2022 3rd International Conference for Emerging Technology (INCET)
Belgaum, India. May 27-29, 2022

Detection of Brain Tumor Using CNN and CNN-SVM

Kavya Duvvuri
Dept. of Computer Science Engineering
Amrita School of Engineering
Bengaluru, India
krishnakavya.duvvuri@gmail.com

Harshitha Kanisettypalli
Dept. of Computer Science Engineering
Amrita School of Engineering
Bengaluru, India
harshithakanisettypalli@gmail.com

Sarada Jayan
Dept. of Mathematics
Amrita School of Engineering
Bengaluru, India
j_sarada@blr.amrita.edu

2022 3rd International Conference for Emerging Technology (INCET) | 978-1-6654-9499-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/INCET54531.2022.9824725

***Abstract*— Brain tumor is a type of cancerous growth that may occur in the brain. Early diagnosis of the disease is crucial for proper treatment. Diagnosis of brain tumors is usually done using images obtained through magnetic resonance imaging (MRI). MRI images can be classified using a Convolutional Neural Network (CNN), which is a technique in deep learning. It is suitable for classifying large image datasets. Support Vector Machine (SVM) is a technique in machine learning that is predominantly used for classification and in various regression problems. In this paper, we classified brain MRI images using pre-trained models like AlexNet, VGG16, InceptionV3, and ResNet50. Finally, a CNN model and an SVM model are trained with the same dataset. Using the results thus obtained a hybrid CNN-SVM model has been built to get better accuracy and prediction results.**

Keywords*— Brain Tumor, CNN, VGG16, AlexNet, InceptionV3, ResNet50, SVM, CNN-SVM

I. Introduction

Tumors are disorders that are caused by the rapid growth of tissues and cells in the body. Diseases like brain tumors can be fatal if identified at their later stages. But sometimes the detection of the presence of brain tumors in MRI images may become a difficult and tedious job for physicians. The job can be made easier with the use of deep learning technology. Convolutional Neural Network is an extension of Artificial Neural Networks (ANN) in deep learning. The concept of CNN is extensively used for dealing with image classification problems as it has good feature extraction capability. In this paper, we check the predictions done using pre-trained models and create a custom CNN model to have better predictions. As a final step, we attempt to increase the prediction accuracy by using SVM as the classification layer of the CNN model.

Section 1 of this paper gives an introduction to the work done and gives information about the various references considered. A brief introduction to Convolutional Neural Networks is given in section 2 of the paper, while section 3 discusses the topic of Support Vector Machine. Section 4 gives general information about the different pre-trained models considered, while section 5 puts forth the details about the project and the results obtained. Finally, section 6 gives the conclusion drawn from the entire work done.

In [1] a 19-layer hybrid CNN-SVM model has been developed for classifying brain tumor images into four categories, Pituitary Tumor, Glioma Tumor, Meningioma Tumor, and No Tumor. The final accuracy achieved by this model is 97.1%.

Reference [2] presents a CNN-SVM hybrid model for classifying brain tumor images into two different classes, benign and malignant. Multiple models like, Discrete Wavelet Autoencoder (DWA), K-Nearest Neighbors (KNN), Deep Neural Network (DNN), and plain CNN have been tested with the dataset. But the CNN-SVM model gives better accuracy of 98.5%.

CNN-SVM hybrid model may be used for classifying EEG signals too, as shown in [3]. The method of transfer learning can be used for retraining pre-trained models for smaller datasets. This method has been vividly explained in [4]. Reference [5] puts forth a fuzzy c-means and SVM-based hybrid technique for MRI brain tumor image classification.

The use of CNN in the classification of Diabetic Retinopathy (DR) images into different classes like Mild DR, Moderate DR, Severe DR, Proliferative DR, and No DR has been shown in [6]. Reference [7] gives a complete analysis of SVM for image classification. In [8], we can see the use of a technique called Shallow CNN, which is computationally more effective than plain CNN models, for the detection of Tuberculosis. A probabilistic ensemble-based CNN classifier is used for image classification in [9]. This method helps in reducing the computational time.

Reference [10] provides a literature survey on the architecture of CNN using a pre-trained model, AlexNet, as the base. In [11] we find an interesting analysis of the accuracies of various SVM kernels in the prediction of rainfall in India. In [12] we are introduced to a CNN and CNN-LSTM network-based method that has been used for diabetes detection using heart rate signals. The link for the dataset used has been provided in [13].

II. Convolutional Neural Network

Convolutional Neural Network is an extension of a concept known as an ANN. It is a deep learning technique that is frequently used for image classification. CNN is very effective when the dataset is very large. CNN is said to have two phases. They are the feature extraction phase and the classification phase. The first phase, which is also called the Feature extraction phase, involves the extraction of important features that form the basis of further classification. The classification phase is the decision taking phase where the model assigns a class to every input image.

CNN involves three main types of layers called input layers, output layers, and hidden layers. The first set of layers is the input layers, and they are responsible for taking images from the given datasets, while the hidden layers involve a series of layers such as convolution and pooling. Convolution layers extract important information from images using filters. The spatial dimensions of feature maps obtained from convolution layers are reduced by the pooling layers. Finally, output layers like the fully connected layer classify the given image based on the features extracted in the hidden layers. Fig. 1 gives a visualization of what any general CNN model looks like.

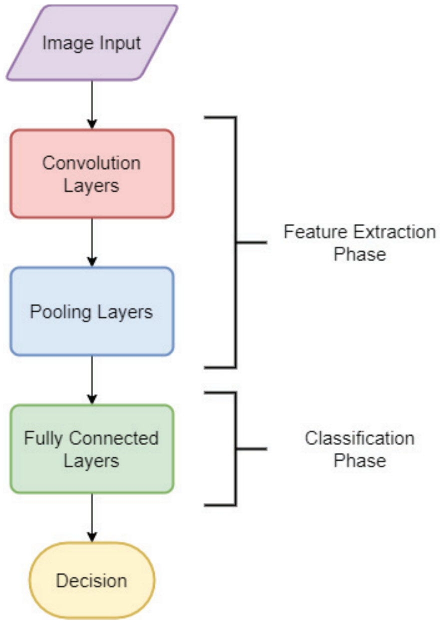

Fig. 1. Phases of CNN

There are pre-trained CNN models which can be used for training our model through the transfer learning approach. In this approach, we can tune our model to suit our dataset and train it. In this paper, we have considered four different pre-trained models called AlexNet, InceptionV3, VGG16, and ResNet50. These models have been trained with the brain tumor dataset and the corresponding accuracies have been compared.

III. Support Vector Machine

Support Vector Machine is a technique in machine learning that can be used for classification as well as in regression problems. In binary classification problems, the algorithm works to insert a line or hyperplane separating data points into two classes. The data points from both the classes which are nearest to the hyperplane are called support vectors. The hyperplane is inserted in such a way that the margin between the support vectors of both the classes is maximized. However, the SVM method does not give good accuracy when used over large datasets. In this project, we used SVM to classify brain tumor images. SVM can also be used in the classification phase of CNN.

IV. Pre-Trained CNN Models

A. *AlexNet*

AlexNet is a pre-trained CNN model that is made up of eight layers. Out of the eight layers present in AlexNet, the initial five layers are convolution layers and the final three are dense layers or fully connected layers that are responsible for classification. AlexNet has the benefit of having a lesser number of layers.

B. *VGG16*

VGG16 is a 16-layer pre-trained CNN model. The 16 layers consist of 13 convolution layers and three dense or fully connected layers. Apart from these 16 layers, the model contains 5 pooling layers. This model has an advantage over the AlexNet model as the AlexNet model has kernels of larger size. VGG16 had kernels of 3×3.

C. *InceptionV3*

InceptionV3 is a pre-trained model that consists of 42 layers. It is said to have better accuracy and efficiency than its previous models. Also, it has low error rates and computational costs than InceptionV1 and InceptionV2.

D. *ResNet50*

ResNet50 is a deep learning CNN model that consists of 50 layers in total. There are 48 convolution layers, one average pooling layer, and one max-pooling layer. The model consists of 25,636,712 parameters and is 98 MB in size. It is said to give more accurate results than other models.

V. Experiments and Results

In this paper, we considered a brain tumor dataset with 3645 brain MRI images. The dataset has been split into train set and test set in the 80-20% ratio. There are 2934 images in the training dataset and 711 images in the testing dataset. The dataset was taken from Kaggle. The brain tumor type that was considered is Meningioma Tumor. Fig. 2 shows the entire dataset at a glance.

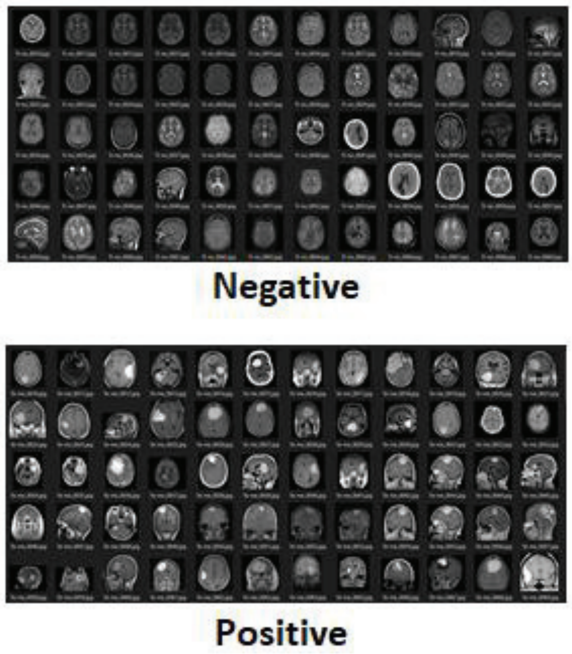

Fig. 2. Dataset at a Glance

The table given below shows the splitting of the data into positive and negative categories.

TABLE I. Dataset Distribution Table

Brain Tumor	Train	Test
Positive	1339	306
Negative	1595	405

All the images in the dataset are in grayscale. The presence of brain tumors can be detected in the images by identifying the presence of whitish formations in the brain. Fig. 3 shows an image that has a brain tumor, shown by the white spot in the grayish background.

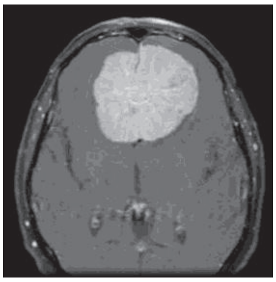

Fig. 3. MRI Image with Brain Tumor

An MRI which does not contain a brain tumor is shown in Fig. 4.

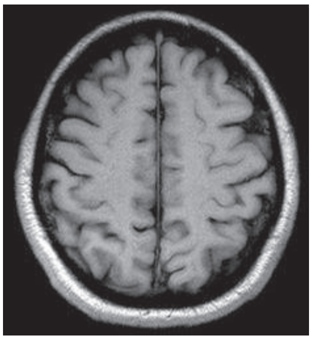

Fig. 4. MRI Image without Brain Tumor

Initially, the images in the dataset were trained with four pre-trained models, AlexNet, VGG16, InceptionV3, and ResNet50. But all the models proved to be overfitting.

TABLE II. ACCURACIES OF PRE-TRAINED MODELS

Model	Accuracy	Validation Accuracy
VGG16	52%	59.96%
AlexNet	86%	65.96%
InceptionV3	98%	43.04%
ResNet50	60%	56.96%

The validation accuracies of the pre-trained models are very less than the accuracies. This shows us that the pre-trained models are overfitting the data. Hence, these models are not suitable for the detection of brain tumors in the given dataset. To deal with this problem we built a custom CNN model in order to properly classify the images.

The plain CNN model presented in this paper consists of nine layers. The nine layers consist of two convolution layers, one dropout layer, three dense or fully connected layers, one flatten layer, and two max-pooling layers.

There are 32 filters and 64 filters in the first convolution layer and second convolutional layer respectively. They have a kernel size of 3×3. A dropout layer of value 0.2 is used between the convolution layers to avoid the overfitting of data. The pool size is 2×2 for the max-pooling layers.

The second pooling layer's output is given as input to the flatten layer to convert the extracted image features from a matrix format to a vector format. These flattened images are passed into dense or fully connected layers.

The first dense layer is made up of 128 neurons while the second and third dense layers consist of 32 and 1 neuron respectively. The last dense layer makes use of the sigmoid activation function. Finally, the classification is done by minimizing the loss using the Adam optimizer. The decision-making depends on a threshold of 0. An image is predicted to be having a brain tumor when the final result after the prediction is lesser than or equal to 0. The prediction shows an image with a prediction result greater than 0 to be having no brain tumor. The architecture of the plain CNN model is shown in the flowchart given in Fig. 5.

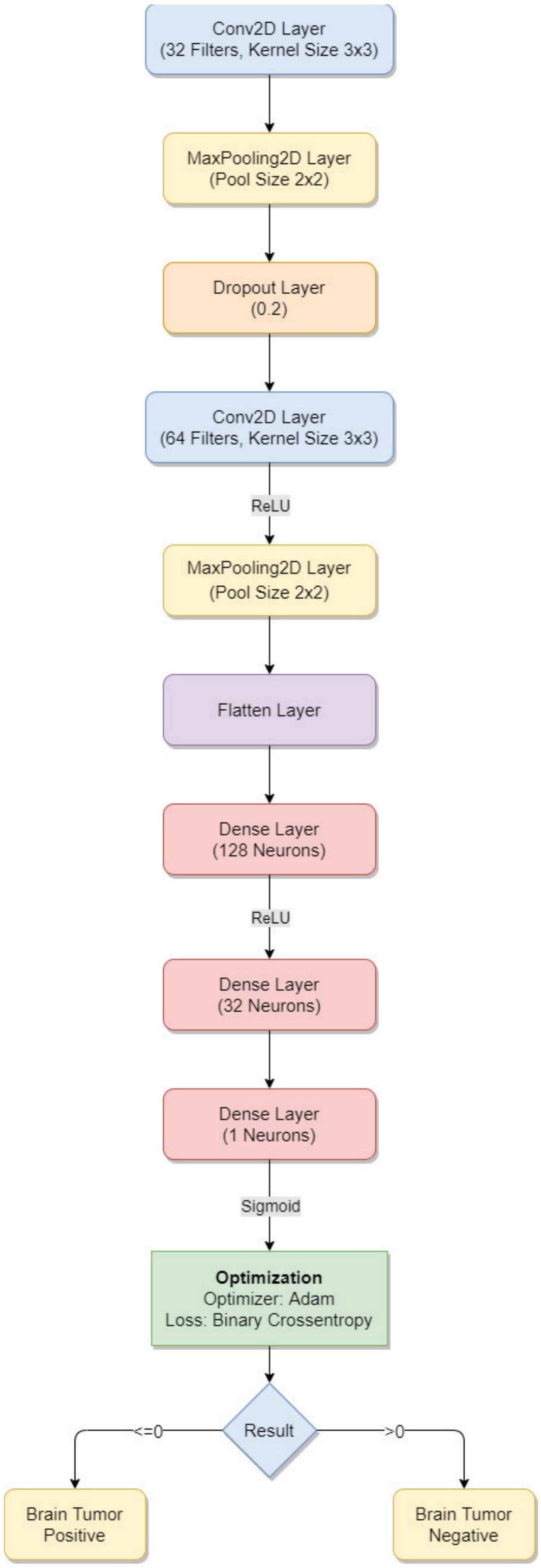

Fig. 5. Flowchart for the CNN Model

The model was trained with the brain tumor dataset in 30 epochs and 10 steps per epoch. The values of loss, validation loss, accuracy, and validation accuracy of the CNN-SVM model are tabulated in the table given below.

TABLE III. EPOCH ACCURACY TABLE FOR CNN MODEL

Epoch Number	Loss	Validation Loss	Accuracy	Validation Accuracy
1	1.5471	0.5277	0.6777	0.7876
2	0.4453	0.5734	0.7640	0.7623
3	0.3618	0.4631	0.8740	0.7665
4	0.3330	0.4565	0.8492	0.7792
5	0.2001	0.3883	0.9100	0.8172
6	0.2197	0.3417	0.9000	0.8453
7	0.2103	0.3180	0.9280	0.8650
8	0.1722	0.4061	0.9480	0.7975
9	0.1592	0.3082	0.9520	0.8748
10	0.1572	0.2692	0.9460	0.9100
11	0.1113	0.2433	0.9520	0.9241
12	0.1377	0.2897	0.9540	0.8875
13	0.1397	0.2441	0.9500	0.9156
14	0.1135	0.2308	0.9560	0.9241
15	0.1391	0.2882	0.9540	0.8847
16	0.1102	0.2233	0.9640	0.9114
17	0.0904	0.2154	0.9720	0.9184
18	0.0681	0.2266	0.9740	0.9156
19	0.0892	0.2209	0.9700	0.9226
20	0.1196	0.1764	0.9560	0.9339
21	0.0573	0.1367	0.9876	0.9494
22	0.1151	0.1649	0.9660	0.9494
23	0.0608	0.1327	0.9800	0.9508
24	0.0503	0.1305	0.9780	0.9592
25	0.0714	0.1136	0.9740	0.9606
26	0.0424	0.1441	0.9840	0.9437
27	0.0337	0.1458	0.9876	0.9606
28	0.0751	0.1329	0.9760	0.9564
29	0.0833	0.1070	0.9800	0.9677
30	0.0681	0.1069	0.9800	0.9662

The change in accuracy and validation accuracy values and their variation with the epoch count is shown in Fig. 6, while Fig. 7 shows the change in loss and validation loss with respect to the epoch count.

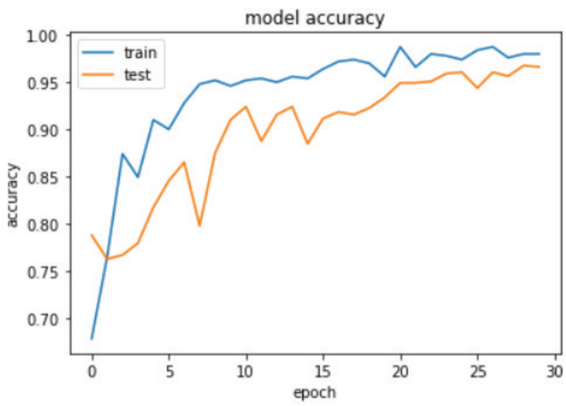

Fig. 6. Accuracy Curve for the Plain CNN Model

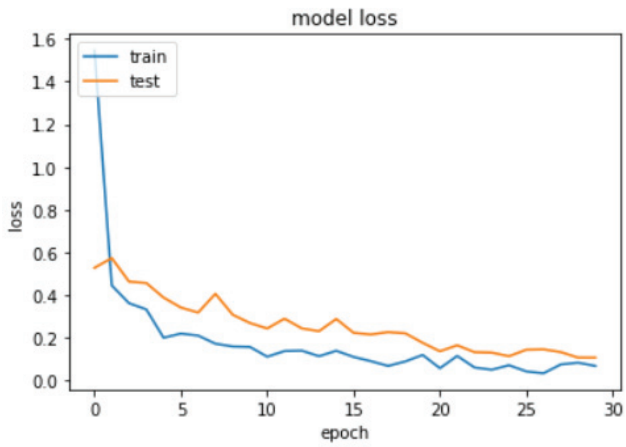

Fig. 7. Loss Curve for the Plain CNN Model

Next, we trained an SVM model to classify the images in the brain tumor dataset. The accuracy value obtained by the SVM model was 93.82. The false positives (FP), true positives (TP), false negatives (FN), and true negatives (TN) are shown in the confusion matrix given in Fig. 8.

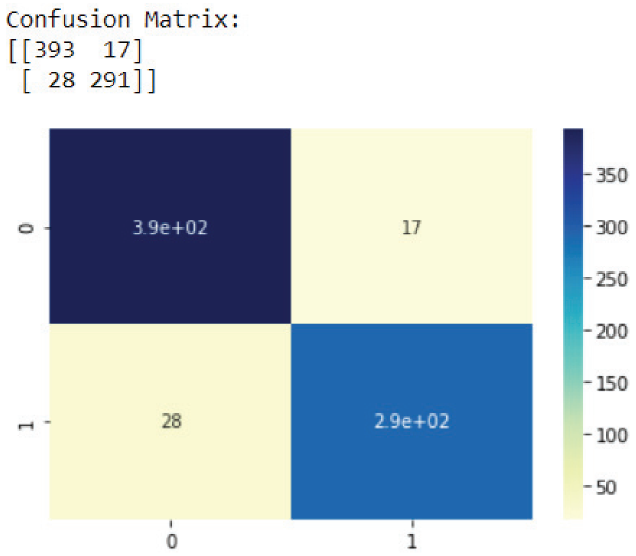

Fig. 8. SVM Model Confusion Matrix

To improve the accuracy of our CNN model, we use SVM in the last classification layer of the initial plain CNN model. The architecture of the CNN-SVM is very similar to the plain CNN model. The last layer is modified to classify the images using SVM.

The optimization is done using Adam optimizer while the loss is considered to be ‘Hinge’. The threshold for the CNN-SVM model was taken as 0.3.

Fig. 9 represents the CNN-SVM model architecture.

The model was trained with the dataset in 30 epochs and 10 steps per epoch. The values of loss, validation loss, accuracy, and validation accuracy, of the CNN-SVM model, are tabulated in table 4.

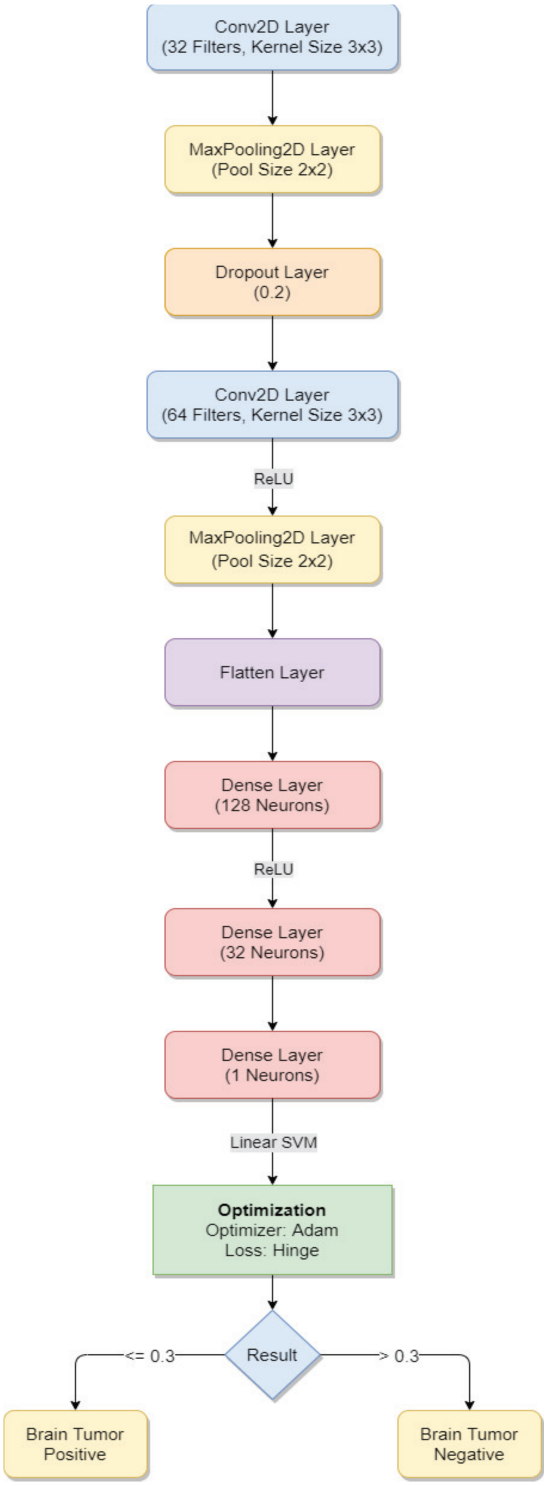

Fig. 9. Flowchart for the CNN-SVM Model

TABLE IV. EPOCH ACCURACY TABLE

Epoch Number	Loss	Validation Loss	Accuracy	Validation Accuracy
1	2.0154	0.8567	0.5760	0.5696
2	0.4999	0.5901	0.7020	0.7722
3	0.3631	0.4361	0.8680	0.8143
4	0.2498	0.4566	0.9080	0.8017
5	0.1798	0.5101	0.9280	0.7932
6	0.1871	0.4124	0.9300	0.8228
7	0.2490	0.4054	0.9040	0.8003
8	0.1905	0.3253	0.9120	0.8383
9	0.2274	0.3589	0.9000	0.8312
10	0.1756	0.3264	0.9380	0.8495
11	0.1421	0.3111	0.9520	0.8340
12	0.1361	0.3731	0.9380	0.8298
13	0.2142	0.3609	0.9020	0.8579
14	0.1876	0.3254	0.9200	0.8917
15	0.1535	0.2733	0.9340	0.8551
16	0.1701	0.2293	0.9300	0.9058
17	0.0971	0.2448	0.9480	0.8959
18	0.1098	0.2146	0.9640	0.9044
19	0.1275	0.1503	0.9566	0.9367
20	0.0911	0.1613	0.9545	0.9269
21	0.0686	0.1700	0.9800	0.9255
22	0.0878	0.1385	0.9680	0.9367
23	0.0587	0.1246	0.9760	0.9522
24	0.0548	0.1118	0.9720	0.9451
25	0.0762	0.1230	0.9780	0.9494
26	0.0514	0.0958	0.9780	0.9691
27	0.0647	0.1745	0.9720	0.8959
28	0.0455	0.0961	0.9793	0.9620
29	0.0602	0.1071	0.9840	0.9423
30	0.0448	0.0950	0.9840	0.9719

The change in accuracy and validation accuracy values in variation with the epoch count is shown in Fig. 10, while Fig. 11 shows the change in validation loss and loss in variation with the epoch count.

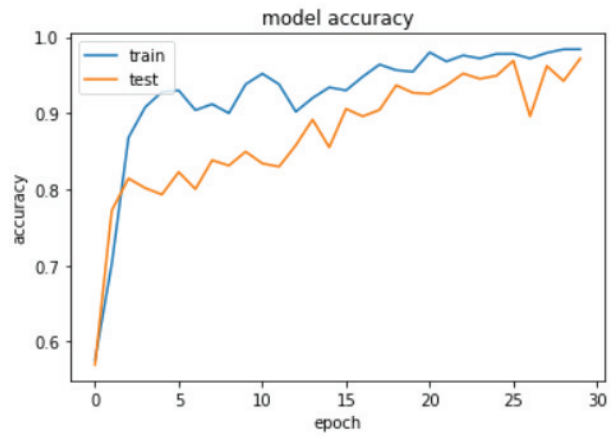

Fig. 10. Accuracy Curve for the CNN-SVM Model

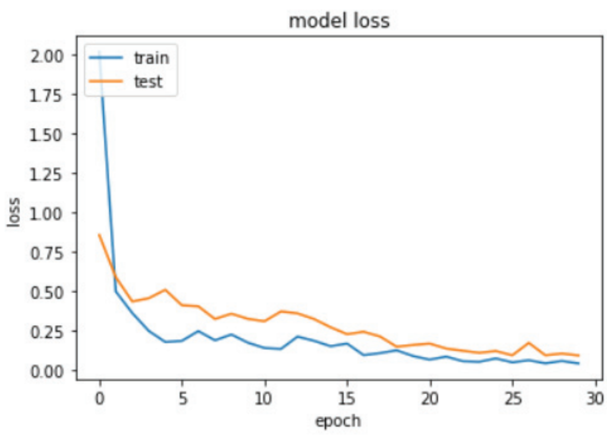

Fig. 11. Loss Curve for the CNN-SVM Model

From table 3 and table 4 we understand that the accuracies of both the models are almost equal. But the validation accuracy of the proposed CNN-SVM hybrid model is greater than the validation accuracy of the CNN model. Also, it can be observed that the values of loss and validation loss are lesser in the case of the CNN-SVM model.

We further try to compare the times taken by both models for making predictions on all the images in the test dataset. Table 5 shows the times taken by both models making predictions for the positive and negative test datasets separately.

TABLE V. TIME TAKEN FOR PREDICTIONS

Test Dataset Given	CNN Model	CNN-SVM Model
Positive	15.2955057	15.3744635
Negative	19.7095533	19.7084334

Table 5 tells us that the times that were taken by the plain CNN model and the CNN-SVM model are almost equal.

To be able to further analyze the models we look into the predictions made by both models. The table given below shows the false positives (FP), true positives (TP), true negatives (TN), and false negatives (FN) predicted by the plain CNN model and the CNN-SVM model. The threshold value was taken to be 0.3 for the CNN-SVM model prediction and 0 for the CNN model prediction.

TABLE VI. PREDICTION TABLE

Prediction Values	CNN Model	CNN-SVM Model
True Positives	228	282
True Negatives	401	379
False Positives	4	26
False Negatives	78	24

The table tells us that the false negatives of the CNN-SVM model are lesser than the false negatives of the plain CNN model. Also, the false positives obtained using the CNN-SVM model are greater than the plain CNN model. However, the total number of false predictions is lesser in the CNN-SVM model when compared with the plain CNN model.

For getting deeper insight into the values obtained, we calculate the values of accuracy using the formula in Eq. 1. The misclassification rate is also calculated for both models, using Eq. 2.

$$\text{Accuracy} = \frac{TP+TN}{Total} \tag{1}$$

The accuracy for the plain CNN model is 88.47 and the CNN-SVM model gives an accuracy of 92.97.

$$\text{Misclassification Rate} = \frac{FP+FN}{Total} \tag{2}$$

The misclassification rate for the plain CNN model is 11.53 and the misclassification rate for the CNN-SVM model is 7.03. This shows us that the CNN-SVM model is better than the CNN model.

VI. CONCLUSION

In this project, we observe that the pre-trained CNN models that have been considered like, AlexNet, InceptionV3, VGG16, and ResNet50 are overfitting the dataset and are not suitable for predicting brain tumors. We then develop a custom CNN model to suit the brain tumor dataset and test the dataset with an SVM model. Finally, we incorporate SVM in the last layer of the existing plain CNN model to develop a CNN-SVM hybrid model. The accuracy of the hybrid CNN-SVM model is 92.96 and the accuracy of the plain CNN model is 88.47. It can be observed that the CNN-SVM model shows a lesser misclassification rate than the CNN model. This shows us that using SVM in the classification layer of a CNN model helps increase its accuracy. However, the accuracy value of the tested SVM model is 93.82 and it seems to be larger than the accuracy of the CNN-SVM hybrid model. But it is suggested to use the CNN-SVM model and not the pure SVM model as SVM does not give good accuracy for larger datasets. However, the accuracy value of the CNN-SVM model would increase if we took a larger dataset. Hence, we can conclude by saying that the hybrid CNN-SVM model shows us better accuracy values than the plain CNN model and it is best suited for the brain tumor dataset. In this project, only binary classification of brain tumor images has been taken up. The model can be further extended for multiclassification. This will help in identifying the type of cancer.

REFERENCES

[1] Sejuti, Zarin Anjuman, and Md Saiful Islam, "An Efficient Method to Classify Brain Tumor using CNN and SVM", In 2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), IEEE, pp. 644-648. 2021.

[2] Khairandish, Mohammad Omid, Meenakshi Sharma, Vishal Jain, Jyotir Moy Chatterjee, and N. Z. Jhanjhi, "A Hybrid CNN-SVM Threshold Segmentation Approach for Tumor Detection and Classification of MRI Brain Images", IRBM, 2021.

[3] Saidi, Afef, Slim Ben Othman, and Slim Ben Saoud, "A novel epileptic seizure detection system using scalp EEG signals based on hybrid CNN-SVM classifier", 2021 IEEE Symposium on Industrial Electronics & Applications (ISIEA), IEEE, pp. 1-6, 2021.

[4] Hussain, Mahbub, Jordan J. Bird, and Diego R. Faria, "A study on CNN transfer learning for image classification", In UK Workshop on Computational Intelligence, Springer, pp. 191-202, 2018.

[5] Singh, Amritpal, "Detection of brain tumor in MRI images, using combination of fuzzy c-means and SVM", In 2015 2nd international conference on signal processing and integrated networks (SPIN), IEEE, pp. 98-102, 2015.

[6] Nikhil M. N., Angel Rose A., "Diabetic Retinopathy Stage Classification using CNN", International Research Journal of Engineering and Technology (IRJET), vol. 6, issue. 5, pp. 5969-5974, 2019.

[7] Sai Surya Teja Gontumukkala, Yogeshwara Sai Varun Godavarthi, Bhanu Rama Ravi Teja Gonugunta, Subramani R, K Murali, *"Analysis of Image Classification using SVM"*, 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), pp. 01-06, IEEE, 2021.

[8] Simon, Anson, R. Vinayakumar, V. Sowmya, and K. P. Soman, "Shallow CNN with LSTM Layer for Tuberculosis Detection in Microscopic Images", 2017.

[9] Neena Aloysius, and M. Geetha, "Image classification using an ensemble-based deep CNN", In Recent Findings in Intelligent Computing Techniques, Springer, pp. 445-456, 2018.

[10] Neena Aloysius, and M. Geetha, *"A review on deep convolutional neural networks"*, In 2017 International Conference on Communication and Signal Processing (ICCSP), IEEE, pp. 0588-0592, 2017.

[11] Kumar, M. Kiran, J. Divya Udayan, and A. Ghananand, *"Efficiency of Different SVM Kernels in Predicting Rainfall in India"*, In

International Conference on Information Management & Machine Intelligence, Springer, pp. 169-175, 2019.

[12] Swapna, Goutham, Soman K P, and Ravi Vinayakumar, "Automated detection of diabetes using CNN and CNN-LSTM network and heart rate signals", Procedia computer science 132, pp. 1253-1262, 2018.

[13] Kaggle Brain Tumor Dataset https://www.kaggle.com/masoudnickparvar/brain-tumor-mri-dataset