PATTERN SENSE: CLASSIFYING FABRIC PATTERN USING DEEP LEARNING

TEAM ID: LTVIT2025TMID42053

TEAM LEADER: PATHAN SHAZIYA KHANAM

TEAM MEMBER: RIZWANA DEVARAGUTTA

TEAM MEMBER: B SUDHA

TEAM MEMBER: I SHARON

COLLEGE NAME: KANDULA LAKSHUMMA MEMORIAL COLLEGE OF ENGINEERING FOR WOMENS

OBJECTIVE:

The objective of the project "Pattern Sense: Classifying Fabric Patterns Using Deep Learning" is to develop an intelligent system that can automatically recognize and classify various types of fabric patterns using advanced deep learning techniques. By leveraging convolutional neural networks (CNNs) and image processing algorithms, the system aims to accurately identify fabric patterns such as floral, striped, checked, abstract, and plain, enabling applications in fashion, textile manufacturing, and retail industries. This project contributes to automating pattern recognition tasks, improving efficiency, reducing manual effort, and supporting quality control in fabric production.

TOOLS AND TECHNOLOGIES USED:

Programming Language:

Python

Deep Learning Frameworks:

TensorFlow / Keras

PyTorch

Libraries for Image Processing & Data Handling:

OpenCV (for image preprocessing)

NumPy (for numerical computations)

Pandas (for data manipulation)

Matplotlib / Seaborn (for data visualization)

Pretrained Models (for Transfer Learning):

VGG16 / ResNet50 / MobileNetV2

Dataset Platforms:

Kaggle

Custom Dataset (scanned or photographed fabric images)

Development Environment:

Jupyter Notebook

Google Colab (for free GPU support)

VS Code (for offline coding)

7.Hardware Requirements:

GPU-enabled machine (NVIDIA CUDA, if using locally)

Minimum 8GB RAM

DATASET DESCRIPTION:

The dataset used for the project consists of high-resolution images of various fabric patterns, categorized into multiple classes. These patterns represent commonly found textile designs used in clothing, upholstery, and fashion industries.

1. Classes of Fabric Patterns:

Floral – Patterns with flower-like designs

Striped – Patterns with horizontal or vertical stripes

Checked – Square or grid-like patterns

Abstract – Irregular or artistic designs

Plain – Solid color fabrics with no design

1. Source of Dataset:

Public repositories like Kaggle, Google Dataset Search, or self-collected images

Images taken from textile websites, online catalogs, or mobile photography (for real-world data)

1. Format and Structure:

File format: .jpg / `.png

Materials and Methods:

Convolutional Neural Networks

In recent years, convolutional neural networks (CNNs), which are capable of recognizing patterns in images, have achieved remarkable performance in the fields of object recognition[25,26] ,tracking[27], and especially, image processing [28]. CNN architectures automatically learn the high-level descriptive features instead of relying on handcrafted features, as is done in traditional machine learning algorithms. Atypical CNN consists of several building blocks, namely convolutional, pooling, and fully connected layers. The shallow CNN networks such as AlexNet [29] and VGGNet [30] are formed by stacking several blocks together. Alternatively, deeper CNN architectures are more complex as they use complex alternating connections among layers, such as ResNet.

VGGNet

TheVGGNetarchitecturecontains144millionparameterswithastackofsmall-sizedconvolutional kernels [30]. It is comprised of 16convolutionallayers withsmall-sizedkernels (3 3), fivemax-pooling layers, three fully connected layers, and an output classifier layer with Softmax nonlinear activation. Since the architecture contains a large number of parameters as compared with AlexNet, it is more expensive computationally because it requires an extensive amount of memory.

RESNET

ResNet He et al., in 2016 [24], were the first to introduce the concept of the residual network (ResNet). The main advantage of ResNet is that it solves the problem of vanishing gradient and degrading accuracy by introducing a concept of shortcut connections making it flexible, task-dependent, and capable of training extremely deep neural networks. These shortcut connections are allowed to skip one or more subsequent layers. The pretrained ResNet-50 model is shown in Figure.

Top of Form

FIGURE(a) ResNet-50 pretrained network with residual connections

Bottom of Form

Problem Statement: Classifying Fabric Patterns Using Deep Learning

In the textile and fashion industries, accurately identifying and categorizing fabric patterns plays a crucial role in inventory management, quality control, and customer satisfaction. Manual classification of fabric patterns is time-consuming, error-prone, and inconsistent due to human subjectivity and visual fatigue. With the increasing diversity of fabric designs — including floral, geometric, striped, plaid, and abstract patterns — a scalable, automated solution is essential.

This project aims to develop a Deep Learning-based fabric pattern classification model that can automatically recognize and categorize fabric patterns from images. By leveraging Convolutional Neural Networks (CNNs), the model will learn to extract visual features and accurately classify patterns into predefined categories.

The proposed solution will benefit manufacturers, designers, and e-commerce platforms by providing faster, more reliable fabric classification, reducing manual workload, and improving search and recommendation systems.

PROPOSED MODEL:

We proposed a pipeline-based approach for our deep learning model. The pipeline consisted of several stages; the first stage received the fabric images, which ended with the classification of the model. The output of each stage in the pipeline acted as the input to the next stage. The proposed pipeline approach is shown in Figure 2 and the details are described as follows

Figure(b) Proposed pipeline approach

- Image Acquisition and Preprocessing The woven fabric images were collected to form a dataset. The dataset required suitable conversions, resizing, and preprocessing of the images. The number of images was smaller in size, so weused various augmentation techniques to increase the dataset, which helped the model have a good generalization and accomplish better recognition.
- Model Generation and Training A learning algorithm that receives input data "X" (map into attributes to the target) and predicts the output "Y" is called a model. For our model, we employed residual network (ResNet-50) architecture. During training, the algorithm performed optimization on the parameters (update weights and biases) which was used for the recognition of the model.
- Model Evaluation The performance of our model was evaluated using various evaluation metrics such as accuracy, balanced accuracy, precision, recall, and F1-score.

DATASET:

The image acquisition of the woven fabric texture images was done using a digital camera surrounded by a light source to control the lighting illumination conditions, as shown in Figure (b). The Nikon D5600 digital camera attached with a micro Nikorr lens of focal length 45 mm was used. The ISO speed value was set to 100 with a f-stop of f/2.8. The woven fabric samples were collected from various warehouses and textile factories. We captured 3540 images from different locations of 880 pieces of fabric. Out of these 3540 images, we kept 2832 images for our testing dataset, while the remaining 708 images were applied through various techniques of data augmentation to generate a total of 11,328 training samples. A few sample images in each class are shown in Figure. These images were subdivided into three classes, namely plain, satin, and twill weave fabric.

Figure(c) Woven fabric image acquisition system

DATA AUGMENTATION:

The problem of insufficient size of the training dataset has been solved using techniques of data augmentation [31]. Data augmentation performs several manipulations such as scaling, skewing, f lipping, and lighting on the entire dataset to form a set of different images as a result expanding the dataset. For larger datasets, deep learning models perform very well. By using augmentation, the total number of images in the dataset increased, allowing the model to train effectively. It is known that data augmentation is a kind of regularization implemented on the overall dataset, consequently, it reduces overfitting problem and the generalization capability is increased by expanding the dataset, which is the major issue, without performing any alterations that affect the structure of the model. Woven fabric image datasets are not easily available and they are difficult to collect.

In this study, we applied several augmentation techniques on the images such as horizontal and vertical flips, shifting, rotation (images are rotated at fixed angles of 30 starting from 0 , 30 , 60 , 90 , and so on), zooming, shearing, and brightness manipulation. An illustration of these augmented images is shown in Figure(e). It was important to rotate the images in order to identify the warp and weft yarns that were oriented in different directions due to the variations that occurred during the image acquisition. Zooming clearly identified the interlacing pattern of woven fabrics. Shearing created the local deformation in the images. All these augmentation techniques related to the situations occurring in the real scenario.

Figure(d) Few samples of woven fabric images from the dataset.

Figure(e) Various augmentation techniques applied to an original image.

EXPERIMENT RESULTS:

The ResNet-50 pretrained CNN architecture is used for the classification of woven fabric images divided into 3 classes. The number of training and testing images in the woven fabric dataset are 11,328 and 2832, respectively. In this work, 80% of the training images are used to train the model and the remaining 20% is assigned to form a validation subset for validating the model. The performance of the proposed deep learning model was evaluated on the test set. The structural properties of the woven fabrics used for the experiments are as follows: Note that the values are mentioned in ranges, yarn linear density (Ne: 6–40), yarn count per cm (25–58 ends/cm), and fabric areal density (125–485 gsm).

EXPERIMENTAL FRAMEWORK:

The woven fabric images were reshaped to 224224 dimensions and the images were also preprocessed by subtracting the mean red-green-blue (RGB) value from each pixel so as to feed it into the model. In this work, we employed transfer learning method to ResNet-50 pretrained CNN architecture, which used the weights of the network learned from ImageNet. The pretrained weights were used to avoid the poor initialization of the model as compared with its counterpart "random initialization of weights". We removed the fully connected layer, which was the last layer that classified the images into ImageNet classes, and the early convolutional layers of the pretrained model acted as a base network for the new customized architecture. Afterward, a global average pooling layer followed by two pairs of batch normalization, fully connected, and dropout layers were, respectively, stacked to the base network. The two fully connected layers encompassed 512 and 256 neurons, respectively. Each fully connected layer was followed by a ReLU activation layer. The batch normalization layers helped to improve the training time of the pretrained model. The inclusion of global average pooling and dropout layers inherently reduced the problem of overfitting. By adding dropout layers, it randomly deleted the redundant neurons; hence, the performance of the model was enhanced. In deep architectures, the problem of overfitting usually fails to have a good generalization on the data which has never been seen before (test data). Finally, the last layer of the proposed model used the Softmax activation function to classify the woven fabric images into three classes. An overall outline of the customized deep learning model is shown in Figure(F).

Figure(f) Overall framework of the proposed model.

The pretrained ResNet-50 model was trained on the woven fabric dataset generated in this work.In this architecture, only the customized newly added layers attached to the base network were trained, keeping the initial convolution all layers as frozen. The main idea of freezing these layers was to improve the convergence rate,as well as avoid the gradient explosion during the training process. After texture features were extracted, then, classification was carried out to compare the predicted class with the actual class. During the training process, the computation cost of the network as decreased since the total trainable parameters of the customized CNN model were also reduced. The proposed model used a method for stochastic optimization, namely Adam optimizer for the optimization of the parameters. The learning rate was set to 0.0001. The drop out ratios for both the drop out layers were chosen as 0.50.The batch size was set to 32.

RESULTS:

The proposed model was trained and tested using NVIDIA GeForce GTX 1060 MQ using 6 GB graphical processing unit (GPU) with Intel i7-8750H @ 2.2 GHz and 16 GB RAM. We used Python 3.6 to implement the model, using Keras library as frontend and Tensorflow as backend.

Evaluation Metrics

The most commonly used evaluation metric for classification is accuracy. It is the ratio between the number of correct predictions to the total number of predictions. Usually, when the data set is imbalanced, ween counter high accuracy showing that it is inclined towards the class, having more samples of images. In an exceptional situation, each test case can be assigned to the large class by the classifier; as a result, accuracy is achieved equal to the fraction of the more frequent labels in the test set. Thus, accuracy can be a confusing evaluation metric. In such cases, the appropriate performance evaluation metric used is the balanced accuracy, as shown in Equation(1). The class count is represented by "l". We achieved an accuracy of 99.3% and a balanced accuracy of 99.1%.

Balanced Accuracy=

The confusion matrix is obtained to show the predictions made by the proposed model on the test dataset and to understand the number of images incorrectly classified. The true class and predicted class are represented by the rows and columns, respectively. Confusion matrix results are shown in Figure(g).

Figure(i) Confusion matrix for plain, satin, and twill weave fabrics. (a)Confusion matrix, without normalization;(b)Normalized confusion matrix.

Conclusion

In this paper, we proposed a customized deep learning model for the recognition and classification of woven fabrics. The proposed deep learning model is based on residual network (ResNet-50) architecture. First, the image acquisition and preprocessing of fabric images are completed and the data augmentation techniques are applied to increase the size of dataset. Second, a pretrained CNN model is used where only the newly attached layers are trained keeping the other layers frozen. The high-level texture features are extracted, and then finally classified based on the types of woven fabric (plain, twill, and satin). We evaluated the performance of our model using various performance metrics such as accuracy, balanced accuracy, precision, recall, and F1-score. A comparative analysis was carried out against other baseline approaches and we also compared the results of the VGGNet pretrained model. The experimental results showed that the proposed method performed better than the other existing studies. The model is robust when variations such as fabric colour, yarn thickness, rotational orientation, and uneven light are considered. The proposed model uses fewer parameters while training making it computationally cost-effective, and thus possess potential for the textile and fashion industry. In the future, we intend to extend our work to other woven and nonwoven fabric types.