

Article - Biological and Applied Sciences

Virus Texture Classification of TEM Images Using Fusion of Chebyshev Moments and Resnet50 Features

Chandra Mohan Bhuma¹

<https://orcid.org/0000-0002-7566-4739>

Ramanjaneyulu Kongara²

<https://orcid.org/0000-0003-0711-0547>

¹Bapatla Engineering College, Department of ECE, Bapatla, Andhra Pradesh, India; ²PVP Siddhartha Institute of Technology, Department of ECE, Vijayawada, Andhra Pradesh, India.

Editor-in-Chief: Paulo Vitor Farago

Associate Editor: Marcos Pileggi

Received: 01-Oct-2021; Accepted: 22-Mar-2022.

*Correspondence: kongara.raman@gmail.com; Tel.: +91-8328175703 (R.K.).

HIGHLIGHTS-

- Classification of virus texture based on pre-trained CNN and ECOC classifier is proposed.
- Model accuracy improved using Chebyshev moments and the pruned features of Resnet50.
- The proposed algorithm outperformed in terms of classification accuracy.

Abstract: Classification of viruses is an essential step before suitable treatment of a disease. In this work, a novel approach for the classification of virus texture from TEM images is proposed. Chebyshev moments are used to classify virus textures with the Pre-trained convolution neural networks. Resnet50 is a pre-trained deep learning model used to classify images into object categories. Chebyshev moments and the pruned features extracted from the last global pooling layer of Resnet50 are fused to improve the classification accuracy. The fused feature vector along with corresponding labels is used to train a multiclass Error Correcting Output Code (ECOC) classifier to give the output. The proposed approach is tested on a standard benchmark virus texture dataset. Peak, mean, and median classification accuracies are calculated and compared with the state of the art approaches. In addition, various other classification metrics i.e., sensitivity, specificity, Mathews Correlation Coefficient, and kappa are given to justify the validity of the proposed method. 80% of the image dataset is chosen for training and the remaining 20% for testing. A peak classification accuracy of 90.33%, mean accuracy of 86.99% and median accuracy of 86.66% is achieved. Superiority of the proposed method is justified with simulations.

Keywords: Virus Texture classification; Chebyshev Moments; ECOC Classifier; Resnet50; Convolutional Neural Network.

INTRODUCTION

Detection and classification of viruses is a crucial step before suitable treatment of a disease. Imaging of viruses is not possible with commercial digital cameras as their size is very small. An expert needs to use a microscope for it. Transmission Electron Microscope (TEM) is used for imaging the viruses. Negative stain TEM [1] is more suitable for early diagnosis of virus. Different viruses have a different texture which is used to distinguish between them. Even though the type of virus can be identified by visual means through a

microscope, it requires a lot of experience and skill. In this regard, an automatic virus classifier is very useful. Here, Machine learning techniques are deployed to simplify the task of classification.

The sizes and shapes of the viruses vary from one category to the other. Mere sizes and shapes may not be sufficient to distinguish. In addition, if texture information is used then the classification task would be more effective. There are many works in the literature addressing the problem of texture analysis and the texture descriptors. Gabor filters, image moments, Haralick features, GLCM features and Local Binary Pattern (LBP) are some well-known features used in the texture analysis. Kylber and coauthors [1] have demonstrated the power of two local and two global texture descriptors in identifying the type of virus in the virus texture dataset. They have proved that a multi scale extension (MSE) and the radial density profiles (RDP) in Fourier domain do a better job than standard LBP. They have used a Random Forest classifier for classification. They have given an RDP measure and a variant of LBP, Local Binary Pattern Filtered (LBPF) and analysed textures of virus texture dataset for both object level and fixed scale datasets. A mean accuracy of 73.8% is reported on the object scale dataset.

Loris Nanni and coauthors [2] used various texture descriptors for image classification. They included quinary coding of LBP variants, an approach based on co-occurrence matrix, and ensemble of local phase quantization variants with ternary encoding. On the object scale dataset, they have achieved a mean accuracy of 80.7%. There are many works [3-5] on virus images in the literature.

J Y Ren and X J Wu [6] have proposed a novel image descriptor known as Covariance Descriptor (CovD) for representing the image. CovD encodes the second order statistics of the features extracted from the image pixels. They have tested the performance of CovDs on UIUC material dataset, Virus texture dataset, and human face recognition. The best accuracy obtained on virus texture dataset was $79.4 \pm 3.3\%$.

K X Chen and coauthors have improved the CovD by extending from Euclidean space to Semi Positive Definite (SPD) manifold. It is a low dimensional and more discriminative as claimed by the authors of [7]. These descriptors were tested on Cambridge hand-gesture (CG) dataset, ETH-80 dataset, MDS dataset and Virus cell dataset. For the virus cell dataset, the best accuracy obtained was only $77.93 \pm 5.03\%$. Instead of considering full dataset, a subset of dataset was chosen at random. This is obtained for the Log-Euclidean framework based arc-cosine kernel (LogE.Arc kernel).

MATERIAL AND METHODS

Image moments in Classification

Image moments are used in many diversified fields of computer vision, object recognition, and shape extraction. Both orthogonal and non-orthogonal moments [8] are available and have been used in image processing applications extensively. Orthogonal moments are able to remove the redundancy. Hence, orthogonal moments are preferred over non-orthogonal moments. Some of the moments are discrete ones and others are continuous ones. Zernike moments, Legendre moments, Chebyshev moments, Jacobi moments, Gaussian Hermite moments are orthogonal moments. Chebyshev moments are the moments obtained from the Chebyshev polynomials. For higher order moments, numerical instability in the computation of moments is a problem. Mukundan [9] has proposed an orthonormal version of Chebyshev moment computation even at the higher order and is briefed here.

Chebyshev moments are generated from the Chebyshev polynomials. One important property of the Chebyshev moments is that they preserve the orthogonality even in the discrete case. As this property is useful in the characterization of regions, they are suitable for texture analysis [10][11][12]. 2D Chebyshev moments [8] of a digital image are calculated as follows.

$$T_{mn} = \frac{1}{\rho(m, N)\rho(n, N)} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} t_m(x)t_n(y)f(x, y) \quad m, n = 0, 1, 2, \dots, N-1 \quad (1)$$

Where the function, $f(x, y)$, is a digital image and x, y represents the coordinates of the image in spatial domain. Size of the image is $N \times N$. The scaled Chebyshev polynomial $t_n(x)$ is given by:

$$t_n(x) = \frac{(2n-1)t_1(x)t_{n-1}(x) - (n-1)\left(1 - \frac{(n-1)^2}{N^2}\right)t_{n-2}(x)}{n} \quad n = 2, 3, \dots, N-1 \quad (2)$$

Where,

$$t_0(x) = 1$$

$$t_1(x) = \frac{2x + 1 - N}{N}$$

$$\rho(n, N) = \frac{N \left(1 - \frac{1}{N^2}\right) \left(1 - \frac{2^2}{N^2}\right) \dots \left(1 - \frac{n^2}{N^2}\right)}{2n + 1}$$

It is easy to compute the Chebyshev moments as per the equation (1). However, the computation of 2D Chebyshev moments, T_{mm} leads to erroneous results when the size of the image is large. A procedure for computation of higher order discrete orthogonal Chebyshev moments by constructing orthonormal version of Chebyshev polynomial is used for texture analysis. The feature vector is formulated from the Chebyshev moments after discarding the dc component.

Resnet50 Architecture

There are many pre-trained deep learning models and Convolutional Neural Networks (CNN) available for extracting the features of the images. Alexnet, Googlenet, Squeezenet, Resnet18, Resnet50, Inceptionnet, and Mobilenetv2 [13][14] are some examples under this category. Rather than considering a typical CNN architecture and training it from the scratch for the given dataset, utilizing a pre-trained CNN is a popular approach. Most of the pre-trained CNNs are trained on ImageNet dataset. There are millions of images in the ImageNet dataset. The number of object categories is 1000. Wide varieties of objects are there in the dataset. Resnet50 [15][16] is a 50 layers deep neural network. Hence, wide variety of features from various objects is already learnt by the Resnet50 network.

Resnet50 accepts 224x224 size images as input. Hence, images of both training and test dataset must be resized to 224x224 before applying to Resnet50. Extracted features from lower level layers represent low level features. Deeper layers extract high level features. Pooling layers are used for feature vector size reduction. In 5 stages, it has 48 convolutional layers, 1 max pooling layer and one global average pooling layer. Its architectural details are shown in Figure 1. Approximately twenty three millions of trainable parameters are there in Resnet50.

The ResNet-50 model consists of a convolution and Identity block in each stage. There are three convolution layers in each block. Input activation dimension and output activation dimension is same in identity block and is not the case with the convolution block. A two dimensional convolution layer is there in the short cut path of convolution block and is absent in the identity block.

Virus Texture Dataset v.1.0

The Virus Texture Dataset v.1.0 [1] consists of TEM images of various types of virus. There are 1500 images in the dataset. 15 categories/classes are there. Resolution of the images is 41x41. All are grayscale images. The diameter is relatively constant within a virus type. Shape of the virus ranges from icosahedral to highly pleomorphic. The pixel sizes used for imaging are from 0.5 to 5 nm. There is a sampled version also in the dataset. Both 16 bit and 8 bit images are there. There are known under fixed scale and object scale. Size of a pixel is 1 nanometer in fixed scale and 20 pixels are used to represent the virus in the object scale.

In the present work, the object scale dataset is used and is shown in Figure 2. Sample images from one class 'Dengue' are shown in Figure 3.

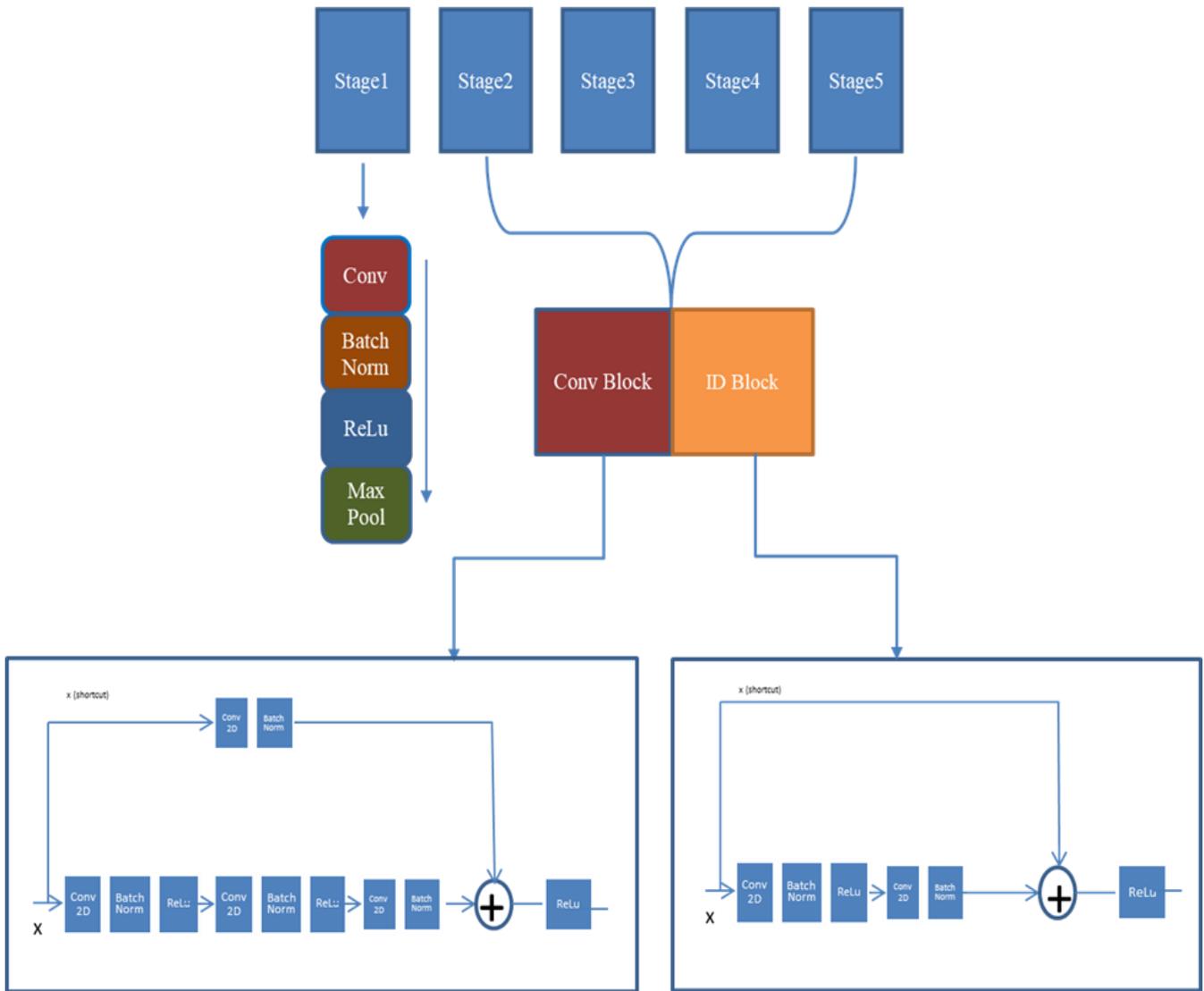


Figure 1. Resnet50 Architecture

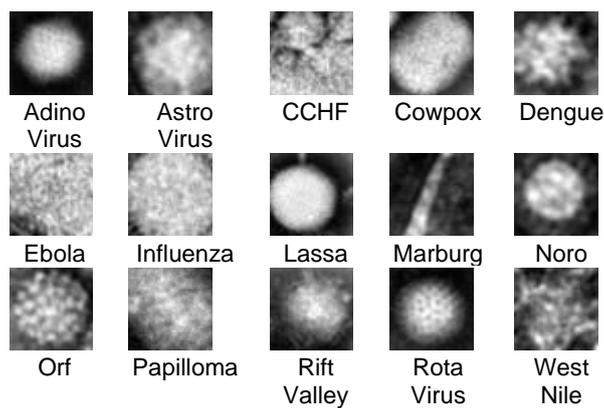


Figure 2. Sample Images of each category from virus texture dataset



Figure 3. Sample images from the Dengue Class images

Proposed Algorithm

The proposed algorithm uses Resnet50 which is a Pre-trained convolution neural network and Chebyshev moments for the classification of virus texture from TEM images. The virus image dataset is divided into two parts. Part1, 80% of the image dataset, is chosen for training and the remaining 20% as part2 for testing. 70%, 30% and 75%, 25% are the other possibilities the splitting the data. For an image of size $N \times N$, Chebyshev moments of size $2N-2$ are computed for all the images in training dataset. First feature vector is formed from this. The images are then resized to 224×224 and applied to Resnet50 as it accepts 224×224 images only. All the training images are applied to the Global Average Pooling layer of the Resnet50 network. Second feature vector is obtained. Both the feature vectors, i.e., from the moments and from the Global Average Pooling layer are concatenated so that a final feature vector is obtained which is used for the classification.

An Error Correcting Output Code (ECOC) classifier [17] is trained with these features vectors along with the corresponding labels. The hyper parameters are chosen so as to maximize the classification accuracy. As the classification of virus texture dataset is a multiclass problem, ECOC classifier is chosen. In ECOC, the multi-class [18][19][20][21] problem is divided into various binary class problems. It requires coding and decoding. A basic binary learning mechanism is also required. SVM, KNN, and Linear Discriminant are some of the popular examples for binary learners. In this work, SVM binary learner is chosen. Results of predictions on binary learner are aggregated in ECOC.

Chebyshev moments and the feature vector from the last Global Average Pooling layer are calculated for the test dataset also. The two features are concatenated and given to the trained ECOC classifier. Various binary losses can be considered while predicting with ECOC classifier. The outline of the proposed algorithm is shown as a flow chart in Figure 4.

There are many choices that can be used in the ECOC classifier. The type of basic binary learner & kernel in the binary learner, coding, decoding, and the binary loss function in the prediction are few options to quote. The overall classification test accuracy does depend on the selection of these parameters. One can use optimization strategies for selection of these parameters or fine tune manually. There are many combinations of selections and hence it is difficult to obtain the best manually. Hence, optimization is preferred.

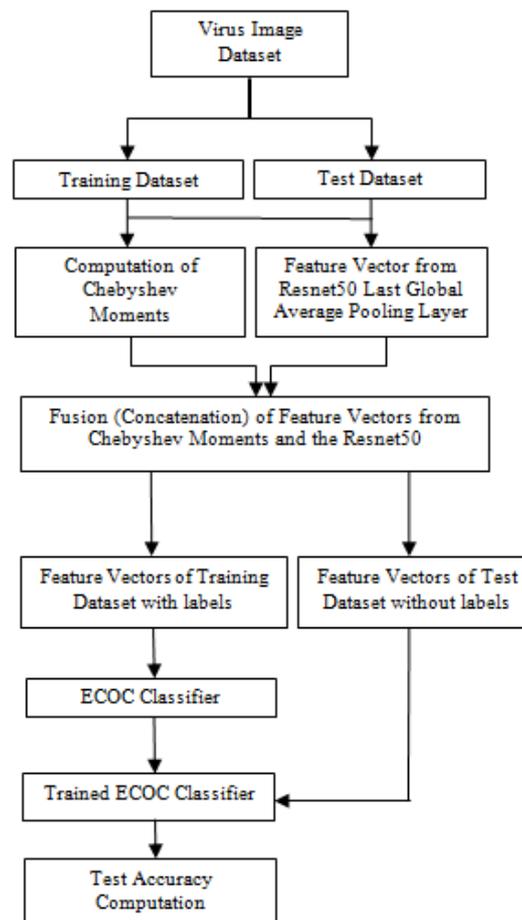


Figure 4. Flow chart of the proposed algorithm**RESULTS**

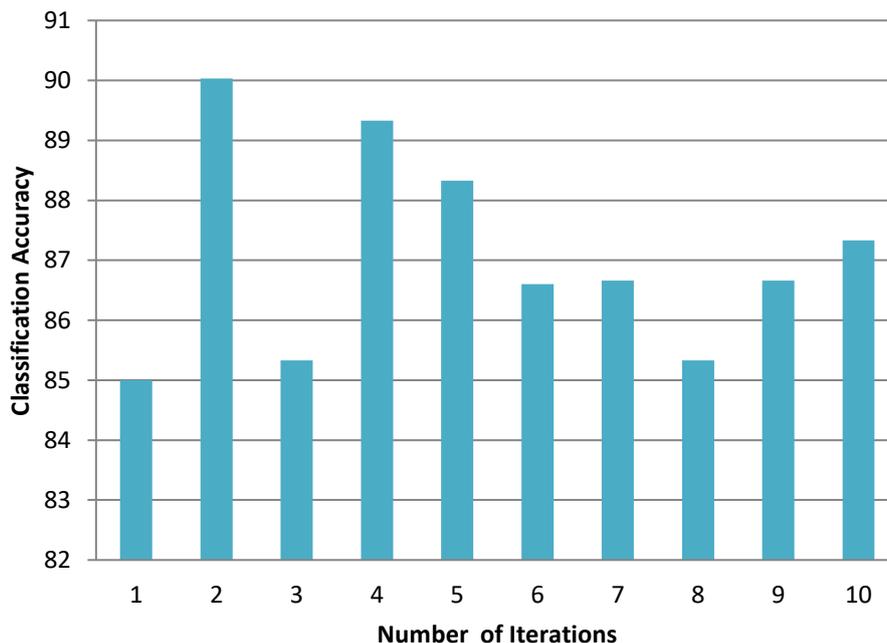
Simulations are carried using MATLAB R19b software. The system configuration consists of an i3 processor with 8 GB RAM with no GPU. The size of the images in the virus texture dataset is 41×41. Hence, the size of the feature vector obtained from the Chebyshev moments is 80 (2×41-2). Size of the feature vector obtained from the Global Average Pooling layer of Resnet50 is 2048. However, in the process of fine tuning, the size of the feature vector is restricted to 2000 by removing the last 48 features. Hence, the total size of the feature vector is 2080. There are total 1500 images in the virus texture dataset. 1200 images are chosen for training the classifier and 300 images are used in the testing process.

Various binary classifiers can be used in the multi class classification using ECOC. A classification accuracy of 75% with KNN, 77.33% with discriminant classification, 88.33% with linear classification and 62.33% with tree classification are obtained. The best classification accuracy is obtained with SVM learners i.e. 90.33%.

Choosing the type of binary learner loss function is also crucial while predicting the labels of the test dataset. The best test classification accuracy of 90.33% is obtained when the binary loss function is chosen as 'hamming'. Let us assume that y_j is a class label for a particular binary learner (in the set $\{-1,1,0\}$) and s_j is the score for observation j , then $g(y_j, s_j)$ denotes the binary loss associated with the binary learner

and is given by $[1 - \text{sign}(y_j s_j)]/2$. The other binary learner loss functions (and their classification accuracies given in brackets) that are popular are Binomial deviance (89.33%), exponential (86%), hinge (89%), linear (73.33%), and logistic (84.33%) functions.

Ten iterations are carried, as the random selection of training and testing images affects both the training and testing classification accuracy. For the 10 iterations, the testing classification accuracies are shown in Figure 5. The highest classification accuracy obtained is 90.33%.

**Figure 5.** Classification accuracy versus Number of iterations

The mean accuracy is 86.99% and the median accuracy is 86.66%. It may be inferred that, out of 15 classes, 4 classes (Adeno virus, Astro virus, Papilloma, Rota virus) were accurately classified without any misclassified samples. Next best ones are 'Marburg', 'Orf', and 'West Nile' with one misclassified sample.

From the Figure 6, the performance of the classifier on 'Dengue' class is not that good. Seven samples are misclassified. As the data set is a multiclass one, the performance measures given here are averaged across all the classes. Various classification metrics are computed and are given in Table 1.

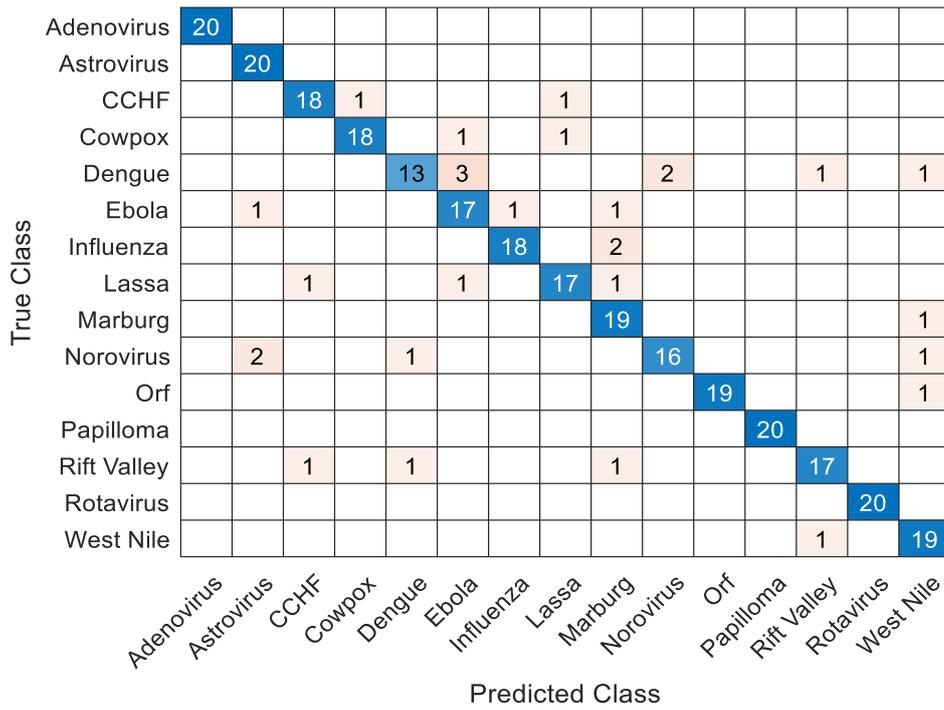


Figure 6. Confusion matrix for the test data

Table 1. Computed values of various classification metrics

| Parameter | Value |
|----------------------------------|--------|
| Accuracy | 0.9033 |
| Error | 0.0967 |
| Sensitivity | 0.9033 |
| Specificity | 0.9931 |
| Precision | 0.9067 |
| False Positive Rate | 0.0069 |
| F1 Score | 0.9024 |
| Matthews Correlation Coefficient | 0.8970 |
| Kappa | 0.2232 |

The results obtained with the proposed algorithm are compared with the existing works in the literature and are shown in Table 2. Classification accuracy of the proposed method is better in comparison with the existing methods [6][7].

The performance of the proposed algorithm is obtained with different Resnet architectures like Resnet18, Resnet34, Resnet50, Resnet101 and Resnet152. Results are shown in Figure 7. It is observed that the performance with Resnet50 is better than the others.

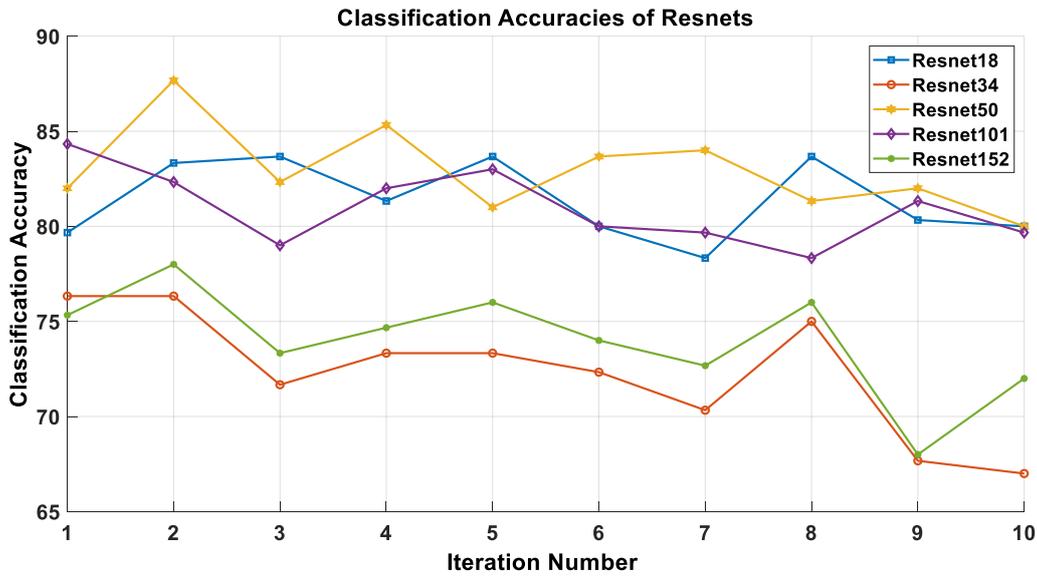


Figure 7. Classification accuracy versus iteration number with Resnet18, Resnet34, Resnet50, Resnet101 and Resnet152.

The feature vector obtained from the Chebyshev moments and the feature vector obtained from the Global Average Pooling layer of Resnet50 are fused (Concatenated) to improve the performance of the proposed method. In this fusion process, there are two cases. In case 1, full features obtained from the Resnet50 are used. In case 2, the last insignificant features are removed. These removed features are known as pruned features. The process of removing insignificant features is called pruning.

From the Figure 8, Figure 9 and Figure 10, it is observed that the performance of the proposed method is better with the fusion of Chebyshev moments and pruned features of Resnet50.

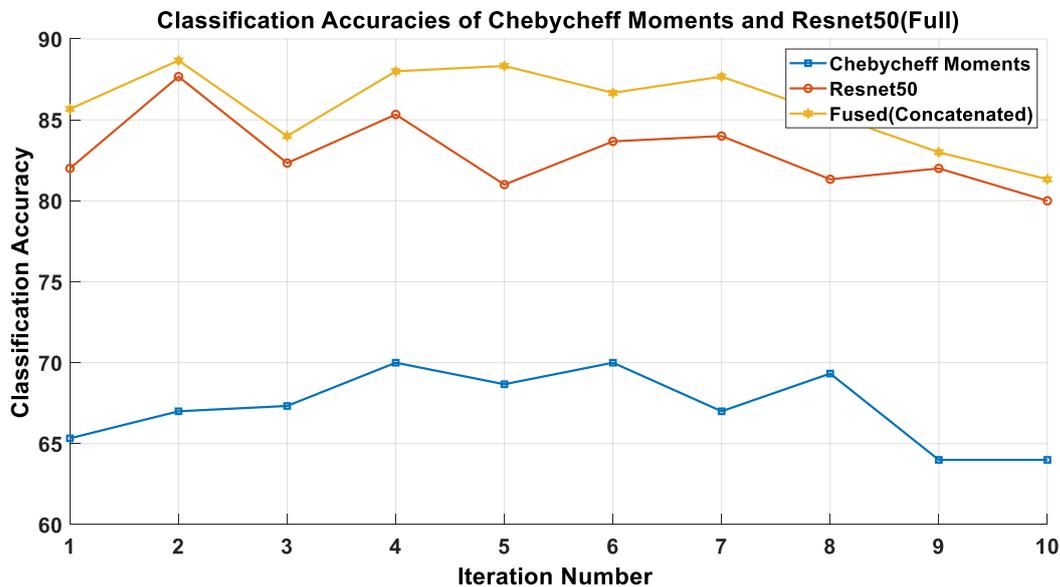


Figure 8. Classification accuracies with Chebyshev Moments, Resnet50 and Fused full features.

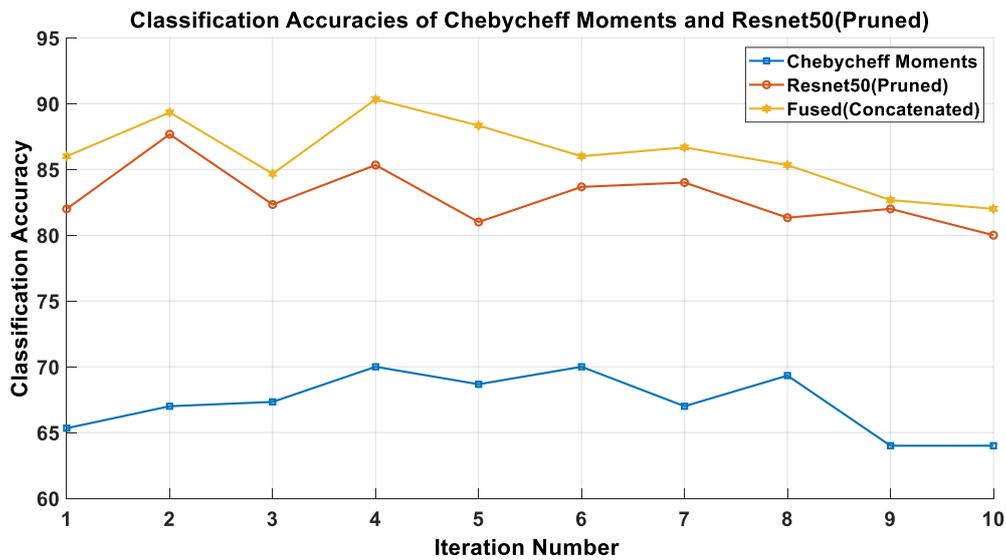


Figure 9. Classification accuracies with Chebyshev Moments, Resnet50 and Fused pruned features.

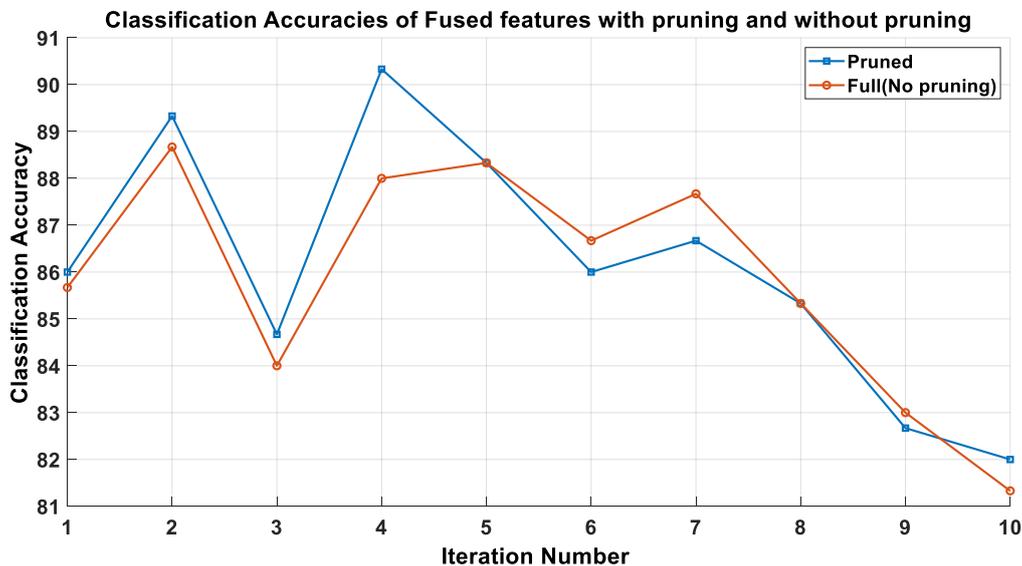


Figure 10. Classification accuracies of Fused features with pruning and without pruning.

DISCUSSION

An algorithm for effectively classifying the TEM images of virus particles is presented in this work. To analyse the texture content present in the virus structure, the fusion of Chebyshev moments and the high level features extracted from the deeper layers of Resnet50 are considered. With appropriate choosing of the classifier, kernels, binary loss function in the prediction, it is possible to classify the texture data of virus images with good accuracy. Peak classification accuracy of 90.33% is achieved with the proposed work. A mean accuracy of 86.99% is achieved with a strategy of 80% training and 20% testing. Consistent results of more than 85% classification accuracy are achieved even when the entire data is randomly shuffled. This proves the robustness of the features obtained from the Chebyshev moments and the deeper Resnet50 features. Significance of the proposed methods lies in the usage of the traditional image moments with the CNN features. Traditional image moments i.e., Chebyshev moments are able to capture the shape and texture very well with limited number of features. Hence, CNN features can be supplemented with the image moments to have higher classification accuracy.

CONCLUSION

A novel approach is proposed for the classification of Virus texture data which is useful in disease treatment. Fusion of Chebyshev moments and pruned features of ResNet50 effectively improves the classification accuracy. Compared to the state of the art works, the proposed algorithm performs better in terms of classification accuracy both in mean sense and peak sense. Further improvement can be achieved

with the extraction of low level and middle level features from Resnet50 and by fine tuning the feature vector selection. Further work can be done in the direction of utilizing the meta heuristic optimization algorithms in the selection of appropriate layers from Resnet50 for effective classification.

Funding: This research received no external funding.

Conflicts of Interest: Authors express that there is no conflict of interest with respective publication of this work in this journal.

REFERENCES

1. Kylberg G, Uppström M, Sintorn IM. Virus Texture Analysis Using Local Binary Patterns and Radial Density Profiles. Martin César San, Kim S-W, editors. Progress in pattern recognition, image analysis, computer vision, and applications. In Iberoamerican Congress on Pattern Recognition (CIARP); 2011 November 15-18; pucón, Chile. Berlin: Springer; 2011. pp. 573–80.
2. Nanni L, Paci M, Brahnam S, Ghidoni S, Menegatti E. Virus image classification using different texture descriptors. In The International Conference on Bioinformatics and Computational Biology (BIOCOMP'13); 2013 July 22-25; Las Vegas. Providence, RI, USA: CSREA Press; 2013. pp. 56-61.
3. Dos Santos FL, Paci M, Nanni L, Brahnam S, Hyttinen J. Computer vision for virus image classification. Biosyst Eng. 2015 Oct 1;138:11-22.
4. Ito E, Sato T, Sano D, Utagawa E, Kato T. Virus particle detection by convolutional neural network in transmission electron microscopy images. Food Environ Virol. 2018 Jun; 10(2): 201–8.
5. Kylberg G, Uppström M, Hedlund KO, Borgfors G, Sintorn IM. Segmentation of virus particle candidates in transmission electron microscopy images. J. Microsc. 2012 Feb; 245(2):140–7.
6. Ren JY, Wu XJ. Vectorial approximations of infinite-dimensional covariance descriptors for image classification. Comput Vis Media. 2017 Dec; 3(4):379–85.
7. Chen KX, Wu XJ, Ren JY, Wang R, Kittler J. More About Covariance Descriptors for Image Set Coding: Log-Euclidean Framework Based Kernel Matrix Representation. In Proceedings of IEEE/CVF International Conference on Computer Vision Workshops (ICCVW); 2019 27 Oct-2 Nov; Seoul, South Korea. Washington; Tokyo: Conference Publishing Services, ICS; 2019. pp. 2923-32.
8. Flusser J, Suk T, Zitová B. 2D and 3D image analysis by moments. 1st ed. [Place unknown]: John Wiley & Sons; 2016 Dec 19. 560 p.
9. Mukundan R. Some Computational Aspects of Discrete Orthonormal Moments. IEEE Trans Image Process. 2004 Jul 19; 13(8):1055-9.
10. Marcos JV, Cristóbal G. Texture classification using discrete Tchebichef moments. J Opt Soc Am. 2013 Aug 1; 30(8):1580-91.
11. Di Ruberto C, Putzu L, Rodriguez G. Fast and accurate computation of orthogonal moments for texture analysis. Pattern Recognit. 2018 Nov 1; 83:498-510.
12. Kotoulas L, Andreadis I. Fast Computation of Chebyshev Moments. IEEE Trans Circuits Syst Video Technol. 2006 Jul 24; 16(7): 884-8.
13. Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L. ImageNet: A large-scale hierarchical image database. In 2009 IEEE Conference on Computer Vision and Pattern Recognition; 2009 Jun 20-25; Miami, FL, USA. [Place unknown]: IEEE; 2009 August. pp. 248-55.
14. 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 (CVPR); 2016 June; Los Vegas, USA. [Place unknown]: IEEE; 2016. pp. 770-8.
15. Keras API reference / Keras Applications [Internet]. [Place unknown: publisher unknown]; [Cited 2020 Feb 12]. Available from: <https://keras.io/api/applications/#resnet50>.
16. Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems; 2012 Dec; Lake Tahoe, USA. [Place unknown]: IEEE; 2012. pp. 1097–105.
17. Escalera S, Pujol O, Radeva P. On the Decoding Process in Ternary Error-correcting Output Codes. IEEE Trans Pattern Anal Mach Intell. 2008 Nov 7; 32(1):120–34.
18. AllweinEL, Schapire RE, Singer Y. Reducing multiclass to binary: A unifying approach for margin classifiers. J Mach Learn Res. 2000 Dec; 2000(1):113-41.
19. Kindermann J, Leopold E, Paass G. Multi-class classification with error correcting codes. Treffen der GI-Fachgruppe. 2000 Oct; 1(3).

20. aliBagheri M, Montazer GA, Escalera S. Error correcting output codes for multiclass classification: application to two image vision problems. In The CSI International Symposium on Artificial Intelligence and Signal Processing (AISP 2012); 2012 May 2-3; Shiraz, Iran. [Place unknown]: IEEE; 2012 Sep 27. pp. 508-513.
21. Dietterich TG, Bakiri G. Solving multiclass learning problems via error-correcting output codes. *J.Artif.Intell.Res.* 1994 August; 2(1): 263-86.



© 2022 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY NC) license (<https://creativecommons.org/licenses/by-nc/4.0/>).