to the inspector to check the manufacturing procedure and remove the defective products.

For the most part, fabric defect recognition utilizes two kinds of investigation models. The essential one is the human-based inspection systems [2]. The second framework is automated-based inspection systems. Accordingly, human-based defect detection done by specialists' turns out to be rapidly a mind-boggling and fussy task. In this manner, having proficient and automated-based frameworks nearby is a significant necessity for improving unwavering quality and accelerating quality control, which may expand profitability. The subject of automated-based defect detection has been examined in a few works in the most recent decades. In spite of the fact that there is no widespread methodology for handling this issue, a few strategies dependent on image processing procedures have been proposed in recent years. These strategies were utilized to recognize defects at the image level, so the precision rate is little, and additionally, it is hard to find the defects precisely. In this way, they can't be stretched out to various fabrics. As of late, some different techniques dependent on the local image-level have been proposed [3], which utilize the base unit as the fundamental activity object to extract image features.

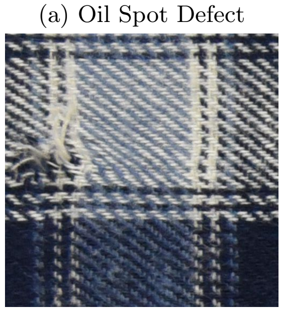
Recognizing defects in fabrics is especially difficult due to a large number of variables. As discussed in the paragraph above, there are many different types of defects, but these defects are different for different types of fabrics. During the course of our research, we tried to find the correlation between the fabric type and its defect detection accuracy for various state-of-the-art deep learning models. We were able to see that certain models performed better for certain types of fabric. Hence we proceeded further to first determine the type of fabric (Type A, Type B, Type C, Type D) using a CNN model, and then we identify whether it has defects or not.

2 Defects in Fabric

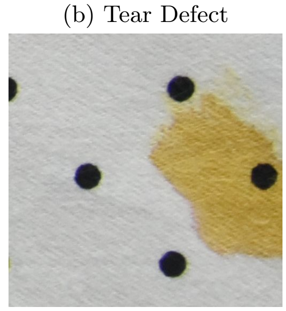
In order to prepare various categories and forms of fabric items in the industry, fabric materials are used. Consequently, yarn quality and/or loom defects affect the fabric quality. Fabric defect has been estimated that the price of fabrics is reduced by 45%-65% [4,5] due to the presence of defects such as dye mark/dye Spot, slack warp, faulty pattern card, holes, spirality, grease oil/ dirty stains, mispick, slub, wrong end, slack end, and so on. In a fabric, defects can occur due to: machine faults, color bleeding, yarn problems, excessive stretching, hole, dirt spot, scratch, poor finishing, crack points, material defects, processing defects, and so on. The textile business has distinguished more than 70 types of defects [3] such as shear, tear, hole, and oil spot, as shown in Figure 1 below.

(a) Oil Spot Defect

(b) Tear Defect



(c) Shear Defect



(d) Paint Spot Defect

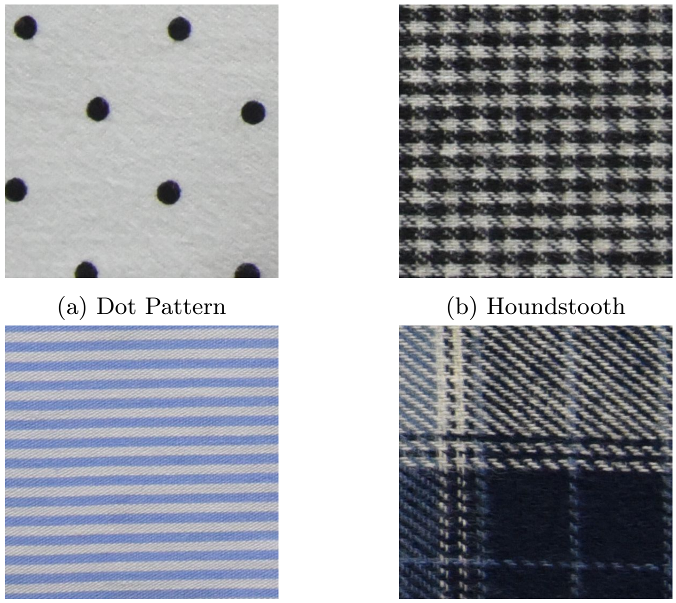
Fig. 1: Types of Defects in Fabrics

Hence it is very difficult to identify the exact defect and classify it into types. We focussed on classifying whether the fabric has defects or not for the course of this research. To do this, we used various state-of-the-art deep learning models like VGG 19, Xcpetion, Inception V3, and MobileNet V2.

3 Fabric Classification

Not only are there different types of defects, but there are also different types of fabrics that one comes across in the textile industry. Fabric can be distinguished on the basis of pattern, design, and the weaving method.

Some of the different types of fabric are shown in Figure 2 below. As we can see, the fabric is broadly classified into four types on the basis of the pattern. These are- (1) Dot Pattern, (2) Thin Stripes, (3) Twill Plaid, and (4) Houndstooth. Each fabric has different sewing patterns, due to which the defects found in each of them be very different. Hence if we classify the fabric type first and then detect the defect, the prediction will be distinct and achieve better accuracy. These four types of fabrics will be classified using a CNN model, which we developed. These fabrics are taken from the ZJU-Leaper dataset [6] which is a benchmark dataset for fabric defect detection and comparative study. This dataset plans to set a benchmark suite for vision-based texture deformity recognition, including picture dataset, assessment convention, and pattern tests. Once we are able to identify the type of fabric, we can further analyze which deep learning model will give the user the best accuracy for detecting the defect.



(a) Dot Pattern

(b) Houndstooth

(c) Thin Stripes

(d) Twill Plaid

Fig. 2: Different Types of Fabrics

4 Proposed Methodology for Defect Detection

Traditional AI-based [7] defect detection methodologies can be ordered into three principle gatherings: Statistical, Structural methodology, and Model-based methodology. We will discuss these methodologies and how our approach differs from them. In the statistical approach, grey-level properties are utilized to describe the textural property of texture images which are called 1st-order statistics and higher-order statistics, separately. The 1st-order statistics can gauge the variance of grey-level intensity among different features between defective areas and background. The higher-order statistics depend on the joint probability distribution of pixel sets. In any case, the inconvenience of this strategy is that the defect size must be sufficiently enormous to empower a compelling estimation of the texture property. So the methodology is feeble in handling little local defects. Additionally, the calculation of higher-order statistics is tedious, not to mention that this approach require a high-quality image of the texture rather than the fabric as a whole. The structural approach is generally utilized on properties of the primitives of the defect-free fabric texture for the nearness of the flawed region and their related placement rules. Apparently, the practicability of this methodology is to congestion to those textures with regular macrotexture.

In the subsequent class model-based methodology, the generally utilized strategies are Markov random field and Gaussian Markov random field. The texture features a contemplated texture and can signify all the more exactly spatial interrelationships between the grey levels in the texture. However, like the methodologies based on second-order statistics, additionally, it is tough for a model-based methodology to deal with identifying small-sized defects in light of the fact that the methodologies, as a rule, require an adequately large region of the texture to assess the parameters of the models. The above-mentioned methodologies have their own advantages and disadvantages. We came up with a Multilevel Model-based approach consisting of a CNN model for categorizing the type of fabric, and then we selected the best deep learning model for detecting whether it has defects or not. As an intermediate step, we try to find a correlation between the type of fabric and the deep learning model, as some models are better at detecting defects in a particular type of fabric.

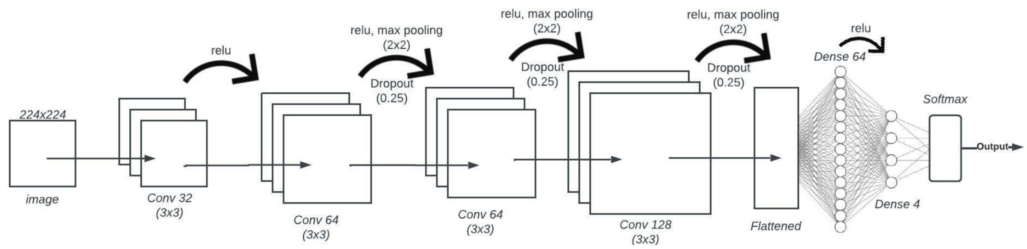


Fig. 3: CNN Architecture Diagram

Step 1- The dataset is prepared from the ZJU-Leaper dataset [6]. The dataset consists of four types of fabrics - Type A, Type B, Type C, and Type D. Each type of fabric has distinct patterns and characteristics. The dataset has a total of 2000 images, where each type of fabric has 500 images, out of which 250 are defective, and 250 are non-defective.
Step 2- A Convolutional Neural Network model is developed for the classification of the fabric. Figure 3 depicts the architecture of the CNN model.
Step 3- The proposed CNN model is trained on the dataset to predict the type of fabric. Hyperparameter tuning is done to achieve an accuracy of 97.6%.
Step 4 - Various state-of-the-art deep learning models- InceptionV3, Xception, VGG19, and MobileNetV2 were selected and then trained to predict if the fabric was defective or non-defective. Table I shows the number of parameters of each model.
Step 5 - The models are compared, and it is concluded that different models give more accurate predictions for different types of fabric. Table II shows the comparison between models.
Step 6 - Two dense layers and one GlobalAveragePooling layer are added to increase the accuracy of the models. Table III shows the improved accuracies of the models. It is observed that different models give more accurate predictions for different types of fabric.
Step 7 - A Multilevel machine learning model is proposed where a CNN model is used for fabric classification, and the best-suited state-of-the-art deep learning model is used to detect defects in the fabric.
Step 8 - K - Fold cross-validation is performed to verify the consistency of the CNN model and the best model selected for each type of fabric.

The flowchart of our proposed methodology is shown in Figure 4.

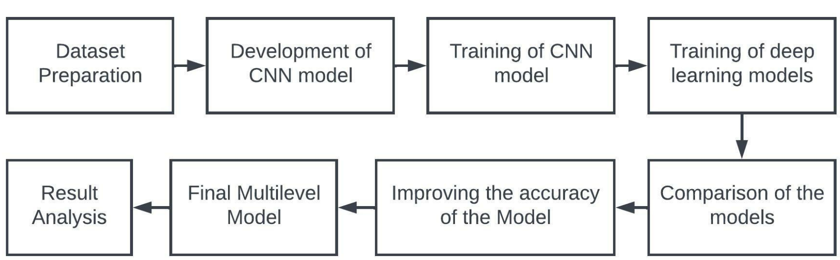


Fig. 4: Flowchart of the Proposed Methodology

5 Deep Learning Architectures

In the proposed approach, we have utilized profound convolutional brain networks in light of VGG (VGG19), GoogLeNet (Inception V3 and Xception) designs, and MobileNet (MobileNet V2) engineering, pre-prepared for fabric defect detection task on the given dataset.

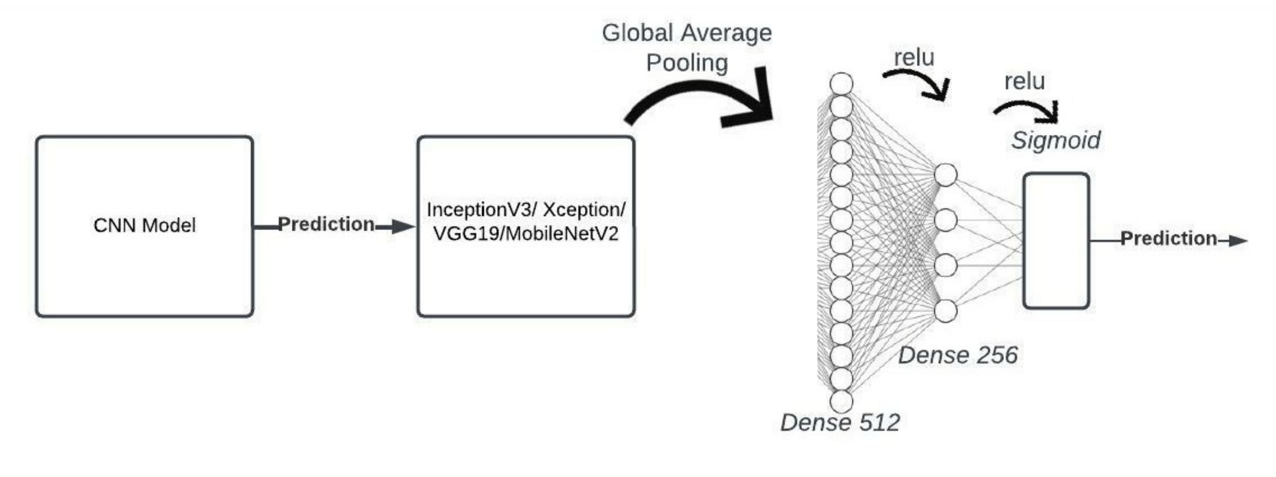


Fig. 5: Proposed Multilevel Model using Deep Learning Models

1) VGG Architecture: The VGG networks [8] with 16 layers (VGG16) and with 19 layers (VGG19) were the premise of the Visual Geometry Group (VGG) accommodation in the ImageNet Challenge 2014, where the VGG group got the first and the second places in the restriction and order tracks separately. The VGG engineering is organized, beginning with five blocks of convolutional layers followed by three completely associated layers. Convolutional layers utilize 3 × 3 parts with a step of 1 and cushioning of 1 to guarantee that every initiation map holds similar spatial aspects as the past layer. A redressed straight unit (ReLU) initiation is performed just after every convolution, and a maximum pooling activity is utilized toward the finish of each block to diminish the spatial aspect. Max pooling layers utilize 2 × 2 bits with a step of 2 and no cushioning to guarantee that each spatial component of the enactment map from the past layer is divided. Two completely associated layers with 4096 ReLU enacted units are then utilized before the last 1000 completely associated softmax layers. A drawback of the VGG19 models is that they are more costly to assess and utilize a ton of memory and boundaries. VGG19 has roughly 143 million boundaries. The greater part of these boundaries (around 100 million) are in the first completely associated layer, and it was since found that these completely associated layers could be taken out with no presentation downsize, altogether decreasing the number of vital boundaries.

2) GoogLeNet Architecture: The GoogLeNet [9] design was presented as GoogLeNet (Inception V1), later refined as Inception V2, and as of late as Inception V3. While Inception modules are thoughtfully convolutional highlight extractors, they exactly have all the earmarks of being fit for learning more extravagant portrayals with fewer boundaries. Conventional convolutional layer endeavors to learn channels in a 3D space, with 2 spatial aspects (width and level) and a channel aspect. In this way, a solitary convolution part is entrusted, all the while planning cross-channel relationships and spatial connections. The thought behind the Inception module is to make this cycle simpler and more proficient by unequivocally figuring it into a progression of tasks that would freely take a gander at cross-channel relationships and at spatial connections. The Xception [10] design is an expansion of the Inception engineering, which replaces the standard Inception modules with depth-wise detachable convolutions. Rather than parcelling input information into a few packed lumps, it maps the spatial relationships for each result channel independently and afterwards plays out a 1 × 1 depth-wise convolution to catch a cross-channel connection. This is basically identical to a current activity known as a "depth-wise distinct convolution," which comprises of a depth-wise convolution (a spatial convolution performed freely for each channel) trailed by a point-wise convolution (a 1×1 convolution across channels). Xception marginally beats InceptionV3 on the ImageNet dataset and incomprehensibly outflanks it on a bigger picture grouping dataset with 17,000 classes. In particular, it has a comparable number of boundaries as Inception V3, suggesting a

more prominent computational proficiency. Xception has 22,855,952 teachable boundaries while Inception V3 has 23,626,728 teachable boundaries.

3) MobileNet Architecture: MobileNetV2 [11] is a basic improvement over MobileNetV1 and pushes the bleeding edge for versatile visual affirmation, including gathering, object revelation, and semantic division. MobileNetV2 is conveyed as a part of the TensorFlow-Slim Image Classification Library, or you can start exploring MobileNetV2 promptly in Colaboratory. Then again, you can download the scratch pad and examine it locally using Jupyter. MobileNetV2 is also available as modules on TF-Hub. MobileNetV2 develops the considerations from MobileNetV1, including depth-wise separable convolution as viable design blocks. Regardless, V2 familiarizes two new features with the design: 1) direct bottlenecks between the layers and 2) backup course of action relationship between the bottlenecks.

Table 1: Number Of Parameters For Deep Learning Models

Deep Learning Model	***Number of Parameters***
MobileNetV2	3,608,678
Xception	22,998,606
VGG19	143,667,240
InceptionV3	23,885,392

6 Multilevel Modelling

The proposed framework consists of three components. (1) Data collection, (2) Fabric type classification using a Convolutional Neural Network and (3) Defect detection using the most suitable deep learning model for that type of fabric. The Block diagram of the proposed framework is shown in Figure 6. The framework takes 224x224 resolution of images, classifies the type of fabric and then uses the best state-of-the-art deep learning model to detect whether the fabric is defective or non-defective.

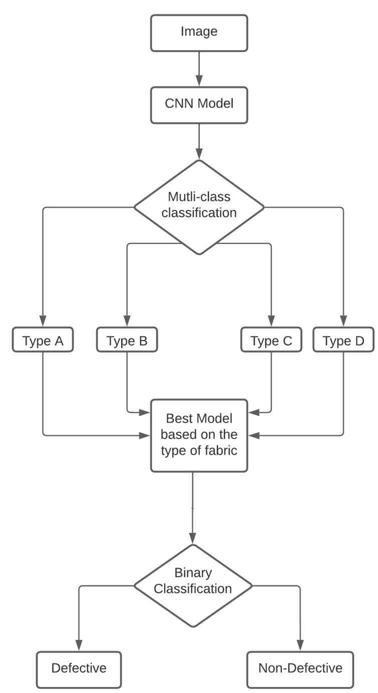


Fig. 6: Block Diagram of the Multilevel Model

Phase I: The first phase includes identifying four different types of fabric with distinct features and characteristics from the ZJU-Leaper dataset [6]. The dataset of 2000 images is then prepared, where each type of fabric has 500 images, out of which 250 are defective and 250 are non-defective. Training and testing data are generated in the ratio of 80:20, respectively. Phase II: The CNN model [12] is used for fabric classification and predicts the type of fabric with an accuracy of 97.6%. Phase III: on the basis of the predictions from Phase II, the best-suited model among MobileNetV2, Xception, VGG 19, and InceptionV3 is used to predict if the fabric is defective or non-defective. Figure 7 shows the entire workflow of the proposed methodology constituting the phases involved.

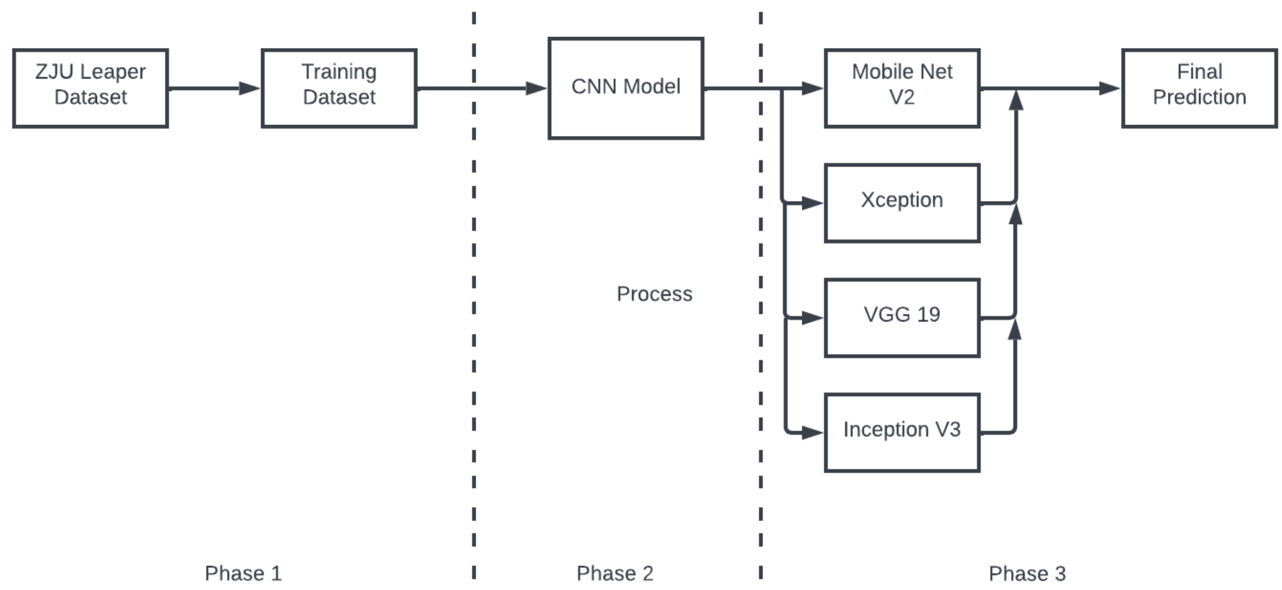


Fig. 7: Workflow of the Proposed Methodology

7 Model Evaluation

Various parameters such as accuracy [13] and K-fold cross-validation [14] are calculated to evaluate the performance of the proposed Multilevel model. The results are compiled in the form of tables and have been shown below. Repeated K-fold cross-validation has been performed to test the robustness of the model.

A. Model Evaluation Parameters Model evaluation parameters are calculated using the Accuracy achieved. Accuracy is the measure of the correctness of the classifier. Accuracy is computed as:

$$Accuracy = (TP + TN)/TotalData \tag{1}$$

Table 2: Comparison of Accuracy Of The Various Deep Learning Models For Different Type Of Fabrics

	Deep Learning Models			
	MobileNetV2	Xception	VGG19	InceptionV3
TypeA	**0.96**	0.95	0.92	0.95
TypeB	0.90	0.94	0.90	**0.95**
TypeC	0.74	**0.76**	0.68	0.72
TypeD	0.83	0.83	0.72	**0.91**

Table 3: Comparison of Improved Accuracy After Adding 2 Dense Layers and Global Average Pooling

	Deep Learning Models			
	MobileNetV2	Xception	VGG19	InceptionV3
TypeA	0.96	**0.97**	0.93	0.96
TypeB	0.92	0.93	0.93	**0.95**
TypeC	**0.80**	0.76	0.71	0.76
TypeD	0.86	0.89	0.72	**0.94**

B. K-Fold Cross Validation Estimating the accuracy of a classifier induced by supervised learning algorithms is important not only to predict its future prediction accuracy but also for choosing a classifier from a given set (model selection). To ensure that the proposed Multilevel model is consistent with low bias and low variance, repeated K-fold Cross Validation is performed. In this present work, 5-fold Cross Validation is repeated ten times. This process of K-fold cross-validation is applied to all four deep learning models and for the CNN model. Figure 8 shows a graph that is obtained by plotting the results of K-Fold Cross Validation against the accuracy for each model, and the overlapping lines signify that the proposed multilevel model is robust.

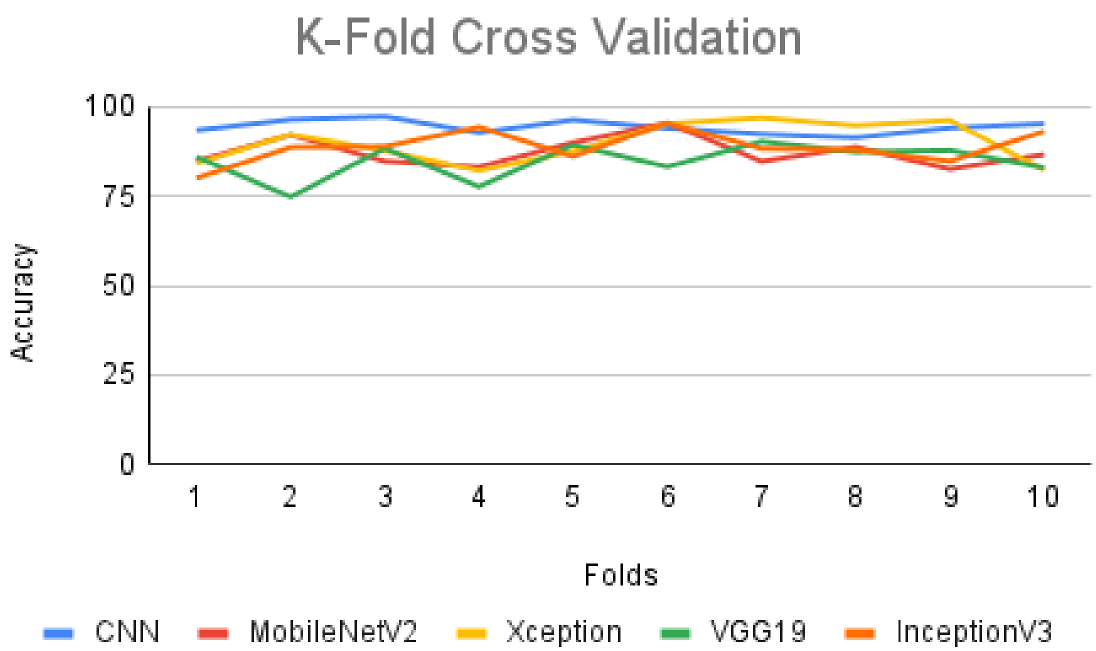


Fig. 8: Graph of K-Fold Cross Validation

8 Result Analysis

The models are trained on the training dataset and are further tested on the testing dataset. The Multilevel ensemble model is a combination of a CNN model and four deep learning models. The models are evaluated on various parameters, as mentioned above. The proposed CNN model has achieved an accuracy of 97.6% for fabric classification. Table 2 shows a comparison between the accuracies of different state-of-the-art models such as MobileNetv2, Xception, VGG19, and InceptionV3 for predicting defects in different types of fabrics. Table 3 shows that the performance of the models has been further improved by the proposed models. These tables help conclude that there is a correlation between the type of fabric and deep learning models as different models perform better for defect detection in different types of fabrics.

A problem that may occur while training is overfitting. To deal with the issues of overfitting, the model should be cross-validated, and if the resultant accuracy after various runs is consistent in all the runs, then the trained models are not overfitted. The accuracy is validated by applying 10-fold cross-validation five times, as shown in Figure 8. After the above analysis, we can conclude that the proposed model is not overfitted. Hence, the multilevel model approach of first classifying the type of fabric and then using the best-suited state-of-the-art model for defect detection provides a significant advantage in improving the overall performance of fabric defect detection.

9 Conclusion And Future Work

In today's world, Fabric Defect Detection can find numerous applications in eliminating wasteful production and improving cost-effectiveness. A Multilevel model for fabric detection is proposed, which combines a CNN model and four deep learning models (InceptionV3, Xception, VGG19, and MobileNetV2), which achieves an average accuracy of 97.6%. In the future, we intend to study different deep learning architectures like Artificial Neural Networks and stacked autoencoders for Fabric defect detection on a larger dataset and with higher computational power. We also intend to further classify the defects detected as a future scope for this research.

References

1. Lee Hok Siew, Robert M. Hodgson, and Errol J. Wood. Texture measures for carpet wear assessment. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(1):92–105, 1988.
2. TC Roland and CA Harlow. Automated visual inspection: A survey ieee transactions on pattern analysis and machine intelligence, vol, 1982.
3. RK Rao Ananthavaram, O Srinivasa Rao, and MHMK Prasad. Automatic defect detection of patterned fabric by using rb method and independent component analysis. *International Journal of Computer Applications*, 39(18):52–56, 2012.
4. P Sengottuvelan, A Wahi, and A Shanmugam. Automatic fault analysis of textile fabric using imaging systems. *Research Journal of Applied Sciences*, 3(1):26–31, 2008.
5. PH Dastoor, P Radhakrishnaiah, K Srinivasan, and S Jayaraman. Sdas: a knowledge-based framework for analyzing defects in apparel manufacturing. *Journal of the Textile Institute*, 85(4):542–560, 1994.
6. Chenkai Zhang, Shaozhe Feng, Xulongqi Wang, and Yueming Wang. Zju-leaper: A benchmark dataset for fabric defect detection and a comparative study. *IEEE Transactions on Artificial Intelligence*, 1(3):219–232, 2020.
7. Bertram Nickolay, Karl-Heinz Schicktanz, and Harald Schmalfuß. Automatische warenschau: Automatic textile inspection, 1993.
8. Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
9. Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.
10. François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017.
11. Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.
12. Xueqing Zhao, Min Zhang, and Junjun Zhang. Ensemble learning-based cnn for textile fabric defects classification. *International Journal of Clothing Science and Technology*, 2021.
13. Tzu-Tsung Wong and Po-Yang Yeh. Reliable accuracy estimates from k-fold cross validation. *IEEE Transactions on Knowledge and Data Engineering*, 32(8):1586–1594, 2019.
14. Ron Kohavi et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, volume 14, pages 1137–1145. Montreal, Canada, 1995.