Deep Learnt Features and Machine Learning Classifier for Texture classification

S.R.Mathu sudhanan¹, K.Priya², P.Uma Maheswari³

¹ Dept. of CS, Mannar Thirumalai Naicker College, Madurai, India.
²,³ ECE, Thiagarajar College of Engineering, Madurai, India.

E-mail: priya5586@gmail.com

Abstract. Texture classification plays a vital role in the emerging research field of image classification. This paper approaches the texture classification problem using significant features extracted from pre-trained Convolutional Neural Network (CNN) like Alexnet, VGG16, Resnet18, Googlenet, MobilenetV2, and Darknet19. These features are classified by machine learning classifiers such as Support Vector Machine (SVM), Ensemble, K Nearest Neighbour (KNN), Naïve Bayes (NB), Decision Tree (DT), and Discriminant Analysis (DA). The performance of the work is evaluated with the texture databases namely KTH-TIPS, FMD, UMD-HR, and DTD. Among these CNN features derived from VGG16 classify by SVM provides better classification accuracy rather than using VGG16 with a softmax classifier.

Keywords: Texture Classification, Pretrained Model, CNN Features, SVM, VGG16.

1. Introduction

The texture is defined as the intensity variation at regular intervals across the image and the degree of texture can be measured by texture characteristics. It is characterized as hard, soft, smooth, and rough that represent the level of texture variation. The texture is detectible by both perceptible and visual. The touchable feel of the surface can be identified by perceptible texture and form or stuffing of the image can be detected by visual texture. Human beings are identified texture in both perceptible and visual easily, whereas machine-based image classification faces complexity to identify texture. It is used in the field of image classification, remote sensing, image segmentation, medical imaging, writer identification and agriculture [1], etc. In the texture classification problem, discrimination of the significant features to classify images and comparison of system measurement are tedious processes. To provide a proper solution, machine or deep learning techniques are introduced in the field of image classification.

The machine learning-based texture classification encompasses preprocessing feature extraction, and classification. In the pre-processing stage, smoothing, image enhancement, noise removal, and filtering techniques can be performed to enhance the detail in an image [2]. In the feature extraction process, some statistical calculations are used such as Gray level Co-Occurrence Histograms [3], Markov Random Field [4], spatial filters [5], etc to take out substantial information from an enhanced image. Then the extracted features are classified by machine learning classifiers like SVM [6], KNN [7], etc., for proper image classification. Sometimes it leads to incorrect classification depends on a wide range of the object in a single texture class and also data is augmented. Whereas deep learning-based texture
classification used pre-trained networks such as Alex net, VGG net, Resnet, etc., for prompt and accurate classification.

In a pre-trained network model, it generates feature maps with all details, and these feature maps are used to classify within the same model. The convolutional Neural Network (CNN) feature consists of low detail features in starting layers and high detail features in forthcoming layers. Fok et al [8] proposed a pre-trained CNN for texture classification that extracts features by stunting inhibitory neurons and classifies features using a sigmoid function. Krishnan et al [9] used a combination of pre-trained networks such as VGG, Resnet50, Inception, and Xception. Collaborative these networks with the different numbers of layers and parameters to get improved accuracy of the texture image classification. Recent techniques that evolved in the deep learning domain for texture classification is by taking deep learned features and classified using various machine learning classifier. Philomina et al [10] used CNN features that are extracted from various pre-trained networks (Alex Net, DenseNet201, ResNet50, Resnet101, InceptionV3) and classified by SVM. In this proposed methodology, CNN features extracted from pre-trained deep learning architectures and classified by machine learning classifiers for texture classification to improve classification accuracy and performance of the proposed work has been evaluated with the existing state of arts. Phase 2 describes texture database collections that are used in this proposed work and in phase 3, the proposed methodology has been explained. Results and conclusion are explained in phases 4 and 5 respectively.

2. Texture Database Collections

Several texture databases were released to classify images. The texture databases used in this paper are KTH TIPS (Textures under varying Illumination, Pose and Scale), Flickr Material Database (FMD), UMD HR (University of Maryland, High Resolution), Describable Texture Dataset (DTD) available from the internet. KTH TIPS Database contains different materials such as Aluminium_foil, Brown_bread, Corduroy, Cotton, Cracker, Linen, Orange_peel, Sandpaper, Sponge, Styrofoam. The resolution of the images in the KTH TIPS database is 200 X 200. FMD database was collected manually from the flickr.com website and it contains 10 classes such as Fabric, Foliage, Glass, Leather, Metal, Paper, Plastic, Stone, Water, and Wood with the image resolution 512X384. These images are classified under the real-world appearance of common material, each class having 100 images, a total of 1000 images were existing in the FMD database. The UMD HR texture database consists of 12 classes, each contains 40 images with a high resolution of 1280x960 and it consists of various classes like fruits, floors, and several types of plants. DTD is a challenging database consists 5640 images with 47 classes each class consists of 120 images. Some of the sample images from these databases are shown in Figures 1, 2, 3, and 4.

![Image 1: A Sample for each of the 10 classes from the KTH-TIPS Database](image-url)
Figure 2. A Sample for each of the 10 classes from the FMD Database

Figure 3. A Sample for each of the 12 classes from the UMD HR Database
3. Proposed Methodology
Deep Neural Networks are the recent developing trend and it gains a lot of consideration due to notable triumphs in the image processing area. Deep networks are designed for specific feature space distribution and each layer has a specific CNN feature map. The CNN features extracted from the pre-trained network have high-level features in the last stage layers which are used to discriminate textures. This proposed system uses a VGG16 pre-trained network model for CNN feature extraction and that classified with an SVM classifier as shown in figure 5.

3.1. Algorithm for Texture Classification
In the training stage, the CNN features are extracted from the VGG16 network and classified by SVM that is tested with the texture database. The algorithm of texture classification following the steps.
1. Download the image databases namely KTH-TIPS, FMD, UMD-HR, and DTD. Labeling these images according to corresponding categories.
2. The input layer of VGG16 expects the image size [224 224 3]. So that resize the images in the size of [224 224 3].
3. Splitting the image database into training and testing sets. 80% of the images were used for training and the remaining were used for testing.
4. Extract CNN features from the 15th layer for the training images. This layer is fully connected and having 4096 CNN features.
5. The CNN features classified by SVM using a 10-fold validation method.
6. Evaluate the network with existing pre-trained network CNN features with machine learning classifiers.

3.2. CNN Features Extraction from Pre-trained Model VGG16
VGG16 architecture contains 16 layers, 16 denotes the weight of the model. The input image size of the VGG16 architecture is 224*224*3, with zero center normalization. The first layer consists of two convolutional layers with a rectified linear unit (relu) and max-pooling layer size [2 2]. The convolutional layer1 has 64 filters and layer 2 consists of 128 filters. Layer 3 and 4 consists of three
convolutional layers with 256 and 512 filters respectively. Layer 5 consists of 3 convolutional layers with a 512 filter size. The filter kernel size [3 3] is used throughout the architecture that is [3 3]. Three fully connected layers are followed by a stack of convolutional layers. The 14th and 15th layers have 4096 channels, the final layer (16th layer ) is the softmax layer which classifies the texture image in the VGG16 model. But in this proposed work, the CNN features are extracted from a fully connected layer (15th layer) and fed to the SVM classifier instead of the softmax classifier. The layer description of the VGG16 network model is explained in Table 1. The basic layers of CNN can be described in the following sections.

3.2.1 Convolutional Neural Network
Local connectivity and weight sharing are important parameters of CNN. The input images are passed through a convolutional network that has a series of convolutional, pooling, and fully connected layers as shown in figure 6. Then the output of the CNN features is considered a significant feature for texture classification.

3.2.2 Convolutional Neural Network
In the convolutional layer, the kernel matrix is multiplied with the input image to produce a feature map to the next layer. The steps of the convolution process are followed by

- Select the kernel size
- Apply convolution to the input image by sliding the kernel matrix.
- Element-wise matrix multiplication is performed at every location and some of the results are taken onto the feature map. The convolution process can be performed using the formula in equation 1.

\[
S(i, j) = \sum_{m} \sum_{n} I(m, n)K(i - m, j - n)
\]  

(1)

Where \(S(i, j)\) is a convoluted image \(I(m,n)\) is the input image and \(K\) is the kernel matrix.

3.2.3 Rectifier Linear Unit (RLU)
After the convolution process, the activation function is performed over the input image. Relu is a piecewise non-linear function, that retains the positive intensity values and discards the others to hold significant detail of the image.

3.2.4 Pooling Layer
Extracting and identifying the feature map in an input image is critical. This creates a problem in the output feature map. Taking a downsample on the feature map is solving the sensitivity problem and that reduced dimensionality on the feature map. Pooling is a downsampling process that solves sensitivity problem and it reduces the feature map dimensionality. There are two pooling techniques available, one is average pooling and another one is max pooling. Average pooling is the average presence of features and max-pooling is the most activated presence of features.

3.2.5 Fully Connected Layer
The last pooling layer output is the input of the fully connected layer. This layer adds some weights and converts them to a single vector for predicting label class. Then it gives the probability of texture class.
3.3. Support Vector Machine

There are a lot of machine learning classifiers available to classify images. One of the best-supervised classifiers is SVM. In the machine learning technique, data should be labeled manually and it is trained by the classifier. The SVM tries to draw a line between data to represent data. If the data points come closer to the line, then it will try to draw a hyperplane to support those data points to classify. Let the feature points \( x_j \) and labeling of feature points \( y_j \) as denoted in equations 2 and 3. Then that feature points are classified by hyperplane equation in 4.

\[
x_j \in R^d
\]

\[
y_j = \pm 1
\]

\[
f(x) = x^T \beta + b = 0
\]

Where \( \beta \in R^d \) and \( b \) is a real number called bias.

### Table 1. Layer Description of VGG16

| Layer         | Feature Map Size | Kernel Size | Stride | Activation |
|---------------|------------------|-------------|--------|------------|
| Input         | Image            | 224x224x3   | -      | -          |
| 1             | 2xConvolution    | 224x224x64  | 3x3    | 1          | Relu       |
|               | Max Pooling      | 112x112x64  | 3x3    | 2          | Relu       |
| 3             | 2xConvolution    | 112x112x128 | 3x3    | 1          | Relu       |
|               | Max Pooling      | 56x56x128   | 3x3    | 2          | Relu       |
| 5             | 2xConvolution    | 56x56x256   | 3x3    | 1          | Relu       |
|               | Max Pooling      | 28x28x256   | 3x3    | 2          | Relu       |
| 7             | 3xConvolution    | 28x28x512   | 3x3    | 1          | Relu       |
|               | Max Pooling      | 14x14x512   | 3x3    | 2          | Relu       |
| 10            | 3xConvolution    | 14x14x512   | 3x3    | 1          | Relu       |
|               | Max Pooling      | 7x7x512     | 3x3    | 2          | Relu       |
| 13            | Fully Connected  | -            |        |            |            |
| 14            | Fully Connected  | -            |        |            |            |
| 15            | Fully Connected  | -            |        |            |            |
| 16            | No of Classes   | -            |        |            | Softmax    |
\[ y_i f(x_i) \geq 1 \]  

(5)

The SVM tries to minimize the space between hyperplanes as shown in figure 7 and gives the best match for feature data points by using equation 5.

4. Results and Discussions

In this section, the experimental results of the texture classification using CNN features extracted from pre-trained network VGG 16 and classified by SVM are described. The databases used in this work are KTH-TIPS, FMD, and UMD HR. All experimental results obtained using Matlab 2020a software installed in Intel i7 ninth-generation processor, Nvidia GeForce GTX 1650 graphics card laptop. In the training phase, 80% of the images in each class were used to train the classifier, and the remaining were used for testing. First, the texture classification of all databases done by pre-trained deep learning networks such as Darknet19, MobilenetV2, Googlenet, Resnet18, Alexnet, and VGG16. Among these networks, VGG16 provides better classification accuracy of 88.13%, 83.74 %, 73.45%, and 53.09% for the databases KTH-TIPS, FMD, UMD-HR, and DTD respectively, while using softmax classifier as depicted in figure 8.

To further improve the texture classification accuracy of the database, the feature map taken from VGG16 is classified by SVM. Table 2 shows the experimental results of the KTH-TIPS database and obtained a cross-validation accuracy of 98.62. In comparison, using the VGG16 network with a softmax classifier, the proposed methodology increases accuracy by around 10%.
Table 2. Results of Classification Accuracy for KTH-TIPS Database

| Training Images | Testing Images | Pretrained Network | Number of CNN Features | Classifier | 10 Fold Validation Accuracy ( %) |
|-----------------|----------------|--------------------|------------------------|------------|--------------------------------|
| 10 classes, per class 65 images | 10 Classes, per class 16 images | VGG 16 | 4096 | SVM | 98.62 |

The performance evaluation of texture classification for the KTH-TIPS database by SVM is compared with existing machine learning classifiers such as KNN, NB, DA, DT, and ensemble as shown in figure 9.

![Figure 9. Performance Analysis of SVM for KTH-TIPS Database](image)

Table 3 shows the experimental results of the FMD database and attained 76% cross-validation accuracy which is slightly higher than other existing machine learning classifiers as shown in figure 10 and this provides nearly 3% of increased classification accuracy compared with other pre-trained network models.

Table 3. Results of Classification Accuracy for FMD Database

| Training Images | Testing Images | Pretrained Network | Number of CNN Features | Classifier | 10 Fold Validation Accuracy ( %) |
|-----------------|----------------|--------------------|------------------------|------------|--------------------------------|
| 10 classes, per class 80 images | 10 Classes, per class 20 images | VGG 16 | 4096 | SVM | 76 |
Table 4 shows the experimental results of the UMD-HR database with a classification accuracy of 99.57%. Figure 11 shows the comparison of classification accuracy for the UMD-HR database with other machine learning classifiers and it is increased around 10% while compared with other pre-trained.

| Training Images | Testing Images | Pretrained Network | Number of CNN Features | Classifier | 10 Fold Validation Accuracy ( %) |
|-----------------|----------------|--------------------|------------------------|------------|---------------------------------|
| 12 classes, per class 32 images | 12 Classes, per class 8 images | VGG 16 | 4096 | SVM | 99.57 |

Figure 10. Performance Analysis of SVM for FMD Database

Figure 11. Performance Analysis of SVM for UMD-HR Database
The classification accuracy for the DTD database is shown in Table 5. This is compared with another classifier with 60.08% accuracy as shown in Figure 12. Among these classifiers SVM provides better results with 7% increased accuracy when compared with other pre-trained networks.

### Table 5. Results of Classification Accuracy for UMD-HR Database

| Training Images | Testing Images | Pretrained Network | Number of CNN Features | Classifier | 10 Fold Validation Accuracy (%) |
|-----------------|----------------|--------------------|------------------------|------------|-------------------------------|
| 47 classes, per class 96 images | 47 Classes, per class 24 images | VGG 16 | 4096 | SVM | 60.08 |

![Figure 12. Performance Analysis of SVM for DTD Database](image)

### 5. Conclusions

In the evolving field of image classification analysis of various structures, that distinguishes various texture details in an image. Texture classification the classic issue in the emerging field of image classification analysis that distinguishing various texture details in an image. For machine learning algorithms, selecting the best features has often been a daunting task and CNN offers a simpler approach for selecting the best features. In this proposed work, the features are extracted from the pre-trained CNN model like Alexnet, VGG16, Resnet18, Googlenet, MobilenetV2, and Darknet19 are analyzed with different machine learning classifier for the texture database namely KTH-TIPS, FMD, UMD-HR, and DTD. The review of the results shows that CNN features extracted from VGG16 along with SVM provide superior classification accuracy.

### References

[1] Li, G., Lu, D., Moran, E., Sant Anna, 2012, Comparative analysis of classification algorithms and multiple sensor data for land use/land cover classification in the Brazilian Amazon. *J. Appl. Remote Sens.* 6

[2] T. Randen and J. H. Husøy, 1999, Filtering for texture classification: A comparative study, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 4.

[3] J. S. Weszka, C. R. Dyer, and A. Rosenfeld, 1976, A comparative study of texture measures for terrain classification, *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-6, no. 4, pp. 269–285.

[4] G. R. Cross and A. K. Jain, 1983, Markov random field texture models, *IEEE Trans. Pattern
Anal. Mach. Intell., vol. PAMI-5, no. 1, pp. 25–39.

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

[6] Rakesh Mehta and Karen Egiazarian, 2016, Texture Classification Using Dense Micro-Block Difference, IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 25, NO. 4.

[7] K. Priya(B), S. Mohamed Mansoor Roomi, B. Sathyabama, and R. NeelavathyR., 2020, Texture Classification by Local Rajan Transform Based Descriptor. V. Babu et al. (Eds.): NCVPRIIPG.

[8] Tivive, FHC & Bouzerdoum, 2006A, Texture Classification using Convolutional Neural Networks, IEEE Region 10 Conference (TENCON 2006), Hong Kong, China, 14-17, 1-4.

[9] Krishan Gupta, Tushar Jain, and Debarka Sengupta, 2018, Texture Classification Using Deep Convolutional Neural Network with Ensemble Learning, c Springer Nature Switzerland AG. A. Groza and R. Prasath (Eds.): MIKE 2018, LNAI 11308, pp. 341–350.

[10] Philomina Simon, Uma, 2020, Deep Learning based Feature Extraction for Texture Classification, Third International Conference on Computing and Network Communications (CoCoNet’19). 1877-0509 © The Authors. Published by Elsevier B.V.