Image Classification for Identifying Social Gathering Types

Chapter · October 2022

DOI: 10.1007/978-3-031-19958-5_10

6 authors, including:

Sumona Yeasmin
East West University (Bangladesh)
4 PUBLICATIONS **23** CITATIONS
SEE PROFILE

Nazia Afrin
East West University (Bangladesh)
4 PUBLICATIONS **23** CITATIONS
SEE PROFILE

Ahmed Wasif Reza
East West University (Bangladesh)
227 PUBLICATIONS **2,186** CITATIONS
SEE PROFILE

Image Classification for Identifying Social Gathering Types

Sumona Yeasmin[1], Nazia Afrin[1], Kashfia Saif[1], Omar Tawhid Imam[2], Ahmed Wasif Reza[1(✉)], and Mohammad Shamsul Arefin[3,4(✉)]

[1] Department of Computer Science and Engineering, East West University, Dhaka, Bangladesh
wasif@ewubd.edu
[2] Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology, Dhaka, Bangladesh
[3] Department of Computer Science and Engineering, Daffodil International University, Dhaka 1341, Bangladesh
sarefin@cuet.ac.bd
[4] Department of Computer Science and Engineering, Chittagong University of Engineering and Technology, Chattogram 4349, Bangladesh

Abstract. Convolutional neural networks are current times state-of-art algorithms widely used in image classification. This paper has explored the image classification of social gatherings with state-of-the-art neural network models. We introduce image classification with the modified VGG16 model and the modified InceptionV3 model. Images are first pre-processed and then given input to the models for multi-class classification. We have modified layers in the models, resulting in the best accuracy for our dataset. Data augmentation and layer modification schemes are applied in this paper. The algorithm learns to identify the classes of an image by performing feature extraction and data augmentations of each image. Throughout this research, we discovered that the approaches suggested in this paper improve the performance of the models. Our task was based on four classes of social gathering images. We concluded that the layer-modified VGG16 model with augmentation gives us the best results with a training accuracy of 90.99% and validation accuracy of 87.18%.

Keywords: Social gathering · VGG16 · Data preprocessing · Image classification

1 Introduction

Image classification is a significant sector of machine learning and computer vision. Image classification is simply a process of images to identify the information of a picture based on features. Image classification has been an important research topic in computer vision for decades. Humans can do the task of image classification pretty smoothly, but it is expensive in the case of computers. In general, each image is made of pixels representing different values. Computers need more space to store data and

P. Vasant et al. (Eds.): ICICO 2022, LNNS 569, pp. 1–13, 2023.
https://doi.org/10.1007/978-3-031-19958-5_10

more computational power to perform image classification. In real-time, making decisions based on the input is impossible because it takes more time to perform these many computations to provide results [1]. Machine learning models can only extract a particular number of features of images. They are unable to distinguish pieces from the training samples. This disadvantage is reduced by deep learning. Deep learning (DL) is a branch of machine learning that can learn through its neural network and backpropagation mechanism. Image classification is a worthwhile study topic since it has several applications in pattern recognition and image processing [2].

This research paper has analyzed three different state-of-art models for image classification and their performance to identify four types of social gathering scenarios.

2 Literature Review

In image processing, image classification has been a popular research area. This section has studied related work for image classification with CNN. We have explored several pieces to enrich our concept about CNN, and they work in the category of images. This paper has made a model to classify social gatherings of four different kinds with the CNN and VGG16 models. To understand the concepts of artificial neural networks, we have explored the other works to do a literature review.

Xue Ren et al. [2] proposed an image classification method in their study that combines (CNN) [2] and eXtreme Gradient Boosting (XGBoost) [2]. The authors have presented CNNXGBoost. The proposed model in this study discusses making a feature extractor using CNN. The proposed extractor is also trainable, making it a more robust architecture. This method dynamically obtains features from input. At the top level of the network, this method works as a recollection. The authors have conducted experiments on the benchmark MNIST [2] and CIFAR-10 [2] datasets. According to the authors' experiments, they have proven to obtain better performance.

Kaniz Fatema et al. [3] have developed a model using Canny Edge Detection and line segmentation [3] to automatically detect a book based on an image of the text taken from a random position. According to the authors, the OCR engine segments images to extract text or book titles. Furthermore, the authors have explained the maximum string-matching score. For all database data, the greatest string-matching score is determined using extracted text, and all book titles are shown [3].

Yanan Sun et al. [4] have proposed an automatic CNN architecture design method using genetic algorithms [4]. The authors of this study's proposed algorithm explained an "automatic" characteristic. For the provided input images, this "automatic" feature produces a stable and robust CNN architecture. Using the GA, this work proposes an architectural design approach for CNNs (in short, called CNN-GA) [4], which can discover the best CNN architecture for image classification and computer vision-related tasks [4]. Hyung Tae Lee et al. [5] have developed a deep convolutional neural network (CNN) image classification [5]. The authors have applied a mechanism to make the proposed CNN deeper for computer vision tasks. The authors have proposed a deep contextual CNN network to optimize local contextual interactions. This is accomplished by combining the spectral connections of pixel vectors [5]. The authors of this study have achieved a convolutional filter bank with several scales [5]. The authors have explained

how they have combined the initial spatial and spectral features [5]. The authors have fed an integrated feature map through a fully convolutional network for predictive results [5].

Vaddi et al. [6] have presented an approach for remotely observing hyperspectral images using normalization and CNN [6]. The authors of this study have normalized HSI data by reducing its scalar values. Then, the authors' proposed methods have been extracted using Probabilistic spectral and spatial information Principal Component Analysis (PPCA) [6] and Gabor filtering [6]. The resulting HSI image is merged with the original HIS image and sent to CNN [6]. Each of the three convolution and pooling layers in the final image [6]. Xiaofei Yang et al. [7] have exploited deep learning methods to identify picture categorization challenges [7]. To enhance classification, the authors presented a strategy to regulate spatial context and spectral correlation. For image classification, this study used four deep learning models with multidimensional CNN, including a two and three-dimensional CNN, a recurrent two-dimensional CNN, and a recurrent three-dimensional CNN [7].

Michael Blot et al. [8] have proposed to modify the convolutional block of CNN to share more information layer after layer [8]. The modified CNN has some invariances too. The convolution maps explore positive and negative scores. This score obtains the CNN maps. This study received the behavior by fine-tuning the activation function. The authors of this study doubled the maps with specified functions to achieve the proposed pipeline, which they call "the MaxMin approach." The authors conducted trials on two datasets, MNIST [8] and CIFAR-10 [8], and found that the author's suggested techniques MaxMin convolutional network outperforms regular CNN in terms of accuracy.

MaxMin Net Architecture combined with ReLU activation function negative detections give information similar to weak negative identification [8]. Authors have proposed the two layers keeping the block's output the same [8]. According to the study, the MaxMin method maintains negative detection values across the network. This technology improves the efficiency of data transit through the network. Li et al. [9] have categorized lung image patches with interstitial lung disease. Researchers of this study have used a Convolutional Neural Network (CNN) with a shallow convolution layer [9]. This research provides a modified CNN framework that automatically learns to extract image characteristics appropriate for classification from lung input images. This study has also proposed using convolution Image pixels as input to the CNN. In image classification tasks, one or more multidimensional pixel value matrices are fed to the network as input of the images, and the multidimensional matrix is generated as output. The authors have normalized lung image patches sent as the input making the unit variance and zero means. By integrating pooling methods with surrounding components, the authors could minimize the number of output neurons in the network layer [9].

To speed up and stabilize neural network training, this study has applied Back Propagation. The authors have also used batch learning which improves learning speed and accuracy. At the start of each training iteration, the dropout map with the exact size in each layer is randomly initialized [9].

Gong et al. [10] have introduced a general problem of traditional deep learning models. Because of the limited training specimen, the thoroughly learned models may not be sufficient. This is attributable to images with large intraclass variance [10] and

low interclass variance [10]. Convolutional neural networks (CNNs) with multi-scale convolution (MS-CNNs) are proposed by the authors [10]. In the case of hyperspectral images, extraction of deep multi-scale features overcomes this issue [10]. According to this study, deep metrics are frequently performed using MS-CNNs [10] to increase the image's capabilities. This approach enhances interclass disparity while decreasing intraclass conflict [10]. The authors then indicated that they would provide a DPP-based deep metric learning [10]. The metric parameter is synthesized with DPP priors in this learning process [10]. Making a set of metric parameter factors improves the ability of the input image [10].

Alex Krizhevsky et al. [11] The authors have introduced a standard deep CNN made with five convolutional blocks and three fully connected layers with a Softmax function. This study has presented a convolutional block combined with a filter layer to filter the input. This output passes through an activation function. The goal of the activation function is to increase the network's capacity. In image classification, the most valuable part is ReLU, as the ReLu performs normalization [11]. LeCun et al. [12] have introduced a set of two convolutional, fully connected layers. The authors presented a Gaussian connection layer integrated with a set of additional layers for pooling. On large-scale image collections, more profound and broader networks, such as AlexNet [12], began to be built [12]. The authors found the essential phrases by using efficient network learning to take input of any size and produce an output of the same size. The primary mechanism of this paper is fine-tuning the segmentation task of the traditional neural networks. Further, the authors have introduced a skip architecture that can take the semantic information of layers to produce detailed segmentation with more accuracy.

Jianjun Qian et al. [13] have introduced images by extracting the images' features. The authors have emphasized that the image composition mechanisms strongly influence feature extraction from the image [13]. This study on image classification has been effectively done using the hierarchical image decomposition approach. The authors have explained that they have decomposed each image into a series of semantic components. According to this study, using various feature extraction methods, the semantic image content can be matched with other images [13].

C. Silva et al. [14] present a Convolutional Neural Network method for extracting characteristics from images of cattle brands and Support Vector Machines for classification. The authors have separated the stages into six different segments. Firstly, the authors have selected a database of images. Next, the authors have pre-trained a CNN. The authors have pre-processed the image set to feed it to the pre-trained CNN in the further steps, following extracting the features of the images for classification. The authors have also experimented with a machine learning model (SVM). This study has trained SVM and sorted the data for practical reasons. Finally, the authors have conducted the experiment results and evaluated them.

Simonyan et al. [15] have proposed a plan to expand the depth of CNN, called VGG-16. VGG-16 is a 16-layer convolutional neural network [15]. Authors have used an architecture with multidimensional convolution filter matrices. An analysis with more depth reveals that expanding the depth to 16–19 weight neural layers improves the overall design [15]. Szegedy et al. [16] have stated that in various use cases such as computer vision and big data, minor parameter frequency and computational efficiency

are enabled [16]. This study has explored ways to scale up networks. The study has utilized the increased computation as efficiently as possible. The authors have achieved it by suitably factorizing convolutions and intense regularization.

The Feed-forward networks can be represented by an open chained graph derived from the input layer(s). This chain establishes a clear path for the data [16]. The researchers discovered that increasing the number of activations in a CNN network allows for additional features to be included. In this investigation, they are using the proposed model. The authors argue that networks will train more quickly. According to the findings, spatial aggregation across lower dimensional embeddings may be done without losing power [16]. By stabilizing the filter number, the study enhanced the network's outcomes. In addition, the authors increased the network's breadth and depth, resulting in higher-quality networks. The authors have proposed many architectural ideas for scaling up convolutional networks. With a low calculation cost, this approach can yield higher precision.

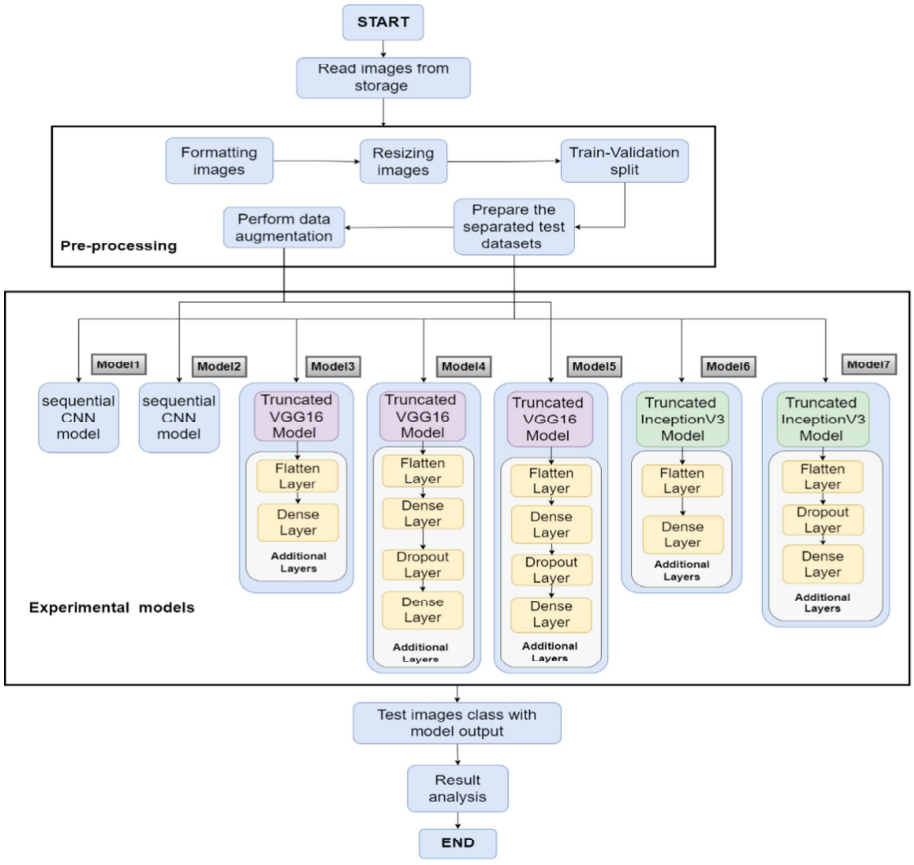

Fig. 1. System architecture

3 System Architecture and Design

The proposed framework's workflow is represented in Fig. 1. The components of the proposed framework are data pre-processing, training of the model, and social gathering image classification with deep neural networks.

This section has explained the working mechanism of the developed model for social image classification. The research process has two main areas: pre-processing the data and the experimental model. For pre-processing, we have resized the images. Next, we have split the dataset into train tests followed by data augmentation. For the experiments, we have taken three state-of-art models, CNN, VGG16, and Inception V3, to test the dataset and the performances. The test images are validated with the model output for further experiments.

3.1 Dataset Description

The dataset for this project is made for different domains or types (Convocation, Prayer, Meeting, Concert). For each class, we have taken 100 images initially. Four hundred photos from four different domains were used to make this dataset. Each image about the social gathering is mainly chosen in Bangladesh's context. The images are taken from google. We have taken colored images to feed the model after pre-processing for this project. The images are augmented later for training the models more accurately. Features are extracted from the photos of different classes, including the dark color of the convocation gown and the prayer images' white intensity. Initially, the dimensions of the pictures were different, but before feeding to the model, the photos were reshaped. The developed models in this paper take the color of the images as the main feature. Figure 2 shows the input image data for this project.

Fig. 2. Sample inputs of the dataset

3.2 Dataset Pre-processing

In this work, we have done some pre-processing on our dataset. The pre-processing of image data is essential for better accuracy and model outcome. We have resized the data by 128 * 128. Next, we have augmented the data. Data augmentation generates samples by transforming training data to improve the accuracy and classifiers. The robustness of the classifier is also enhanced by data augmentation. We have also split the data into training and testing to check model accuracy. The train-test split has been done before feeding data to the model. Finally, we have performed a label encoder as our dataset classes are categorical. We have taken the four class names and converted them into numeric with label encoding to make the model understand.

4 Implementation and Experimental Result

This section presents the detailed implementation process and experimental setup of research along with the model's performance evaluation.

4.1 Experimental Set-Up

Windows Operating system – We have conducted all the experiments in a Windows operating system in this project. The system configuration used in this project, Intel(R),Core(TM) i5-8250U, CPU limit: 1.60 GHz 1.80 GHz, Installed RAM capacity: 16.0 GB. The system type is a 64-bit operating system, and the processor is an x64-based processor.

Google Colab – Google Colab is a powerful python coding tool offering Zero configuration, Free accessed GPUs and Easy sharing. Colab is a platform provided by Google to write codes in python with all the libraries.

Implementation
For classification, we have taken seven CNN models, including two pre-trained models such as VGG16 and inceptionV3, to compare the accuracy of the models and understand the type. We have studied and tested the CNN model with our own set of configurations. Furthermore, we have also augmented the data to understand the influence on the accuracy of the models. We have used the raw VGG16 pre-trained model, followed by layer configuration and data augmentations. For the incenptionV3 model, we have done the same.

Keras Sequential CNN Model 01 – At first, we have done some configuration in the layers of CNN, such as adding layers for better accuracy. The first model includes some convolutional 2D layers, a Max-polling 2D layer, a Dense layer, a flattened layer for better accuracy, and Drop-out Layer to overcome the overfitting issue.

Keras Sequential CNN Model 02 with Data Augmentation – In this section, we describe the model analysis with Data augmentations. This model is similar to model 01 with adjusted dropout values and fed the model with augmented input datasets. It

should be noted that for the sequential model, the Data augmentation does not show much improvement in accuracy.

VGG16 Plain Model 03 – The VGG16 model has become very successful in object recognition. This paper has studied the VGG16 pre-trained model, and to get better accuracy for the dataset, a fattened layer was added, and the output layer was flattened to one dimension. The model then employed a fully connected layer with 128 hidden units and ReLU activation to complete the task. A dense layer consisting of dimension 128, followed by Dropout $= 0.6$, was fine-tuned, and finally, we have fine-tuned more with another parameter of the Dense layer with a softmax activation function. For further inspections, we have tested the dataset in the pre-trained model VGG16, which is highly known for state-of-art accuracy in image processing and computer vision. We have excluded the top layer and flattened the output for better accuracy.

VGG16 Finetuned Model 04 – The fine-tuned model is similar to model 3 VGG16, but as we were dealing with some overfitting issues in the plain VGG16 model, we have added a dense layer to combat the problem.

VGG16 with Data Augmentation Model 05 – We have performed data augmentation to explore more about the concept and increase the accuracy of this dataset. Data augmentation has shown a tremendous positive impact on accuracy and model training. We first augmented the data and then trained the VGG16 model to evaluate the accuracy and investigate whether this approach gives us better accuracy. This model provides the highest accuracy we have obtained among all our models.

Inception V3 Plain Model 06 – In the study by Szegedy et al. [16], the image classification model called Inception v3 is broadly used, which obtained more than 78.1% accuracy on the larger datasets. The model includes several concepts that have been developed throughout time by many academics [16].

Convolutions, average pooling, max pooling, and fully linked layers are all included in the model [16]. This model performs batch normalization. The batch normalization is seen throughout the model and applied to activation inputs. Loss is computed using Softmax. For better accuracy, we have excluded the top layer of the model.

Inception Fine-tuned with Data Augmentation Model 07 – In this section, we have explained the configuration for the inception V3 model with data augmentation. As the accuracy of model 06; Inception V3 is not very promising to us, we have augmented the data and figured that it includes fattened output followed by a dropout layer.

4.2 Experimental Result

Table 1 demonstrates the experimental result analysis of the seven fine-tuned and raw models for this project. Plain inception V3 obtains the highest validation accuracy. But this model has shown the tendency of overfitting. The Modified VGG16 with Data Augmentation model also performs well as it has only a 3% difference with the highest accuracy obtained model. It should be noted that Modified VGG16 with Data Augmentation has the lowest validation loss.

Table 1. Result analysis

Model	Training accuracy	Validation accuracy	Training loss	Validation loss
Sequential CNN	87.50%	77.50%	0.2932	0.6759
Sequential CNN with Data Augmentation	70%	75%	0.6160	0.8332
Plain VGG16	100%	85%	6.538×10^{-4}	2.9171
Modified VGG16	85%	87%	0.3902	0.4001
Modified VGG16 with Data Augmentation	**90.99%**	**87.18%**	0.2754	**0.3899**
Plain InceptionV3	**95%**	**90%**	0.2770	0.5379
Modified InceptionV3 with Data Augmentation	88.82%	84.62%	0.6077	1.0332

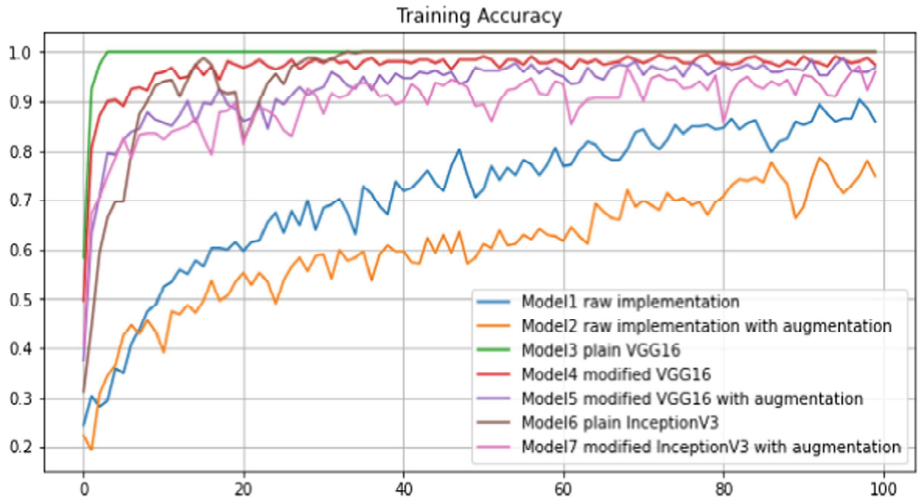

Fig. 3. Training accuracy comparison (epoch vs. accuracy)

In Figs. 3, 4, 5, and 6, we have shown an accuracies comparison of all the tested models with our dataset. The graphs show the difference in the accuracy vs. epoch numbers.

The InceptionV3 model does not perform well with our dataset. Therefore, we excluded the graph from some of the results, which helped us visualize better accuracy results.

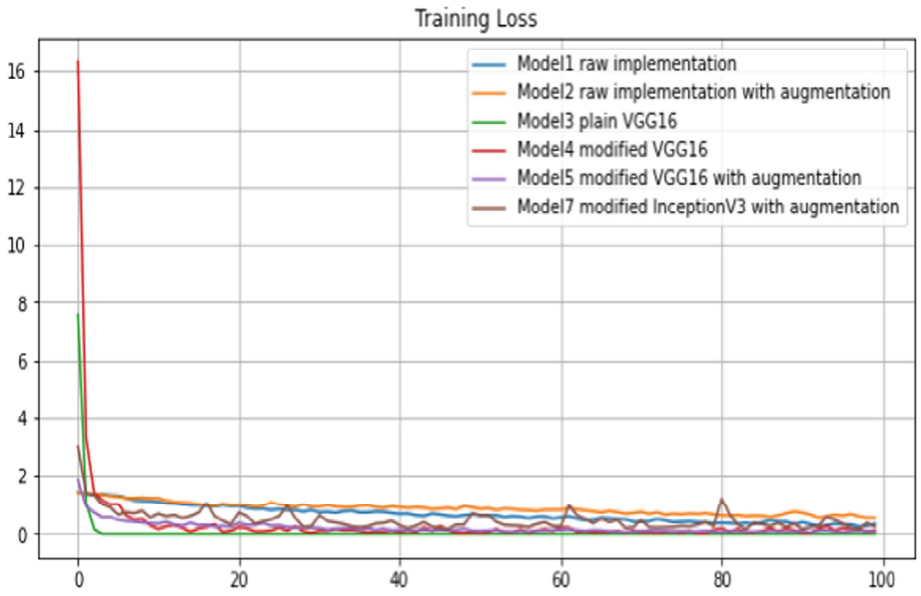

Fig. 4. Training loss comparison (epoch vs. loss)

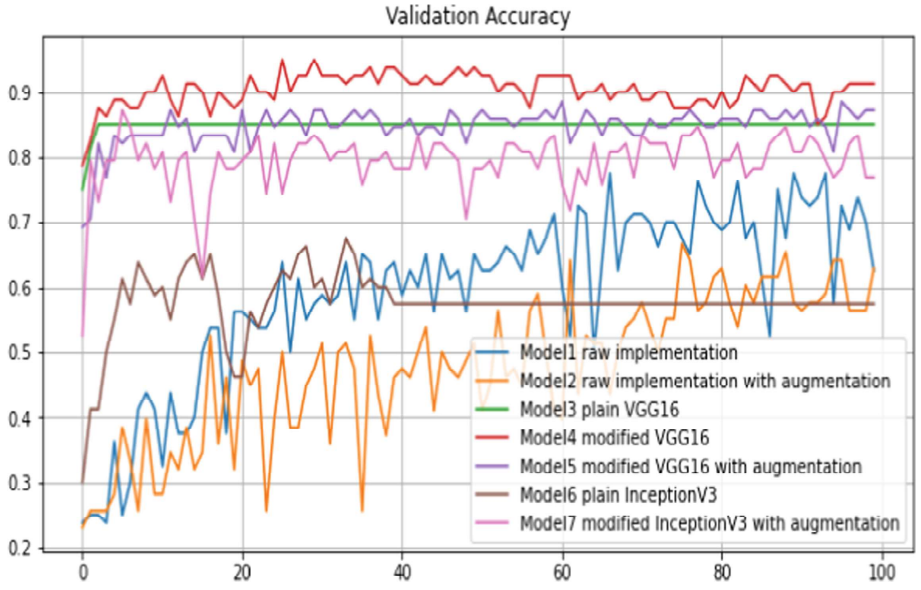

Fig. 5. Validation accuracy comparison (epoch vs. accuracy)

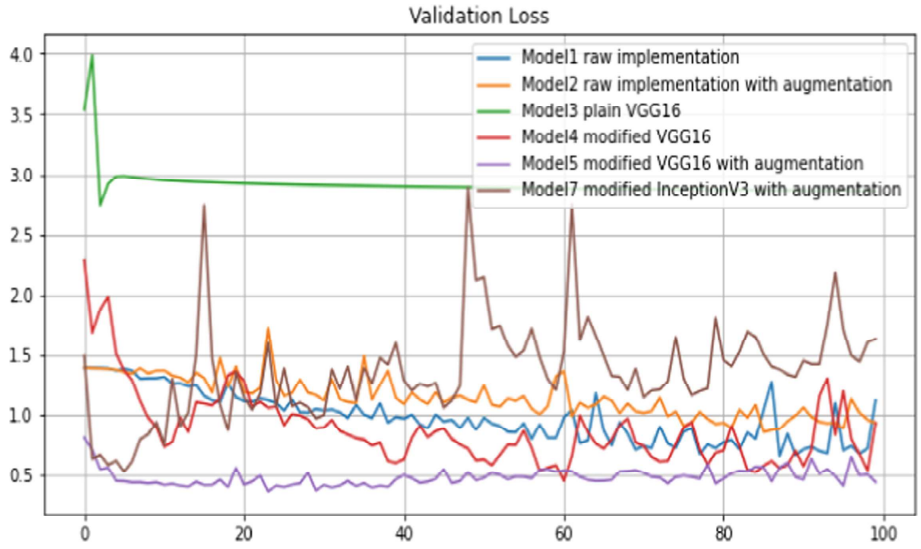

Fig. 6. Validation loss comparison (epoch vs. loss)

4.3 Performance Evaluation

Table 2 shows the testing result of the test images. We have separately taken images not included in training and testing for this project. These images are used to test the model prediction results.

The plain VGG16 model performs better in training. In the case of validation accuracy, we have obtained that the modified VGG16 performs better than all other models; as for the validation loss, the modified VGG16 with Data augmentation performs well. For further inspection, we wanted to analyze the difference in Training and Testing accuracy of all the models; The most negligible difference obtained between models in training and testing can be more stable, modified VGG16 with data augmentation in our case.

Also, the modified VGG16 with data augmentation is better for the loss difference between training and validation. Therefore, we can conclude that the best-performed model is modified VGG16 with data augmentation if we only look at the model in the training time. We tested our models with 40 unseen datasets (10 images from each category) and analyzed the models. The modified VGG16 model gives us the most accurate result for the unseen test dataset based on the unseen 40 images.

The inception V3 result analysis shows that the model is the least accurate. As to our study, this is due to batch normalization. The batch normalization does not perform well in CNN and overfits the data. The batch normalization process works well in RNN. As we are using CNN, the performance is not good. The inception V3 result analysis shows that the model is the least accurate. As to our study, this is due to batch normalization. The batch normalization does not perform well in CNN and overfits the data. The batch normalization process works well in RNN. As we are using CNN, the performance is not good.

Table 2. Performance of the models against the test dataset

Model	Prayer (out of 10)	Concert (out of 10)	Meeting (out of 10)	Convocation (out of 10)	Total correct images (out of 40)	Accuracy (%)
Sequential CNN	3	10	0	9	22	55
Sequential CNN with Data Augmentation	1	2	1	10	14	35
Plain VGG16	5	10	8	8	31	77.5
Modified VGG16	7	10	8	10	35	87.5
Modified VGG16 with Data Augmentation	6	1	9	10	26	65
Plain InceptionV3	1	8	5	7	21	52.5
Modified InceptionV3 with Data Augmentation	0	6	2	0	8	20

5 Conclusion

We experimented with seven different CNN models, including VGG16 and InceptionV3. We have performed data augmentation and resizing of the images. We have added several layers in VGG16 and removed the top layer in the InceptionV3 model to get better accuracy. Throughout research and analysis of the models, we have achieved the highest accuracy with the modified VGG16 model with data augmentation. This research shows that data augmentation and suitable layers in the pre-trained VGG16 model resulted in the highest accuracy. The fine-tuning of VGG16 surpasses the accuracy of Plain VGG16.

References

1. Ramprasath, M., Anand, M.V., Hariharan, S.: Image classification using convolutional neural networks. Int. J. Pure Appl. Math. **119**, 1307–1319 (2018)
2. Ren, X., Guo, H., Li, S., Wang, S., Li, J.: A novel image classification method with CNN-XGBoost model. In: Kraetzer, C., Shi, Y.-Q., Dittmann, J., Kim, H.J. (eds.) Digital Forensics and Watermarking, pp. 378–390. Springer International Publishing, Cham (2017). https://doi.org/10.1007/978-3-319-64185-0_28

3. Kaniz Fatema, M., Ahmed, R., Arefin, M.S.: Developing a system for automatic detection of books. In: Chen, J.I.-Z., João, M.R., Tavares, S., Iliyasu, A.M., Ke-Lin, D. (eds.) Second International Conference on Image Processing and Capsule Networks: ICIPCN 2021, pp. 309–321. Springer International Publishing, Cham (2022). https://doi.org/10.1007/978-3-030-84760-9_27
4. Sun, Y., Xue, B., Zhang, M., Yen, G.G., Lv, J.: Automatically designing cnn architectures using the genetic algorithm for image classification. IEEE Trans. Cybern. **50**, 3840–3854 (2020)
5. Lee, H., Kwon, H.: Going deeper with contextual CNN for hyperspectral image classification. IEEE Trans. Image Process. **26**, 4843–4855 (2017)
6. Vaddi, R., Manoharan, P.: Hyperspectral image classification using CNN with spectral and spatial features integration. Infrared Phys. Technol. **107**, 103–296 (2020)
7. Yang, X., Ye, Y., Li, X., Lau, R., Zhang, X., Huang, X.: Hyperspectral image classification with deep learning models. IEEE Trans. Geosci. Remote Sens. **56**, 5408–5423 (2018)
8. Michael, M., Cord, M., Thome, N.: Max-min convolutional neural networks for image classification. In: 2016 IEEE International Conference on Image Processing (ICIP), pp. 3678–3682 (2016)
9. Li, Q., Cai, W., Wang, X., Zhou, Y., Feng, D.D., Chen, M.: Medical image classification with convolutional neural network. In: 2014 13th International Conference on Control Automation Robotics and Vision (ICARCV), pp. 844–848 (2014)
10. Wang, D., Xu, Q., Xiao, Y., Tang, J., Luo, B.: Multi-scale convolutional capsule network for hyperspectral image classification. In: Lin, Z., et al. (eds.) PRCV 2019. LNCS, vol. 11858, pp. 749–760. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-31723-2_64
11. Krizhevsky, A., Sutskever, I., Ilya, H., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, vol. 25 (2012)
12. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440 (2015)
13. Qian, J., Yang, J., Xu, Y.: Local structure-based image decomposition for feature extraction with applications to face recognition. In: IEEE Transactions on Image Processing, pp. 3591–3603 (2013)
14. Silva, C., Welfer, D., Gioda, F., Francisco, P., Dornelles, C.: Cattle brand recognition using convolutional neural network and support vector machines. IEEE Lat. Am. Trans. **15**, 310–316 (2017)
15. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
16. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2818–2826 (2016)