An Improved Convolutional Neural Network with LSTM Approach for Texture Classification

M.Subba Rao[1], B.Eswara Reddy[2]
[1]Department of Information Technology, Annamacharya Institute of Technology & Sciences, Rajampet, Andhra Pradesh, India. msraoswap@gmail.com
[2]Director, Software Development Centre, JNTUA, Anantapuramu, Andhra Pradesh, India

ABSTRACT

Texture classification is a problem that has several applications, such as remote detection and recognition of forest species. Solutions tend to be customized for the dataset used, but are not generalized. Machine learning algorithms play an important role in the current Texture Classification. However, these algorithms are suffering with low accuracy and classification rate. Deep learning is another sophisticated technique to solve these challenges because texture classification performance is not strong in traditional machine learning systems. The Convolutional Neural Network (CNN) in combination with Long Short-Term Memory (LSTM) forms a robust selection between a powerful invariant feature extractor and an accurate classifier. This model should automatically determine the efficient properties of the feature samples so that the texture samples can be classified accurately. Expert fusion provides stability in classification rates between different data sets and the proposed model will significantly increases texture classification performance. From the experimental analysis, it is ascertained that CNN-LSTM is outperforms with existing state of the art of the algorithms SVM and CNN.

Key words: Texture classification, Machine Learning, Convolutional Neural Network (CNN), Long-Short Term Memory (LSTM), Support Vector Machines (SVM) and Performance Measures.

1. INTRODUCTION

Texture classification is the method by which specific textures are identified from the specified images. While the description of the textures themselves sometimes seems irrelevant in its own way, however, it is possible to implement a large number of real-life problems related to unique textures of various materials [1]. The texture is characterized by a non-uniform spatial distribution of image intensities. There are mainly three different ways of pattern recognition on Texture Classification [40,41] Statistical pattern recognition, Structural pattern recognition and Neural pattern recognition Texture-based classification techniques are being used in a variety of real-world applications, such as content-based image recovery [2], face recognition [3], rock classification, and wood species recognition, classification of fabrics and geographic segmentation of the landscape. The purpose of texture classification is to classify a sample image into one of a set of known texture classes. There are two types of texture classification, supervised and unsupervised classification. In the supervised classification method, a classifier trained with the characteristics of the known classes. In the unsupervised classification method, the classifier recognizes different classes based on the similarity of the input characteristic, no previous training of the classifier occurs. Classification methods texture can be classified into three categories based on pixels, based on local characteristics and based on regions [4].

The first two methods of texture classification are the Co-Occurrence matrix [5] and the Markov Random Fields (MRF) [6], which has been widely used to distinguish textures to a relatively small region [7], Gabor and wavelet filters [8]. The extraction and classification of texture features within a unified framework that incorporated multi-channel filtering principles are presented in [9]. Neural network architecture is widely gained popularity in texture classification. The first network they have built is a three-layer control network (including the input layer), that is, a multi-layer sensor (MLP), with each input node fully linked to a specific M X M scale area in the input image. The second network is close to that suggested by [10], and uses a weight distribution method to link secret neurons to the anterior layer. The previous network has an approximate cost of 5,000 pesos and has been equipped with an automated back-propagation algorithm.

A class of machine learning algorithms, called deep learning, has been used more and more in classification and pattern recognition in recent years. In a hierarchical architecture, deep learning applies multiple layers of information processing to generate a deep model [36]. A small range of academic studies have examined deep learning in the area of

texture recognition, but none of them have successfully used the maximum strength of deep learning techniques [24]. Furthermore, although CNN was recognized as highly accurate, it was not exploited in the field of texture classification [29].

In this paper, we propose an LSTM-based convolutional neural network for texture classification. The network is based on the same theoretical concepts as CNN [11], but with a much more simplified network configuration and a formal connection scheme that results in small free network parameters. The CNN-LSTM is evaluated for comparison on the plot images taken from [12]. Section 2 describes related research and its associated work and highlights its new aspects. Section 3 will present a more detailed description of the proposed CNN-LSTM integration along with Max-pooling to efficiently classify the texture images. The Results and performance analysis is presented in section 4 followed by the Conclusion& Future scope in section 5.

2. BACK GROUND

This section explores the basic preliminaries and related research work associated with texture analysis and classification. This section also presents feature extraction methods and machine learning methods to predict the accuracy of the texture classification being explored.

2.1 Texture Analysis and Classification

Texture is a primary aspect in various styles in photographs, and various methods of describing texture i.e. extracting features to explain texture information have been developed over the decades. Work on the representation of textures in the past century has focused largely on two different approaches: filtering-based and statistical-based [24]. The image is convoluted with a bank of filters in filtering-based approaches. Statistical modeling approaches using texture statistics k-order and using probability distributions to characterize images such as the Markov Random Field, fractal models, and the Gray Level Co-Occurrence Matrix (GLCM).

Various approaches were used to explore the problem of texture analysis and improve classification results. [25], they developed a new Complete LBP function extractor (CLBP). The central pixel defines a local area in the gray image stage and a regional threshold is performed before the binary encoding of the central pixel to produce invariable rotational characteristics. Local binary patterns (LBP) for feature extraction, and K nearest neighbor (KNN) for sorting are used in [26]. Using matrices (GLCM) and Gabor filters for joint occurrence are suggested in [27]. Color-based features and GLCM in [28], using a combination of LBP, CLBP, GLCM

and color feature extractors in [29]. The Gabor local color binary matching pattern (CLGBOCP) that finds the spatial relationship of the neighborhoods of a pixel using local matching binary edges. CLGBOCP can integrate LBP with matching matrix features for full feature extraction [30].

Artificial neural network is a theoretical or statistical construction focused on neural biological networks. Artificial neural networks exist in many types. There is a feedback neural network, one of the most commonly used and easiest types of neural network, where knowledge simply advances. Recurrent neural networks that spread information in both directions, forward and backward. Other famous neural networks include Kohonen Self-Organizing Networks, Hopfield Networks, Boltzmann Machines, and Spiked Neural Networks. It can result in multiclass SVM performing worse than MLP. Other classifiers are Bayes classifier [31], the Learning Vector Quantization (LVQ) [32] and the Hidden Markov Model (HMM) [33] used for texture classification.

2.2 Feature Extraction

There are several common extraction methods for texture classification. Most of those techniques that have become popular in recent years are statistical and signal processing techniques. The Gray Level Co-Occurrence Matrix (GLCM) is an extraction of ancient characteristics proposed for the classification of shapes. Traditionally, texture properties are premeditated from GLCM, like Correlation, Strength, Entropy and Homogeneity, on the contrary, [13]. However, GLCM has often been pooled with other methods in recent years, and has rarely been used individually [14, 15]. There are a few other GLCM implementations in addition to the traditional implementation [15] was through the introduction of a second-order statistical method on the shape properties of a one-dimensional GLCM [16] and the use of a raw GLCM instead of a first-order statistic [17]

Wavelet transformation is another signal processing method implemented during the last two decades in image processing and pattern recognition [18]. One of the most common wavelet transitions is Discrete Wavelet Transitions (DWT). Therefore, good characteristics are produced that lead to higher precision, albeit more complicated and slower. Some of the other transitions used are wavelet-based classification of the dynamic form using curve transformation [14], as well as some other lesser-known transitions. This sign allows [18], logarithmic polar transformation [33].

Independent Component Analysis (ICA) has been used to extract characteristics from PCA, but there is a limit to obtaining only second order statistics, while ICA can prevent this difficulty and obtain higher order statistics. The

multivariate signal is used to discern and is also applied in texture classification [19].

There are many other feature extractions that have become controversial including models based on stochastic models, including Fractals [20] and Markov Random Field [6], Sequential Error Approach Curves (SAEC) [21], Simple Image Properties, Spectral Correlation Feature (SCF) [22], Legendre Spectrum and Multilevel Blob Characteristic (MBF) [23]. In recent years, signal processing techniques have become very popular, particularly for Gabor filters and wavelets.

2.3 Deep Learning in Texture Classification

Deep learning has recently gained a lot of interest and popularity in many fields of research. In present research, a well-known deep learning network called, the Convolution Neural Network (CNN) is used to texture classification to exploit the benefits of functional learning and classification. A convolution network is a type of neural network that has been investigated since the 1990s and is primarily used to process images, videos, and any type of spatially organized data. [34]

In general, we can divide a neural network of convolutions into two well-defined components: the feature extraction part and the classification part. In most DCNNs, the last part, like a typical multilayer neural network, is organized into one or more fully bonded layers. On the other hand, the abstraction part of the function consists of convolutional layers that differ significantly from one network to another, because over time many of the convolutional blocks are represented. However, it does contain some convolutional and clustering layers in the early models, [34] and, following the principle of deep learning; modern structures are clustering more and more layers. In addition, several convolutional block structures have been proposed, such as starter modules [35] and residual connections [36].

2.4 CNN Model for Texture Classification

CNN based on [39, 42] are deep learning feature extraction models that have recently been shown to be quite effective in image recognition. As of today, numerous business giants like Google, Twitter, and Amazon are using the templates. And recently, Google researchers applied CNN to video data [39]. It is achieved by performing separate transformations of the filter values as trainable weights in the image. Several filters are added to each path, and they form feature maps along with neural activation functions. A grouping scheme accompanies this, where only the relevant details of the function maps are grouped.
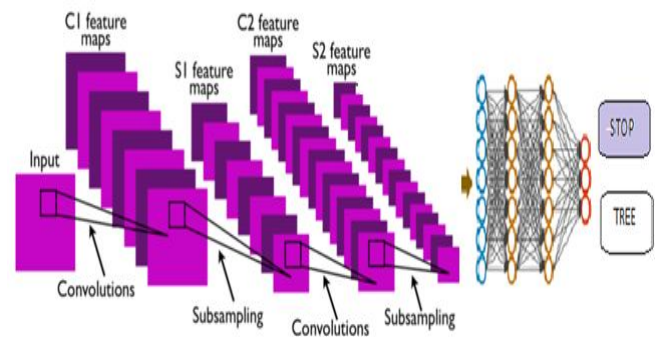

Figure 1: CNN model for Texture Classification

The design of the Convolutional Neural network, as seen in Figure 1, is composed of input, Convolution layer, clustered layer, fully connected layer, and output layer.

The CNN networks have diverse architectures with a specific number of levels of Convolution and grouping. The function map of layer i can be represented as the Convolution method:

$$Mi = f(Mi{-}1 \otimes Wi + bi).$$

Wi is the weight vector of the transfer core of layer i, the process representation describes the transfer phase and is the displacement variable of layer i, and (z) is the excitation function. The Convolution nucleus creates new characteristics in the Convolution cycle by performing the Convolution procedure with the characteristics of the inputs.

3. METHODOLOGY

The CNN - LSTM fusion exploits the invariant features learning of CNN and high accuracy of separation of feature vectors with CNN and SVM.

3.1 CNN-LSTM Model Framework

To get detailed evidence, we built a Neural Convolution network. The entire network is made up of three hidden layers. The Convolutional layer and a grouping layer are converged as the first 2 layers which comprises as hidden layers. The third layer is an output LSTM layer. With each hidden layer, the number of Convolutions is different. The network uses Convolution cores and shared cores to increase efficiency by constantly expanding the network configuration. The number of conversions should improve accuracy. We build 2 Convolution layers one with 64 filters and 128 filters on the other. Figure 2 is a paradigm for texture classification centered on the Convolutional Neural Network algorithm with the LSTM-70 as output. Throughout Figure 2, it can be shown that the structure mainly consists of three phases:

Phase 1: Different Texture data set has a large amount of texture images that allows you to load datasets.
Phases 2: Training and feature extraction: the constructed CNN model is used for the training the texture dataset and

automatically extracts features based on characteristics of the image.

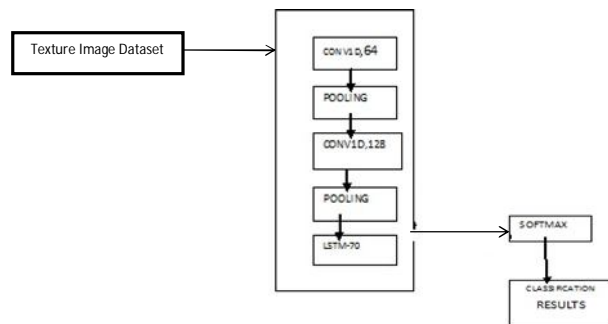**Phases 3**: Test: The SoftMax classifier is used to identify and collect the effects of the test.

Figure 2: Proposed CNN-LSTM Model for Texture Classification

3.2 LSTM (Long Short-Term Memory)

Long Short-Term Memory (LSTM) is a model of artificial recurrent neural network (RNN) that is used in deep learning. LSTM has input links, unlike normal advancing neural networks. It can process not only individual data points such as images, but also entire data streams. LSTM, for example, refers to activities such as non-segmented handwriting recognition, voice recognition and the identification of irregularities in texture images. A typical LSTM unit consists of a cell, an entry door, an exit door, and a forgetting door. The cell holds values for unpredictable periods of time, and the three doors govern the flows of knowledge inside and outside the cell. LSTM networks are well suited for classifying, analyzing, and forecasting based on time series data, because there may be delays of uncertain length in a time series between significant events. LSTMs have been developed to address the breakdown and disappearance gradient problems that can be found in conventional RNN preparation. Gap length insensitivity is a benefit of LSTM over RNN, hidden Markov models, and other sequence learning approaches in many applications.

3.3 Algorithm for Texture Classification using CNN-LSTM model

1. Load the texture dataset
2. For each image in dataset do
3. Build a CNN-LSTM model using Add Conv2D and MaxPooling2D.
4. Now creation of neural net create input layer as a convolution layer with 64 filters then middle layer with 128 filters and final output layer is a LSTM layer with output size 70.
5. Now create the dense neural net with 23 and activation softmax.
6. Compile the model with binary-cross entropy and also fit the model.

7. No of epochs for training the model can be customized.
8. Predict the accuracy of the model using accuracy- score metrics

3.4 Data Set

We have used three texture datasets such as Brodatz Texture Album, VisTex Dataset and DynTex Dataset for texture analysis and classification.

Brodatz Texture Album: The textures in Brodatz are well accepted and broadly used in texture classification. It is composed of 112 textures the album of Brodatz texture [37]. The Brodatz textures sample images are shown in Figure 3.

Figure 3: Brodatz texture album sample images.

VisTex Dataset The Vision Texture (VisTex) is collected by the Massachusetts Institute of Technology (MIT)[38]. This dataset is not commonly used but is more general. Figure 4 shows sample images from the dataset.

Figure 4: VisTex Dataset textures sample images.

DynTex Dataset: DynTex website is a vast set of interactive videos with good quality texture. Typically, dynamic textures are the result of processes like waves, smoke, and fire. Several real-world textures that occur in video databases are dynamic and static characteristics. Figure 5 shows sample images from the dataset.

Figure 5: DynTex textures sample images

3.5 Training and Experimental Evaluation

The proposed model was being trained on three different datasets for 25 epochs. The Test Set is used to track each classifier's output intended to be assigned for correct classifier for the class statistics over a range of empirically specified iterations. The unused evaluation collection is used at the last point for the final efficiency review of the fused classifiers. Occasionally, with certain research tests, the outcomes of both

classifiers' votes fall within each classifier's field of expertise, such that uncertainty matrix statistical knowledge is used to evaluate the precision of each epoch. The rate of misclassification is the percentage of invalid sample predictions predicted to be a class A while belonging to any other class. The Accuracy Rate is the percentage of true sample predictions expected to be Class A

In this experiment we use a held-out testing set to calculate the accuracy of the proposed model, to verity its performance on texture datasets as True_Positive (TP), True_ Negative (TN), Fasle_Positive and False_Negetive to compute the CNN-LSTM model performance with respect to Accuracy as:

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

4. EXPERIMENTAL RESULTS

In this section we will revise the performance CNN-LSTM algorithm with SVM, CNN* [39]. The Accuracy is measured with and compared with SVM and CNN*. CNN-LSTM an enumeration-based hybrid approach algorithm has shown its better performance on texture classification system. The Accuracy of the proposed CNN-LSTM model is measured by setting the Convolution matrix size to 2, convolution duration is set with the duration of the pooling layer by using max pooling. Using the Adam optimization algorithm over convolution matrix the pooling algorithm performs the sub-sampling to minimize the error function. We observed improvements in accuracy in the Convolution Neural Network by setting LSTM for output layer.

Table 1: Performance comparison of SVM, CNN* and CNN-LSTM on Brodatz texture dataset

Model	Accuracy %	Misclassification Rate
SVM	96.92	0.0914
CNN*	97.39	0.0813
CNN-LSTM	98.53	0.0542

Table 1 contrasts SVM, CNN* and CNN-LSTM algorithms to the identification effect by Accuracy on Brodatz dataset. We can see that the CNN-LSTM model's detection impact is stronger than that of other algorithms. Figure 6 provides a summary of the efficiency of algorithms SVM, CNN* and CNN-LSTM. It is ascertained from figure 6 that CNN-LSTM has achieved better accuracy performance over on Brodatz texture dataset.

Table 2: Performance comparison of SVM, CNN* and CNN-LSTM on VisTex texture dataset

Model	Accuracy %	Misclassification Rate
SVM	99.10	0.0375
CNN*	99.20	0.0266
CNN-LSTM	99.36	0.0222

Table 2 contrasts SVM, CNN* and CNN-LSTM algorithms to the identification effect by Accuracy on VisTex dataset. We can see that the CNN-LSTM model's detection impact is stronger than that of other algorithms.

Figure 7 provides a summary of the efficiency of algorithms SVM, CNN* and CNN-LSTM. It is ascertained from figure 7 that CNN-LSTM has achieved better accuracy performance over on VisTex texture dataset.

Table 3: Comparison of SVM, CNN* and CNN-LSTM on DynTex texture dataset

Model	Accuracy %	Misclassification Rate
SVM	99.50	0.0185
CNN*	98.55	0.0438
CNN-LSTM	99.57	0.0175

Table 3 contrasts SVM, CNN* and CNN-LSTM algorithms to the identification effect by Accuracy on VisTex dataset. We can see that the CNN-LSTM model's detection impact is stronger than that of other algorithms. Figure 8 provides a summary of the efficiency of algorithms SVM, CNN* and CNN-LSTM. It is ascertained from figure 8 that CNN-LSTM has achieved better accuracy performance over on DynTex texture dataset
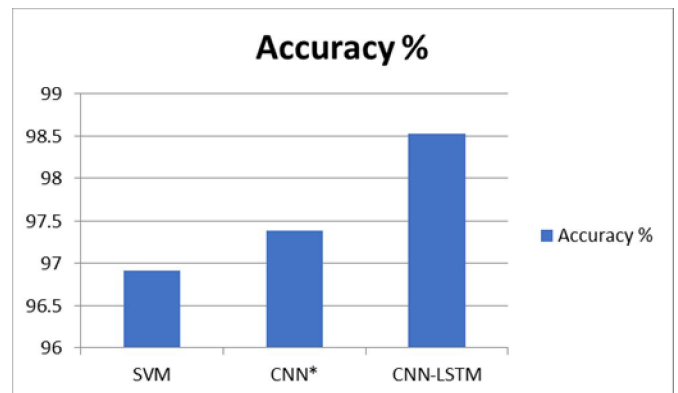

Figure 6: Efficiency of algorithms SVM, CNN* and CNN-LSTM with respect to Accuracy on Brodatz dataset
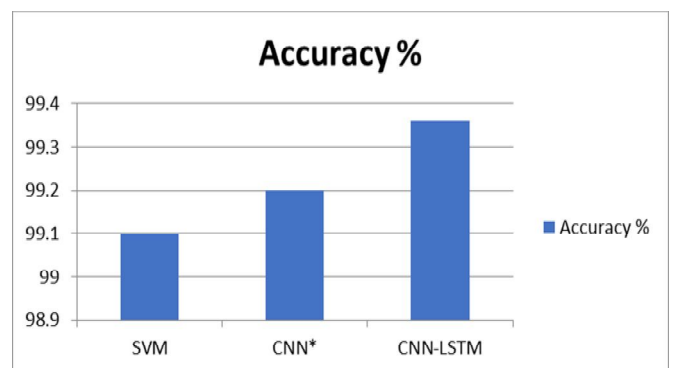

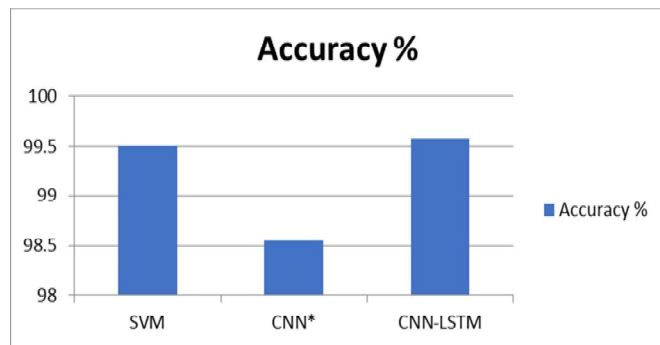**Figure.7:** Efficiency of algorithms SVM, CNN* and CNN-LSTM with respect to Accuracy on VisTex data set

Figure 8: Efficiency of algorithms SVM, CNN* and CNN-LSTM with respect to Accuracy on DynTex data set

5. CONCLUSION

The role of texture classification has important applications in a wide variety of domains, and while effective in several fields, it remains the focus of active research with the potential to increase classification levels. In this article, we develop an improved method using CNN-LSTM for texture classification. This model should automatically determine the efficient properties of the feature samples from texture images. Our experiments with 3 texture data sets have shown that this approach is capable of producing next-generation results. The CNN-LSTM achieves superior classification results than the existing CNN and SVM. The use of LSTM served as powerful invariant feature extractor, as post processing to CNN model certainly increased the accuracy of the classification rate and admired a notable impact on misclassification rate on three different classifiers.

Future work exploring the optimization of hyper-parameters within the convolutional neural network, such as the number of layers, the number of neurons per layer, etc. Because these hyper-parameters were fixed during the experiments on the present work. Furthermore, the technique of extract epochs during experimentation can be further investigated by experimenting with various methods, such as extractions of random functions. Additionally, the transfer learning can be extended for texture classification using convolution neural networks in conjunction with LSTM.

ACKNOWLEDGMENT

REFERENCES

1. M. Tuceryan, and A.K. Jain, **"Texture Analysis," The Handbook of Pattern Recognition and Computer Vision (2nd Ed)**, *World Scientific Publishing, Singapore*, pp. 207-248, 1998.

2. B. S. Manjunathi and W. Y. Ma, **"Texture features for browsing and retrieval of image data,"** *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 8, pp. 837–842, Aug. 1996
   https://doi.org/10.1109/34.531803

3. Chengjun Liu, Wechsler, H., **"A shape- and texture-based enhanced Fisher classifier for face recognition,"** *in Image Processing, IEEE Transactions on*, vol.10, no.4, pp.598-608, Apr 2001

4. Davis Larry S. **"Image Texture Analysis Techniques - A Survey,"** *Springer conference on Digital Image Processing*,1981.

5. R. M. Haralick, **"Statistical and structural approaches to texture,"** *in Proc. IEEE*, vol. 67, 1979, pp. 786–804.

6. G. R. Cross and A. K. Jain, **"Markov random field texture models,"** *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 5, no. 1, pp. 25–39, 1983.

7. X. Liu and D. Wang, **"Texture classification using spectral histograms,"** *IEEE Transactions on Image Processing*, vol. 12, no. 6, pp. 661–670, 2003.
   https://doi.org/10.1109/TIP.2003.812327

8. A. K. Jain and F. Farrokhnia, **"Unsupervised texture segmentation using gabor filters,"** *Pattern Recognition*, vol. 24, no. 12, pp. 1167– 1186, 1991.

9. A. K. Jain and K. Karu, **"Learning texture discrimination masks,"** *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 2, pp. 195–205, 1996.

10. Y. LeCun, L. D. Jackel, B. Boser, J. S. Denker, H. P. Graf, I. Guyon, D. Henderson, R. E. Howard, and W. Hubbard, **"Handwritten digit recognition: Applications of neural network chips and automatic learning,"** *IEEE Communication*, pp. 41–46, 1989.

11. Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, **"Gradient-based learning applied to document recognition,"** *Proc. of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

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

13. M. Petrou, and P.G. Sevilla, **Image Processing Dealing with Texture**, *John Wiley & Sons, West Sussex, England*, 2006.
   https://doi.org/10.1002/047003534X

14. S. Arivazhagan, L. Ganesan, and T.G.S. Kumar, **"Texture Classification using Curvelet Statistical and Co-occurrence Features,"** *ICPR*, 2006.

15. M.B. Othmen, M. Sayadi, and F. Fnaiech, **"Interest of the Multi-Resolution Analysis Based on the Co-occurrence Matrix for Texture Classification,"** *IEEE MELECON*, pp. 852-856, 2008.

16. J.Y. Tou, Y.H. Tay, and P.Y. Lau, "**One-dimensional Greylevel Co-occurrence Matrices for Texture Classification,**" *ITSIM*, vol. 3, pp. 1592-1597, 2008.

17. J.Y. Tou, K.K.Y. Khoo, Y.H. Tay, and P.Y. Lau, "**Evaluation of Speed and Accuracy for Comparison of Texture Classification Implementation on Embedded Platform,**" *IWAIT*, 2009.

18. I. Daubechies, "**Orthonormal Bases of Compactly Supported Wavelets,**" *Communications on Pure and Applied Mathematics*, vol. 41, pp. 909-996, 1988.

19. D.A.A. Nadi, and A.M. Mansour, "**Independent Component Analysis (ICA) for Texture Classification,**" *SSD*, 2008.

20. J.H. Kim, S.C. Kim, and T.J. Kang, "**Fractal Dimension Co-Occurrence Matrix Method for Texture Classification,**" *IEEE TENCON*, 2006.

21. J.G. Rosiles, S. Upadhyayula, and S.D. Cabrera, "**Rotationally-Blind Texture Classification using Frame Sequential Approximation Error Curves,**" *IEEE ICASSP*, pp. 1325-1328, 2008.

22. M.C. Amirani, and A.A.B. Shirazi, "**Texture Classification Using Cyclic Spectral Function,**" *CISP 2008*, pp. 834-838, 2008.
https://doi.org/10.1109/CISP.2008.687

23. Q. Xu, and Y.Q. Chen, "**Multiscale Blob Features for Gray Scale, Rotation and Spatial Scale Invariant Texture Classification,**" *ICPR*, 2006.

24. Liu, L., Chen, J., Fieguth, P., Zhao, G., Chellappa, R., Pietik¨ainen, M.: **From bow to CNN: Two decades of texture representation for texture classification.** *International Journal of Computer Vision* 127(1), 74–109 (2019)

25. Z. Guo, L. Zhang, and D. Zhang, "**A Completed Modeling of Local Binary Pattern,**" vol. 19, no. 6, pp. 1657–1663, 2010.

26. M. Nasirzadeh, A. A. Khazael, and M. Khalid, "**Woods Recognition Sys- tem Based on Local Binary Pattern,**" *Second International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN)*, no. 2, pp. 2–7, 2010.

27. J. Y. Tou, Y. H. Tay, and P. Y. Lau, "**A Comparative Study for Texture Classification Techniques on Wood Species Recognition Problem,**" *2009 Fifth International Conference on Natural Computation*, pp. 8– 12, 2009.

28. P. L. P. Filho, L. S. Oliveira, A. S. B. Jr., R. Sabourin, and E. D. T. Superieure, "**Forest Species Recognition Using Color-Based Features,**" *International Conference on Pattern Recognition*, vol. 0, pp. 4178–4181, 2010.

29. P. L. P. Filho, L. S. Oliveira, S. Nisgoski, and A. S. Britto, "**Forest species recognition using macroscopic images,**" *Machine Vision and Applications*, vol. 25, no. 4, pp. 1019–1031, 2014.

30. Y. L.Li, L.Feng, S.Liu, "**Local Co- occurrence Pattern for Color and Texture Image Retrieval,**" *12th World Congress on Intelligent Control and Automation (WCICA)*, 2016.

31. Y. Xu, M. Sonka, G. McLennan, J. Guo, and E.A. Hoffman, "**MDCT-Based 3-D Texture Classification of Emphysema and Early Smoking Related Lung Pathologies,**" *IEEE TMI*, vol. 25, no. 4, pp. 464-475, 2006.
https://doi.org/10.1109/TMI.2006.870889

32. A.D. Lillo, G. Motta, J.A. Storer, "**Multiresolution Rotation-Invariant Texture Classification Using Feature Extraction in the Frequency Domain and Vector Quantization,**" *DCC*, pp. 452-461, 2008.

33. V.N. Vasyukov, and N.V. Sysoev, "**Texture Image Segmentation Using Multiscale Wavelet-Domain Hidden Markov Model,**" *IFOST*, pp. 151-153, 2007.

34. LeCun, Y., Boser, B.E., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W.E., Jackel, L.D.: **Handwritten digit recognition with a back-propagation network.** *In: Advances in neural information processing systems.* pp. 396–404 (1990)

35. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: **Rethinking the inception architecture for computer vision.** *In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2016)

36. He, K., Zhang, X., Ren, S., Sun, J.: **Deep residual learning for image recognition.** *In: Proceedings of the IEEE conference on computer vision and pattern recognition.* pp.770–778(2016).
https://doi.org/10.1109/CVPR.2016.90

37. Brodatz, **Textures: A Photographic Album for Artists and Designers**, Dover, New York, 1996.

38. Jing Yi Tou1, Yong Haur Tay1, Phooi Yee Lau "**Recent Trends In Texture Classification: A Review**", *Symposium on Progress in Information & Communication Technology* 2009.

39. F. J. Huang and Y. LeCun, "**Large-scale learning with SVM and convolutional nets for generic object categorization,**" *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 284–291, 2006.

40. M.Subba Rao, Dr. B. Eswara Reddy "**Comparative Analysis of pattern Recognition Methods: An Overview**" Indian Journal of Computer Science and Engineering, volume 2, NO 3, 2011 ISSN: 0976-516.

41. M. Subba Rao and Dr. B. Eswara Reddy, "**An Overview of Pattern Recognition Methods on Texture Classification**", CiiT International Journal of Artificial Intelligent Systems and Machine Learning, July 2011.

42. Rufo.I.Marasigan et.al "**Copra Meat Classification using Convolutional Neural Network**" *International Journal of Emerging Trends in Engineering Research*, Volume 8, Issue 2, pp.435-439,2020
https://doi.org/10.30534/ijeter/2020/30822020