A Report on
Binary Classification of Textured Fiber Images

Contents:

1. Introduction

This Report aims at the classification of Synthetic textured fiber images. Synthetic fiber images are designed to replicate the characteristics of real-world cotton fiber images which are generated using holographic imaging techniques. However, the labels assigned to these fiber images are generated using techniques that are not 100% reliable. This introduces ambiguity and uncertainty in the labeling process, making it challenging to assess the accuracy and reliability of the classification results.

To address these challenges and establish a reliable baseline for this classification problem, we have undertaken a project to create and classify synthetic fiber images using deep learning models. By generating these synthetic fiber images, we have full control over the image creation process, ensuring that the ground truth labels are accurate and reliable. This enables us to confidently train deep learning models on this controlled dataset and assess their performance without the limitations and uncertainties associated with the actual cotton fiber dataset.

To achieve accurate classification results and gain insights into the decision-making process of the models, we employed two state-of-the-art deep learning architectures: EfficientNet-B1 and VGG19. These models have demonstrated exceptional performance in various image classification tasks and are known for their ability to learn intricate patterns and features from images.

In addition to achieving accurate classification, we also aim to enhance the interpretability of our models. Deep learning models are often considered black boxes, as they lack transparency in how they arrive at their predictions. To address this, we utilize the Grad-CAM (Gradient-weighted Class Activation Mapping) technique, which generates heatmaps to visualize the regions of an image that are most influential in the model's decision-making process. By visualizing these heatmaps, we gain insights into the important features and regions of the images that contribute to the classification decisions of our models.

2. Generation of Data

A dataset comprising 33,400 images, with 16,700 images per class, was generated to replicate the actual image dataset of fibers. The images have a resolution of 500 x 500 pixels, providing a consistent format for training and evaluation.

2.1 Selection of Foreground and Background images: For both classes, a gray image was chosen as the background image. Additionally, two distinct wood texture images were utilized as foreground images for the respective classes. The following images display the foreground and background components:

Fig. 1: Texture Images

2.2 Generation of Synthetic Fiber Images:

Setup and Image Loading: In the initialization phase, various parameters are set, and the essential images for texture generation are loaded. Subsequently, a starting position and orientation (either vertical or horizontal) are randomly selected. The size of the fiber is also determined randomly, ensuring diversity in the generated textures. These initial configurations lay the foundation for creating synthetic fiber images with varying positions, orientations, and sizes.

Parameters: Final Image resolution - 500 x 500
Fiber Image resolution - 700 x 700
Number of Images - 16,700 for each class
Number of Fibers in an image - 1 to 3
Height of the strip - 3 to 7 pixels
Width of the strip - 30 to 70 pixels

Within each iteration, the process of creating individual fibers involves copying specific sections of the texture image and pasting them onto a black canvas. The following steps outline the procedure.

Random Rotation - The texture image is randomly rotated, introducing variability to the fibers' appearance.

Initial Fiber Creation - A small strip of the rotated texture image is selected and copied onto the black canvas based on the starting position and orientation (vertical or horizontal). If the orientation is vertical, the strip is pasted downwards; if horizontal, it is pasted to the right.

Iterative Fiber Creation - In the image generation process described, an important step involves the creation of individual fibers that collectively form the synthetic texture. During the creation of each fiber, a small strip of the texture image is selected and copied onto a black canvas called as **fiber image** as mentioned in Initial fiber creation. This forms the foundation of the fiber.
Below image shows the Initial fiber creation.

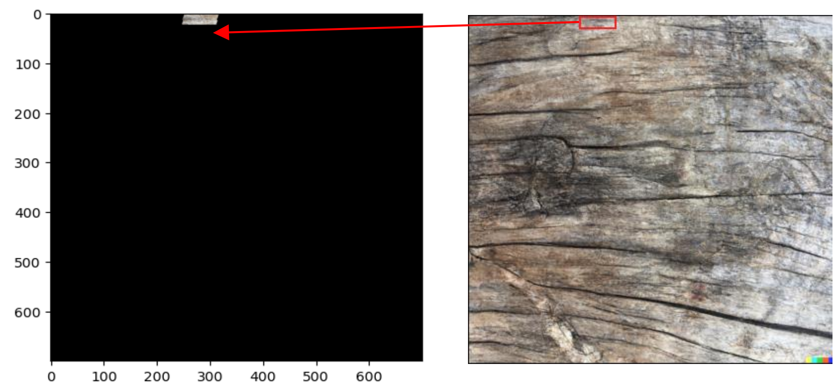


Fig. 2: Initial Fiber Creation

In the next iteration, we move down the existing strip and randomly copy the left or right strip from the existing position in the texture image and paste it on the black canvas in the exact same position as the texture image as explained in the below figure. In the example figure we moved down the existing strip and then moved left by a pixel and copied the strip from the texture image to black canvas. In the image shown below the shift in the red box in the texture image is exaggerated for visual representation.

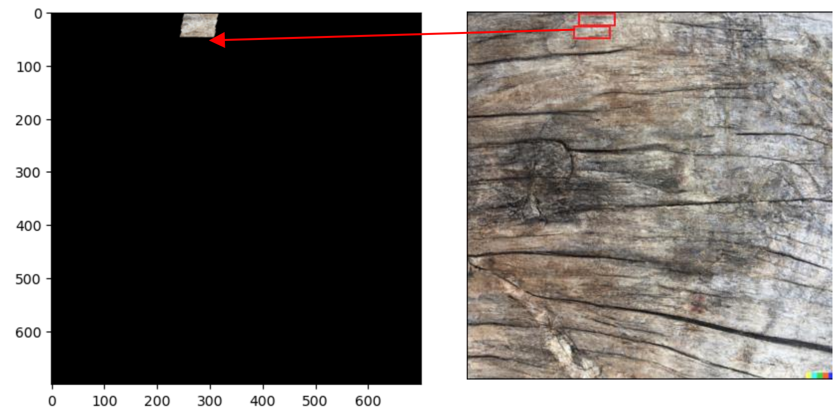


Fig. 3: step 1 of iterative Fiber Creation

Now iterating and pasting the strips onto black canvas continues until reaching the other end of the image. This generates the fiber image on the black canvas.

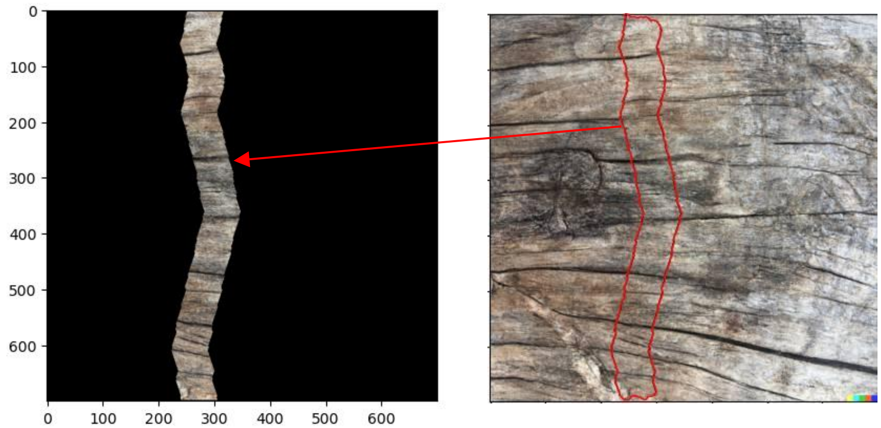


Fig. 3: Single fiber image after all iterations

In the next iteration, the texture image is rotated randomly in either 90º,180º or 270º and a new fiber is created using this texture image in the same way as above. This process continues until the number of fibers created equals the randomly generated parameter value “number of fibers in an image”. Each fiber is placed onto the black canvas, considering random shifts and rotations. If specified, additional noise will also be introduced to the fiber image. This step adds further variations and enhances the visual diversity of the synthetic fibers.

Below is an example gif showing single fiber creation.

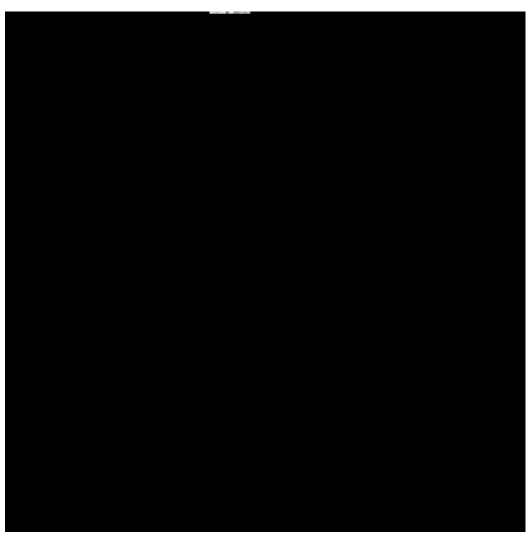


Fig. 4: Example GIF showing the creation of a single fiber in detail.

Example of Complete Fiber image creation.

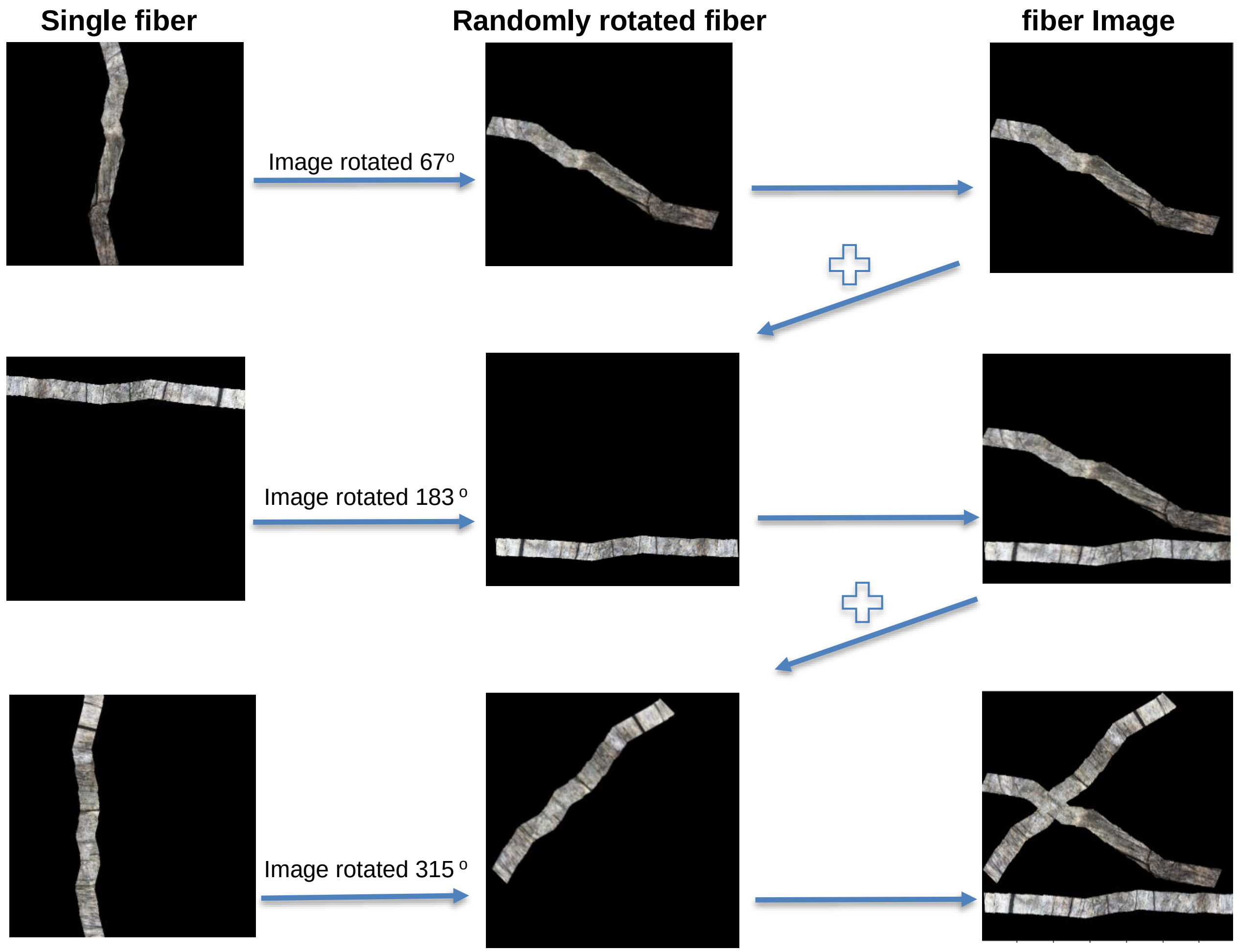


Fig. 5: Complete Fiber Image Creation (All images are rotated in counterclockwise direction).

Blending: After the generation of the **fiber image** with random shifts and rotations, the resulting image is then **cropped** to a size of 500 x 500 pixels to match final output image resolution.

Next, the cropped fiber image is blended with the selected background texture image. The blending process involves pasting the fiber pixels onto the background image while preserving the non-black pixels from the fibers.

To accomplish this, a new image is created with the same dimensions as the background texture image. Each pixel of the new image is determined by considering the corresponding pixels from both the fiber image and the background image. If a pixel in the fiber image is non-black, it is used as the pixel value in the new image. Otherwise, the pixel value from the background image is retained. This selective blending ensures that the fibers are seamlessly integrated into the background texture, creating a visually coherent synthetic fiber image.

This blending step is crucial for producing synthetic images, as it ensures that the fibers appear naturally embedded in the chosen background texture. By combining the cropped fiber image with the background, a final synthetic image is obtained. Noise is also added if specified to this image. The image is then grayed to form a black and white image. This is done to simulate the actual holographic cotton fiber images which are colorless.

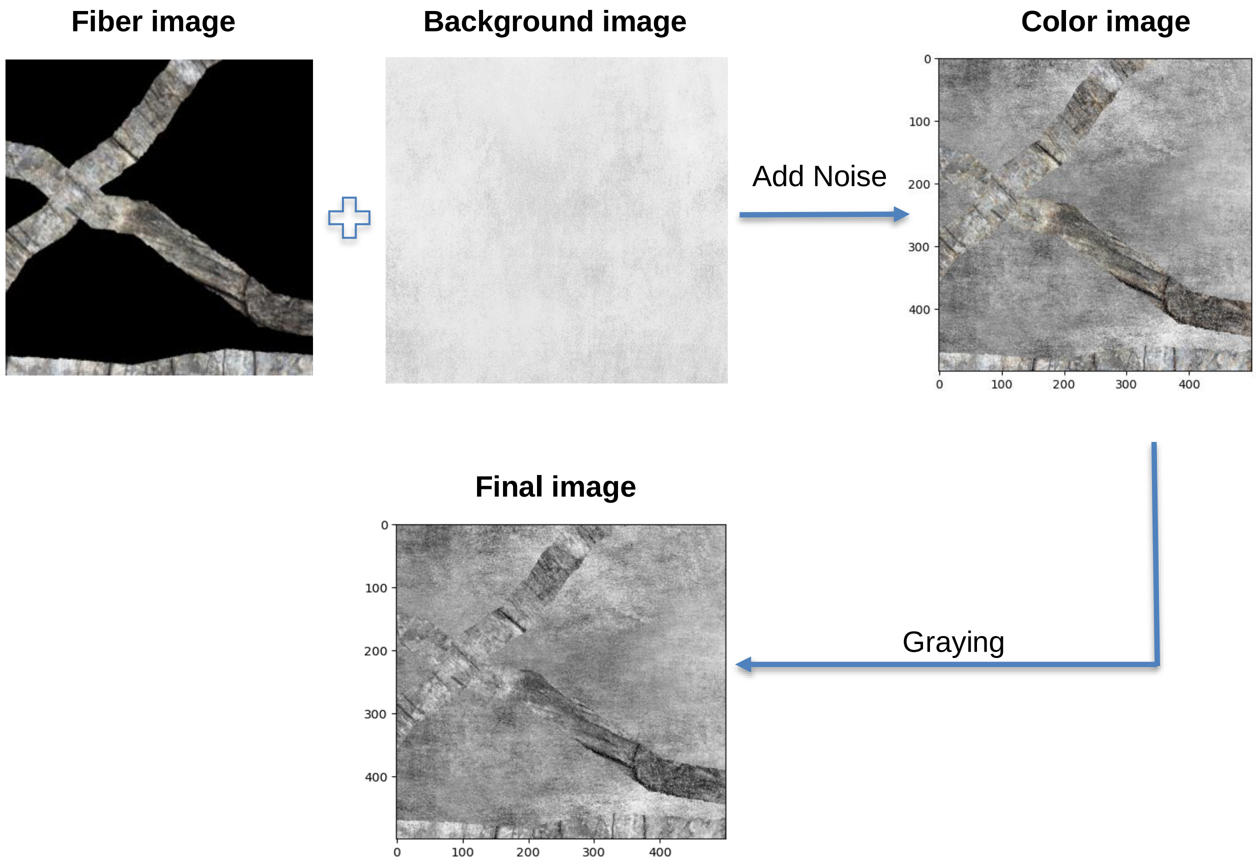


Fig. 5: Final texture Image creation.

Checking the Foreground ratio: After graying the image, various metrics related to the composition of the foreground and background are calculated and recorded. These metrics include the foreground ratio, background ratio, and the count of foreground pixels in the image. These metrics provide insights into the distribution and proportion of fibers within synthetic images.

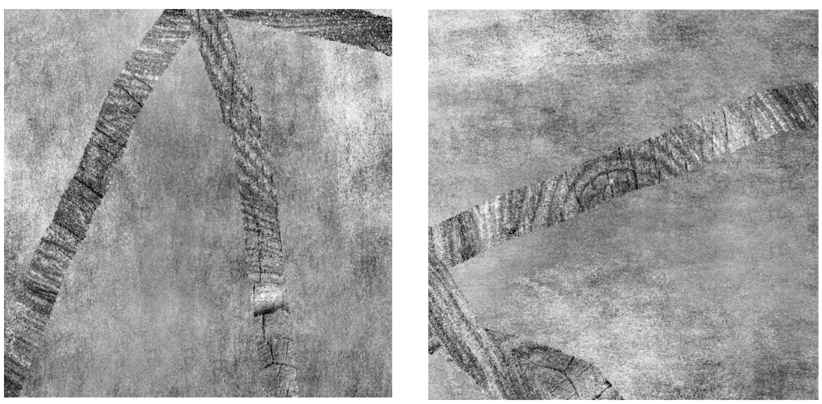
The foreground ratio is calculated by dividing the total number of foreground pixels (representing the fibers) by the total number of pixels in the image. Similarly, the background ratio is obtained by dividing the total number of background pixels by the total number of pixels in the image. These ratios help quantify the relative presence of fibers compared to the background texture.

The foreground pixel count represents the total number of pixels in the image that are part of the fibers. This count provides an estimation of the density or coverage of fibers within the synthetic image.
These metrics are then written to a text file, which serves as a record for further analysis and evaluation of the generated synthetic fiber dataset.

Additionally, a quality control check is performed to ensure the generated images meet certain criteria. If the foreground ratio is found to be below 5% of the total image area, it indicates that the image does not contain enough fibers and may not be representative of the desired dataset. In such cases, the image is discarded, and the image generation process is retried to create a new synthetic image.

On the other hand, if the foreground ratio meets the specified threshold, the image is considered valid and saved in the required directory for further use. This process continues iteratively until the desired number of images is generated, ensuring that a sufficient dataset of synthetic fiber images is created for subsequent analysis and model training.
Below are some sample images.

Class 0:



Class 1:

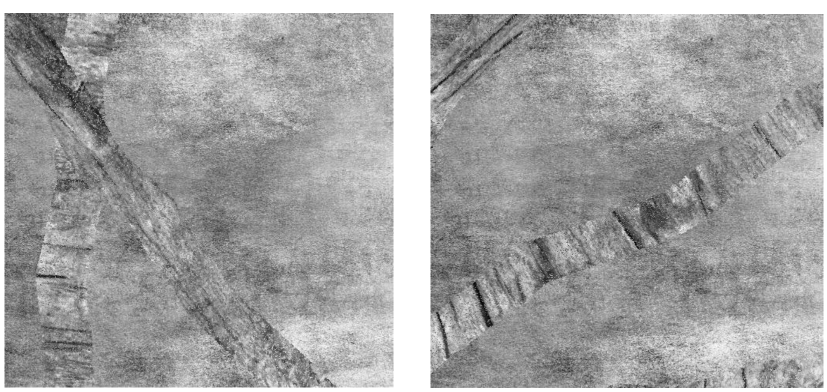


Fig. 5: Sample Images of both classes

2.3 Preprocessing of Data

After generating synthetic fiber images, a total of 33,400 images were created, with 16,700 images belonging to each class. These images are evenly distributed between the two classes to maintain a balanced dataset. The purpose of this balanced distribution is to ensure that the neural networks trained on this dataset are not biased towards any specific class.

To evaluate the performance of the models and assess their generalization capabilities, the dataset is divided into three subsets: the training set, the validation set, and the test set. The train-validation-test split is done with the percentages of 55%, 15%, and 30% respectively. The training set is used to train the models, the validation set is employed for hyperparameter tuning and model selection, and the test set is reserved for the final evaluation of the trained models.

To prepare the dataset for training the neural networks, all the images are uniformly down sampled to a resolution of 240 x 240 pixels. Down sampling the images helps to reduce the computational complexity and memory requirements during model training while still preserving the essential features of the fibers.

Furthermore, the images are converted into NumPy arrays, a commonly used data structure for efficient numerical operations in machine learning. This conversion enables seamless integration of the image data into the neural network frameworks and allows for batch processing and parallel computations during training.

By down sampling the images and converting them into NumPy arrays, the dataset is now in a suitable format to be fed into the neural networks for training, validation, and testing purposes.

3. Training the Model

Two popular pre-trained models, EfficientNet-B1 and VGG19, are employed as base models in this project. Transfer learning allows us to utilize the knowledge and features learned by these well-established models on large-scale datasets and apply them to our specific task of fiber image classification.

To create new models, we build upon the pre-trained base models by adding additional layers on top. The purpose of adding these layers is to adapt the base models to our specific classification task and extract relevant features from the fiber images.

In both new models, a Global Average Pooling layer is added after the base model's convolutional layers. This layer helps to reduce the spatial dimensions of the feature maps while retaining important information. Global Average Pooling aggregates the feature maps by taking the average of each channel, resulting in a fixed-length feature vector for each image.

Following the Global Average Pooling layer, a 2-neuron dense output layer with Softmax activation is added. This final layer is responsible for mapping the learned features to the two classes of fiber images: Class 0 and Class 1. The use of a dense layer allows for non-linear transformations and enables the model to learn complex decision boundaries between the classes.

It's worth noting that all dropout layers in both new models are deactivated. Dropout is a regularization technique commonly used to prevent overfitting by randomly dropping out units during training. In this case, the decision to deactivate dropout layers is based on the observation that the models were not prone to overfitting, and they already exhibited satisfactory generalization capabilities without dropout.

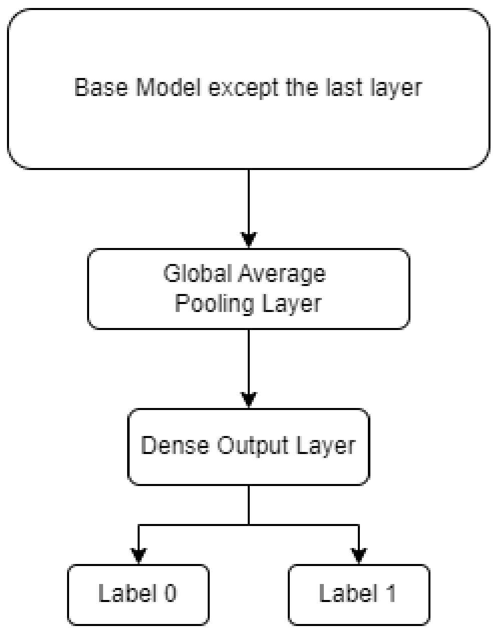


Fig. 6: Layout of the model used.

All base model layers are also deactivated and only the new layers are made trainable.
All hyperparameters are kept the same for both the models. By keeping the hyperparameters consistent, we ensure that both models are trained under the same conditions and can be fairly compared in terms of their performance and generalization capabilities on the fiber image classification task. They are as follows:

Loss function: Sparse Categorical Cross Entropy
Optimizer: Adam
Epochs: 50
Batch Size: 16
Learning Rate: 0.001
Dropout: None

By adding the **Global Average Pooling layer** and the **Dense Output Layer** to the base models, we create new models tailored to the fiber image classification task. These models take advantage of the pre-trained knowledge from weights of ImageNet while adapting to the specific features and patterns of the synthetic fiber images.

The output shape of the last convolutional layer in EfficientNet-B1 is (1, 7, 7, 1280), indicating that it produces 1280 feature maps, each with a spatial shape of 7x7. These feature maps represent the abstract and complex features extracted from the input image. In VGG-19, the last convolutional layer has an output shape of (1, 15, 15, 512), meaning it generates 512 feature maps, each with a spatial shape of 15x15. These feature maps serve as inputs to the subsequent newly added Global Average Pooling Layer.

The Global Average Pooling Layer in both models converts each feature map of size 7x7 for EfficientNet-B1 and 15x15 for VGG-19 into a single value. This pooling operation reduces the spatial dimensions to a single number for each feature map, resulting in an array of shape (1, 1280) for EfficientNet-B1 and (1, 512) for VGG-19.

The output from the Global Average Pooling Layer is then passed to a dense layer with two neurons and a softmax activation function, which generates the final output. Both models have an output shape of (1, 2) for the classification task. This shape indicates that the output consists of probabilities for two classes, representing the model's prediction for the input image.

These two models are trained with 55% of the entire dataset i.e., training set for 50 epochs and evaluated for performance.

4. Evaluating the results

A Validation set is used to keep track of model's generalization and overfitting.
Below are the plots to show the model's training performance of the **EfficientNet-B1** model.

Fig. 7: EfficientNet-B1 Epochs vs Accuracy.

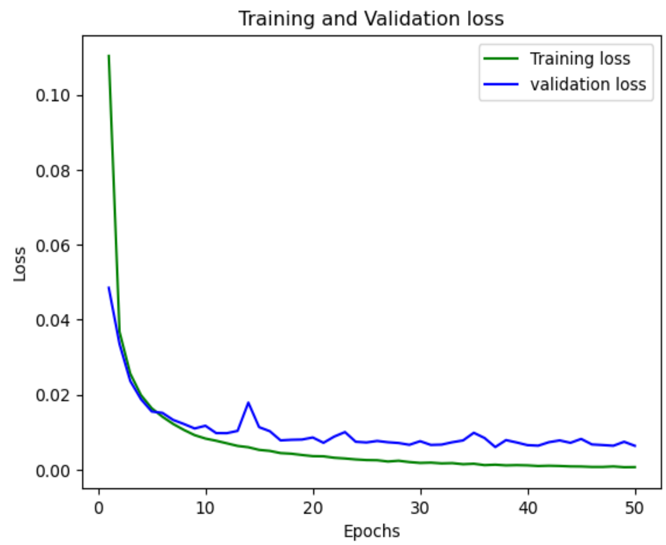


Fig. 8: EfficientNet-B1 Epochs vs Loss.

Over 50 epochs, the training and validation accuracy increased gradually.
Training accuracy reaches 100% whereas validation accuracy hovers around 99.7%.

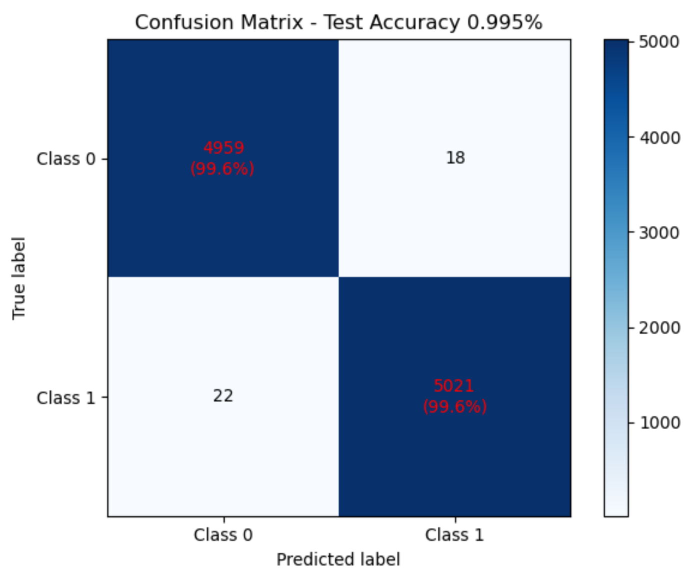


Fig. 9: EfficientNet-B1 Confusion Matrix.

The model performs well on the test data, with an Accuracy of 99.5%.

Below are the plots for **VGG19** model:

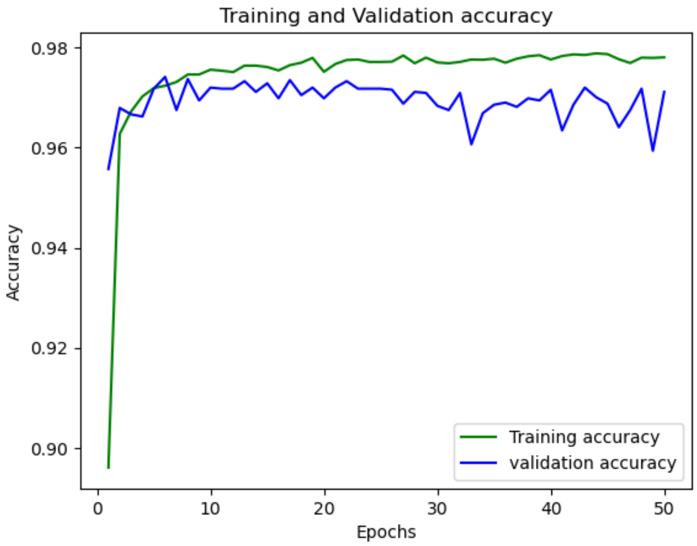


Fig. 10: VGG-19 Epochs vs Accuracy.

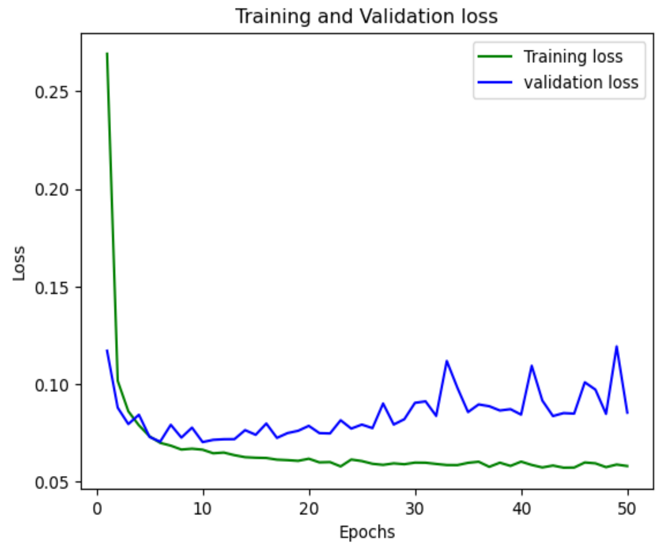


Fig. 11: VGG-19 Epochs vs Loss.

Training Accuracy reaches 98% at some point and validation accuracy hovers around 97%.

Below is the Confusion matrix of the test data.

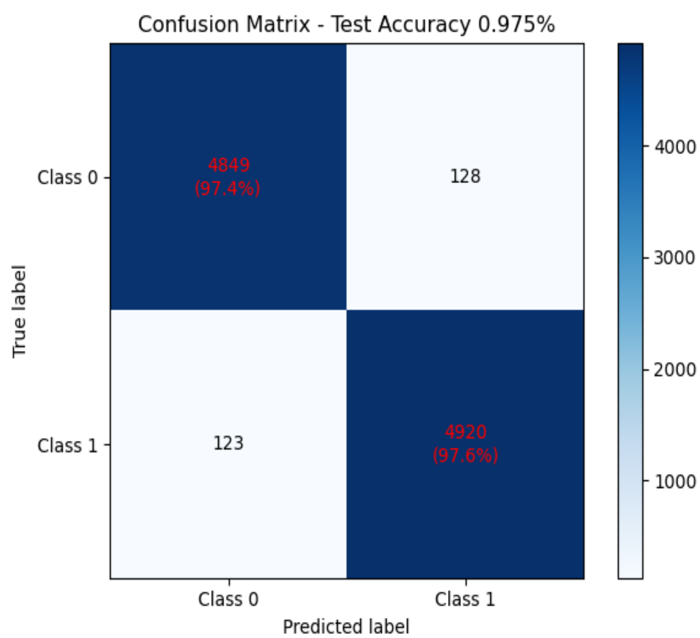


Fig. 12: EfficientNet-B1 Confusion Matrix.

The model performs well on test data with an accuracy of 97.5% which is slightly less compared to the EfficientNet-B1 model.

5. Shuffling the Labels

To assess the validity and reliability of the predictions made by the trained models, a test was conducted by shuffling the labels of the dataset. This means that the ground truth labels of the images were randomly reassigned, breaking the original correlation between the images and their true classes. The shuffled dataset was then split into new train, test, and validation sets using the same percentages as before (55% for training, 15% for testing, and 30% for validation).

By shuffling the labels, we remove any direct relationship between the images and their assigned classes, challenging the models to rely solely on the visual features of the fibers to make accurate predictions.

All the parameters, including the architecture of the models, hyperparameters, and training configuration, were kept the same as in the previous experiment to ensure consistency and fair comparison. The models were retrained using the shuffled dataset, and the train, test, and validation accuracies were measured.

The obtained results showed that the accuracies on the shuffled dataset were around 50%. This outcome provides evidence of the models' ability to identify and utilize meaningful visual patterns in the fiber images to make predictions, even when the original label assignments were disrupted. The fact that the accuracies were not significantly different from random chance indicates that the models were not relying solely on the original label assignments but were genuinely learning the underlying patterns present in the images.

This experiment strengthens the validity of the models' predictions and suggests that they have successfully captured relevant features and visual cues associated with different fiber classes. Below are the plots for the **EfficientNet-B1** model with **shuffled labels**.

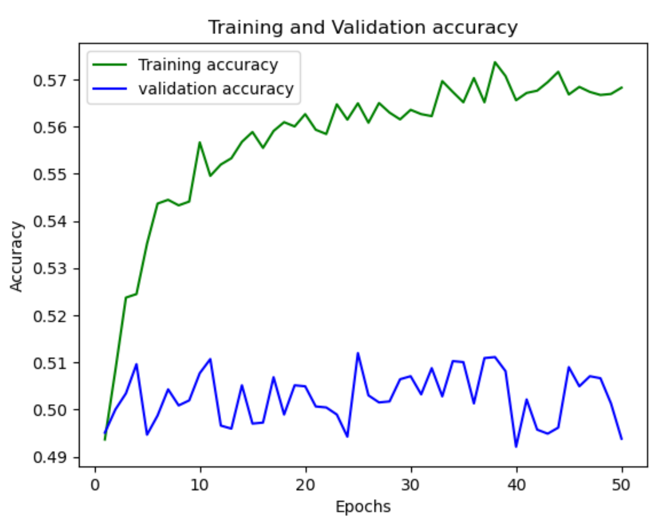


Fig. 13: EfficientNet-B1 Epochs vs Accuracy (shuffled labels).

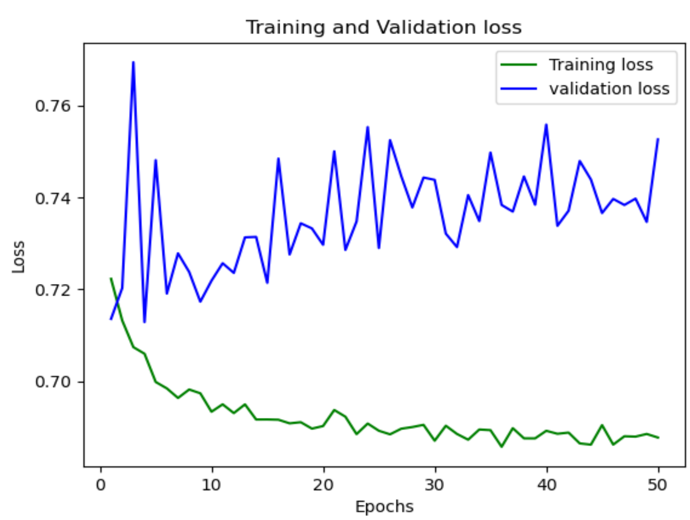


Fig. 14: EfficientNet-B1 Epochs vs Loss (shuffled labels).

Training Accuracy of this model reaches around 57% but the Validation accuracy stays at 50% indicating that the model does not really generalize well on the new data as the labels are shuffled.

Below is the confusion matrix for test data

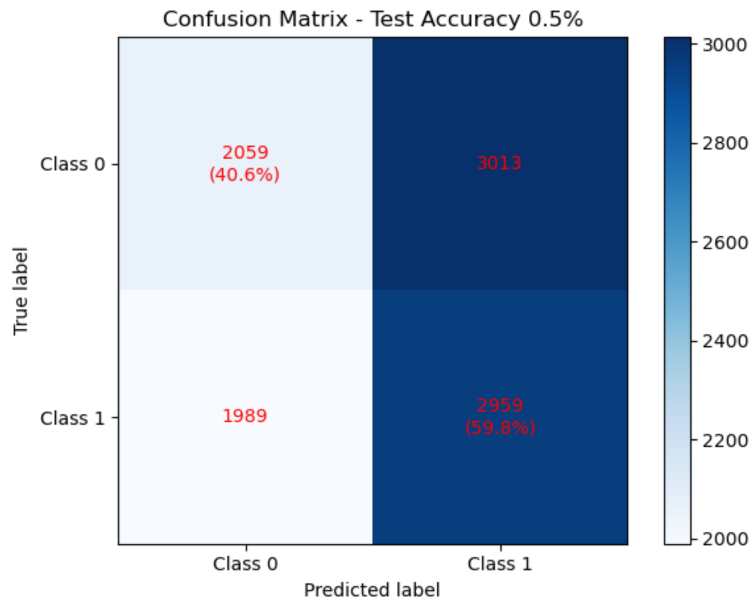


Fig. 15: EfficientNet-B1 Confusion Matrix (shuffled labels).

Testing Accuracy of the model is around 50% which is as expected.

Below are the plots for the **VGG19** model with **shuffled labels**.

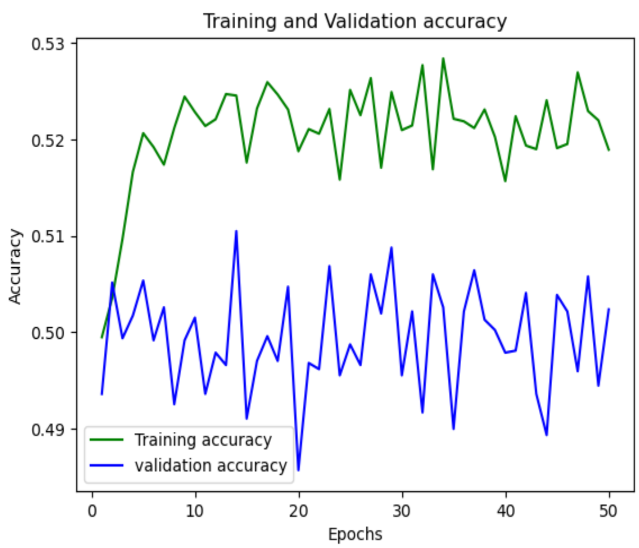


Fig. 16: VGG-19 Epochs vs Accuracy (shuffled labels).

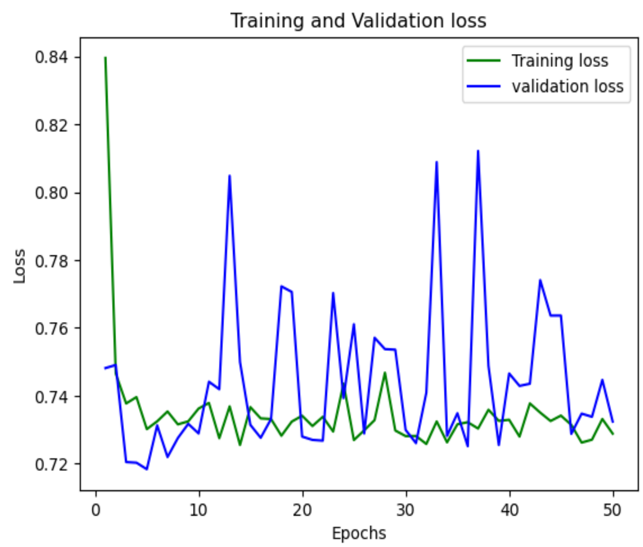


Fig. 17: VGG-19 Epochs vs Loss (shuffled labels).

Even this model has the training and validation accuracy of around 50% which is as expected.

Below is the confusion matrix for test data

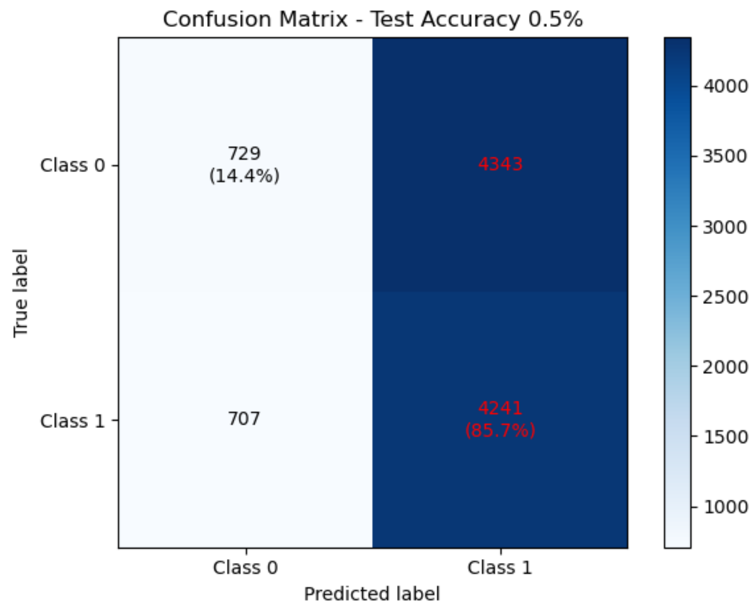


Fig. 18: VGG-19 Confusion Matrix (shuffled labels).

The model performs the same on the test data with an accuracy of 50%.

6. Visualization of Predictions

Grad-CAM (Gradient-weighted Class Activation Mapping) is a technique used in deep learning models to visualize and understand the regions of an image that are important for making predictions. It provides a heatmap overlay that highlights the areas of an image that contribute most to the model's decision.

Grad-CAM utilizes the gradient information flowing into the last convolutional layer of a model to determine the importance of each pixel in the feature maps. By computing the gradients of the predicted class with respect to the feature maps, Grad-CAM identifies the most relevant regions.

The generated heatmap represents the spatial localization of the discriminative features used by the model to make predictions. The intensity of the heatmap indicates the level of importance of each region. The areas with higher intensity are considered more crucial in influencing the model's decision.

Below are the Grad-CAM images for the EfficientNet-B1 and VGG-19 models.

6.1 Grad-CAM’s for Correct Predictions

Class 0 images predicted as Class 0:
Below is an image that belongs to Class 0 and both the models made their predictions as Class 0. It can be seen that the red regions of the heatmap overlay the fiber indicating that both the models focused on the fiber to make its decision. This observation may not be true for all other images in Class 0, but it can be observed for the majority of the images. It can also be observed that the red area in the heatmap of the VGG19 model is usually small and more concentrated than the EfficientNet model.

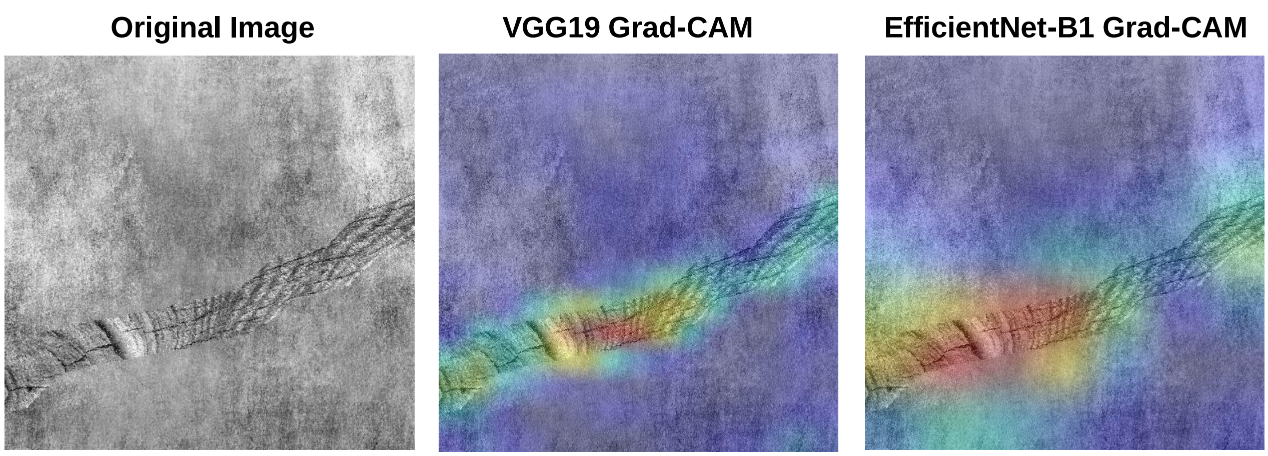


Fig. 19: VGG-19 and EfficientNet-B1 grad-cam’s for class 0 predictions on class 0 images

Class 1 images predicted as Class 1:
Below is an image that belongs to Class 1 and both the models made their predictions as Class 1. It can be seen that the model focuses on the fiber to make its prediction. This observation may not be true for all other images in Class 1 but it can be observed for the majority of the images.

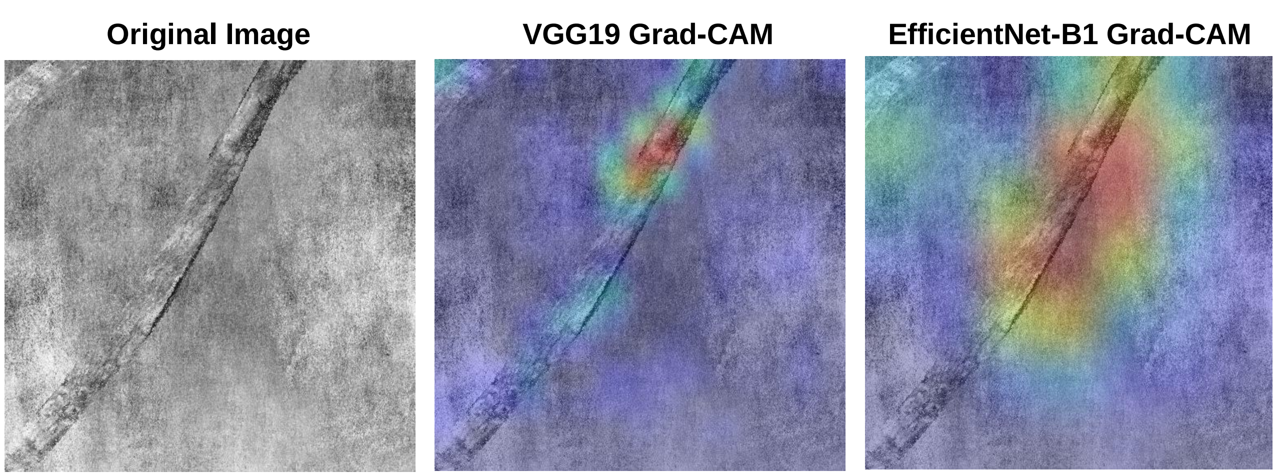


Fig. 20: VGG-19 and EfficientNet-B1 grad-cam’s for class 1 predictions on class 1 images

6.2 Grad-CAM's for Incorrect Predictions

Class 0 images predicted as Class 1:

In the experiments conducted, it was observed that there were cases where the heatmap focused more on the fiber, indicating the model's sensitivity to texture. However, despite this focus, the model sometimes failed to classify correctly, indicating the complexity of the classification task.

Among the 33,400 images in the dataset, only 22 images were classified as class 0 but misclassified as class 1 by the EfficientNet-B1 model. These misclassifications comprised 18 images from the test set and 4 images from the validation set.

Similarly, the VGG19 model misclassified 397 images as class 1 instead of class 0. Among these misclassifications, 128 images were from the test set, while the remaining 269 were from the validation set and training set.

For the specific examples provided below, the EfficientNet-B1 model had a prediction probability of 80%, while the VGG19 model had a prediction probability of 94%.

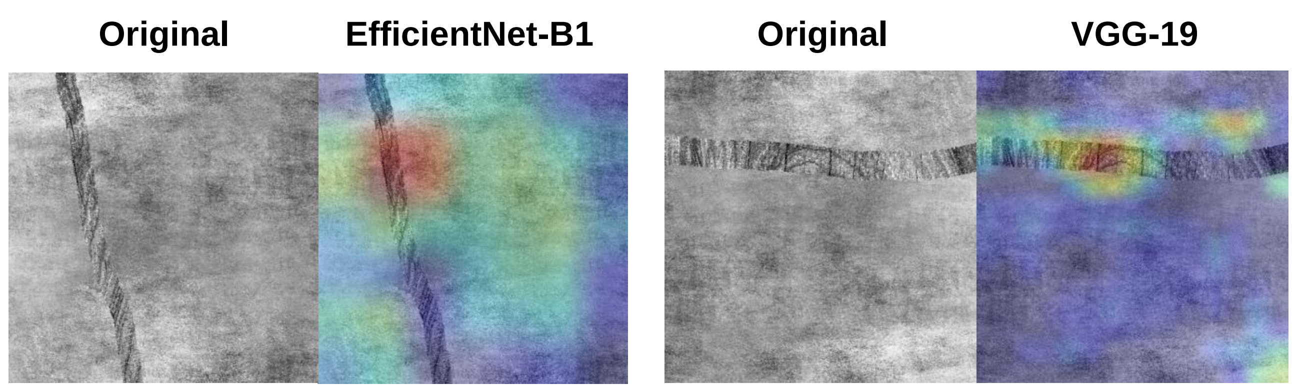


Fig. 21: VGG-19 and EfficientNet-B1 grad-cam's for class 1 predictions on class 0 images

Class 1 images predicted as Class 0:

In certain instances, Class 1 fibers were misclassified as class 0, with the focus primarily on the fiber itself. For the specific examples provided below, the EfficientNet-B1 model had a prediction probability of 87%, while the VGG19 model had a prediction probability of 71%.

Among the 33,400 images in the dataset, there were only 30 images that were class 0 but were misclassified as class 1 by the EfficientNet-B1 model. These misclassifications comprised 22 images from the test set and 8 images from the validation set.
Similarly, the VGG19 model misclassified 395 images as class 1 instead of class 0. Among these misclassifications, 123 images were from the test set, while the remaining 272 were from the validation set and training set.

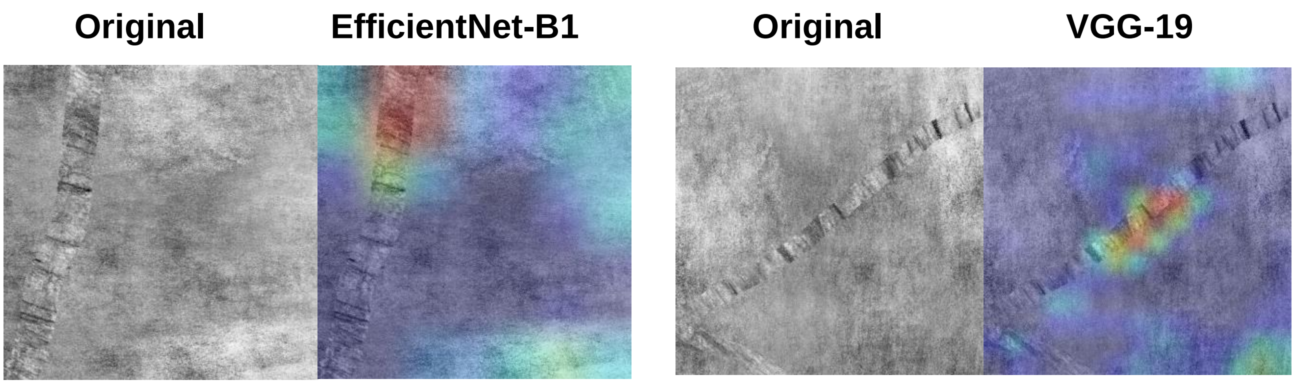


Fig. 21: VGG-19 and EfficientNet-B1 grad-cam's for class 0 predictions on class 1 images

These observations highlight the challenges involved in accurately classifying synthetic fiber images and suggest areas for further investigation and improvement in model performance.

It is noteworthy that several low probability predictions exhibit grad-cams that are not primarily focused on the fiber region. In the case of VGG19 grad-cams, certain portions of the heatmap display emphasis on the background rather than the fibers. This trend is also observable in the EfficientNet-B1 grad-cams. Moreover, images with clear and prominent focus on the fibers tend to have relatively higher prediction probabilities compared to those with focus around the fiber region. While this trend is not universally applicable to all images, it can be observed across a significant number of instances.

6.3 Grad-CAM'S where the focus is outside the fiber.

There are instances where the focus of the model is outside the fiber. This phenomenon is observed for both class 0 fibers being classified as class 1 and class 1 fibers being classified as class 0. This is also observed in cases of correct classifications. The following images exemplify this behavior in the EfficientNet model. On top of each image, given the Actual Label of the image, the prediction the model has made and prediction probability of the decision.

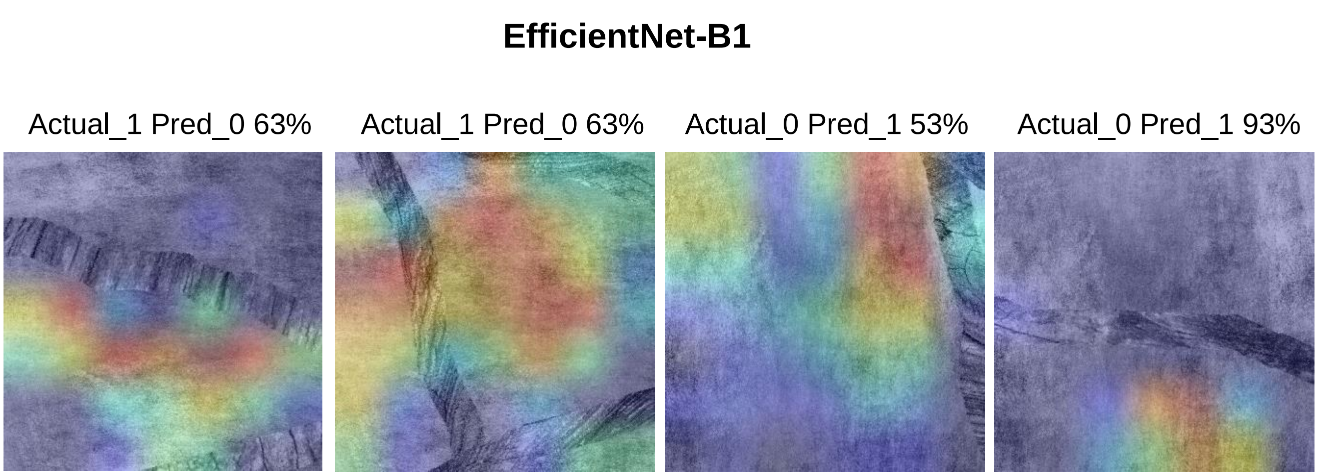


Fig. 22: EfficientNet-B1 grad-cam's with focus outside the fiber.

VGG-19

Actual_1 Pred_0 55% Actual_1 Pred_0 51% Actual_0 Pred_1 55% Actual_0 Pred_1 83%

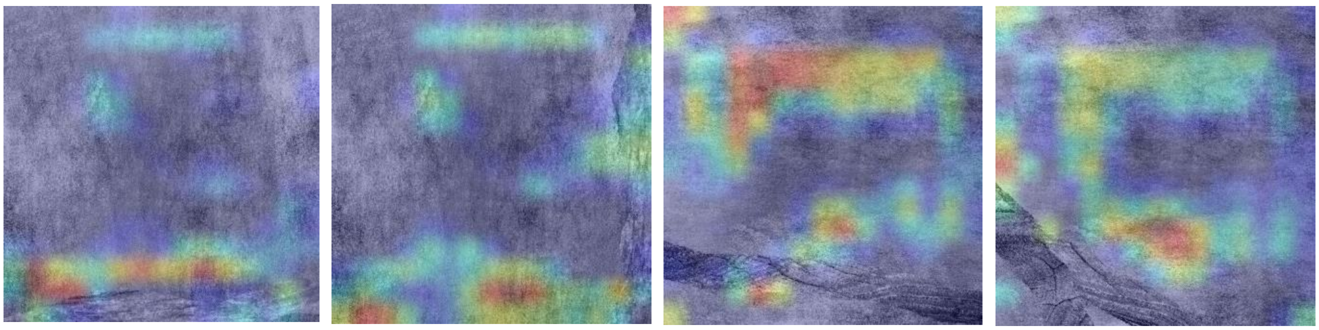


Fig. 23: VGG-19 grad-cam's with focus outside the fiber.

As mentioned above there are also cases where the focus is completely out of fiber, but the prediction is made correctly. Below are a few examples for both the models and both the labels.

VGG-19 **EfficientNet-B1**

Actual_1 Pred_1 82% Actual_0 Pred_0 57% Actual_1 Pred_1 58% Actual_0 Pred_0 75%

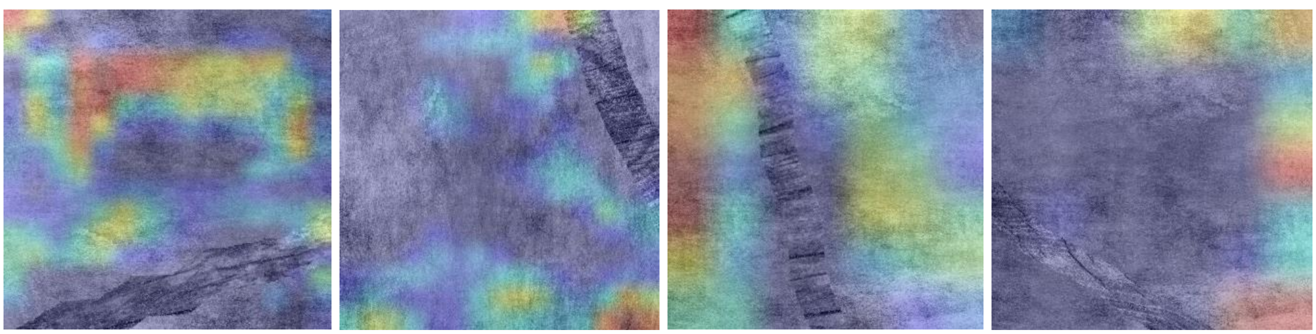


Fig. 24: VGG-19 and EfficientNet-B1 grad-cam's with focus outside the fiber but made correct prediction.

The analysis of predictions on synthetic fiber images using the EfficientNet-B1 and VGG19 models reveals interesting observations. Both models generally focus on the fiber region for Class 0 and Class 1 predictions, but misclassifications occur due to the complexity of the classification task. Some misclassifications occur despite a clear focus on the fiber, while low probability predictions sometimes show grad-cams emphasizing the background rather than the fibers. These findings highlight the challenges of accurate classification and suggest areas for improvement in model performance.

7. Conclusion

In conclusion, this project aimed to create a synthetic dataset of cotton fiber images to establish a baseline for accuracy in classification. The dataset consisted of 33,400 synthetic images, generated to replicate the characteristics of actual fiber images. The images were of a resolution of 500 x 500 pixels and were split into train, validation, and test sets.

Transfer learning was applied using EfficientNet-B1 and VGG19 as base models, with new layers added for classification. The models were trained with consistent hyperparameters, and the resulting accuracies indicated their ability to learn meaningful visual patterns.

By shuffling the labels of the dataset, a test was conducted to validate the models' predictions. The achieved accuracies of around 50% demonstrated that the models were not relying solely on the original label assignments, but rather learning from the inherent features of the fiber images.

This project successfully established a baseline for accuracy in synthetic fiber image classification and provided insights into the effectiveness of transfer learning and synthetic dataset creation. It demonstrated the models' ability to generalize and make accurate predictions based on the learned visual patterns.