Estimation of Automatic Video Captioning in Real Applications using Machine Learning Techniques and Convolutional Neural Network

Vaishnavi J[1]†and Narmatha V[2]††,

Department of Computer and Information Science, Annamalai University, Annamalai nagar, India

Summary

The prompt development in the field of video is the outbreak of online services which replaces the television media within a shorter period in gaining popularity. The online videos are encouraged more in use due to the captions displayed along with the scenes for better understandability. Not only entertainment media but other marketing companies and organizations are utilizing videos along with captions for their product promotions. The need for captions is enabled for its usage in many ways for hearing impaired and non-native people. Research is continued in an automatic display of the appropriate messages for the videos uploaded in shows, movies, educational videos, online classes, websites, etc. This paper focuses on two concerns namely the first part dealing with the machine learning method for preprocessing the videos into frames and resizing, the resized frames are classified into multiple actions after feature extraction. For the feature extraction statistical method, GLCM and Hu moments are used. The second part deals with the deep learning method where the CNN architecture is used to acquire the results. Finally both the results are compared to find the best accuracy where CNN proves to give top accuracy of 96.10% in classification.

Keywords:

Online videos, hearing impaired, machine learning, feature extraction, GLCM, Hu moments, CNN architecture.

1. Introduction

The long journey of video captioning started very earlier during 1800, the makeover of movies without audio used inter-titles to help the audience understand the flow of the story. By 1920, the movies were created with sound but to help deaf people and the hearing impaired public, captioned films were produced. Captions slowly spread from entertainment media to the educational system and soon television media for news channels started to use captions for breaking news. Now development in artificial intelligence paves the way for automatic captioning which saves the producers time and money in writing transcripts. Summing up, captions in video frame gives more precise facts about the video scenes and can be used for multiple purposes like scanning, knowledge gaining, and recovering of latest reports. [1].

But producing captions is not a much easier task while playing the video. Captions added in the video can gain more audience that enables more viewers to watch either online videos or blogs or any entertainment channels. Nowadays many video captioning methods can solve the problems where a huge amount of messages can be generated. [2]. One of the major issues in video captioning is the inability to capture all the frames in a single video and describe all the actions as captions. So there should be standard methods for solving the above problem using machine learning methods and neural network embedded techniques. Some of the video frames used in the media as samples are shown in Fig.1.

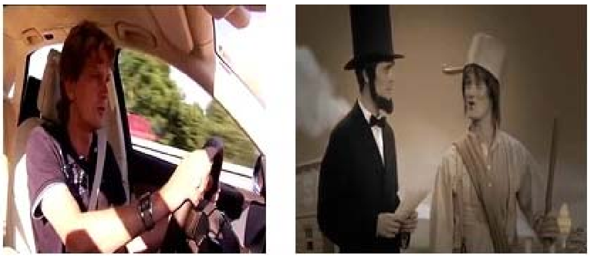


Fig.1 Sample Frames for captioning.

Captions can be generated for the above frames in the video like a person driving the car and two persons chatting. This captioning can be generated automatically using current techniques like a machine and deep learning methods.

In the recent paper [3] text detection algorithm is discussed which can generate short messages in video frames, images, etc. Machine learning contributed a lot to generating automatic messages during videos or short films. The videos are first preprocessed and features can be extracted from any existing methods like statistical[9], or empirical methods. Using the extracted features the images are classified using standard algorithms. Training and testing are the two phases used for classification purposes. Machine learning evolved into deep learning architectures where CNN [35],[36], models are used for video caption [37]generation. The images are first preprocessed and classified with feature extraction using pre-trained models for producing captions during the relay of videos or other media. This paper compares the classification results of the two methods discussed later.

Manuscript received September 5, 2022
Manuscript revised September 20, 2022
https://doi.org/10.22937/IJCSNS.2022.22.9.42

2. Related works

The development made in the field of artificial intelligence is proof of automatic caption generation for short films, videos, and images. Researchers are still rendering hard work for the correct message display along with the scenes or frames in the video. Various contributions of the authors are briefly described in this part.

According to the latest work [4],[10], algorithms are proposed to identify captions from videos like news channels or educational media. This algorithm also helps in decreasing the frames to be processed by first generating captions [14],[15],[16] in the beginning frame so that the candidate region for the caption [25],[26],[28] is constructed [11],[12][17]. The features extracted are wavelet structures that are fed to the classifier [19], for identification of the text messages [18],[27],[29]. Another text detector [5] proposed is based on the fundamental structure of the text. Density-based techniques are employed to remove noise to produce true positive results. Time alignment between text and video sequences [20],[21],[24], is discussed in the recent paper [6] where the segment [38],[39],[40] is experimented with using pattern matching [33],[34] with the messages to be displayed for the particular scene. The time interval for the correct and expected captions [30],[31] are measured and adjusted in this work.

The growth of video captioning automatically [7] using markup language and VCML player is discussed briefly in this paper. This greatly reduces the cost and burden of producing suitable messages with new properties like auditory symbols and tree-structured player files. New methods are adopted to order voice intervals and the messages so that sound can be aligned [22] properly with the text. The next step is the video encoding method [8],[13] for automatic captioning. This method produces a macro-block level map to produce information for every frame in the video and a further rate allotment outline is used to automatically calculate the value of a parameter of each macroblock [23],[32].

3. Contribution and outline of the work

This paper deals with automatic video captions to be generated in parallel with the scenes in the videos. For this purpose, first, the videos are converted into frames and resized into the standard size of 200 x 200. This is called pre-processing stage and after the frames are converted, features are extracted from the frames using three standard methods called Hu moments, GLCM, and statistical measures. These features extracted are then used as input for classification. For classifying, the Random Forest method is used and finally, the results are obtained. The results are compared with another proposed model where the frames are processed using deep learning models. CNN architecture is used for classification and the results are obtained. The results from both methods are compared to find which method produces the best accuracy.

The proposed work is organized as follows. The first section deals with preprocessing of the images and feature extraction with classification is done in the next sector. The following section deals with deep learning classification and the succeeding section deals with the experimental analysis and result. The last section contains the conclusion and references.

4. Proposed approach

The proposed work contains various steps. The primary step is to acquire the input videos from the data set.Pre-processing of images takes place. All the phases are clearly explained in the block diagram given below in Fig.2.

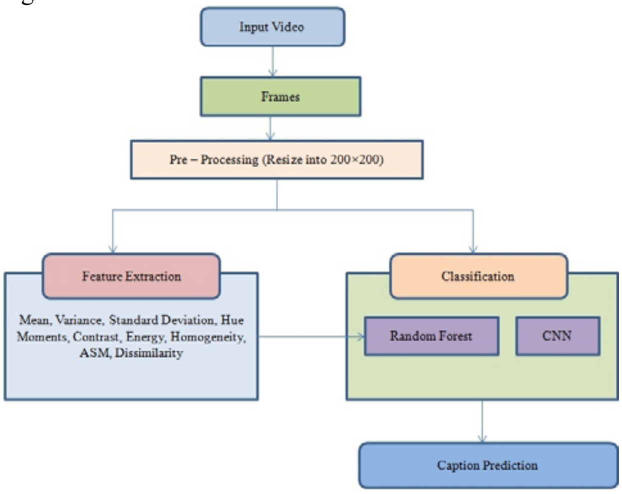

Input Video
Frames
Pre – Processing (Resize into 200×200)
Feature Extraction
Mean, Variance, Standard Deviation, Hue Moments, Contrast, Energy, Homogeneity, ASM, Dissimilarity
Classification
Random Forest
CNN
Caption Prediction

Fig.2 The architecture of the proposed method.

As the block diagram depicts, two parts are explained in this paper.

- The first part deals with the machine learning method where the video frames are preprocessed then features extracted and finally classified.

- The second part deals with the deep learning method where the video frames after preprocessing are classified using CNN architecture.

The results obtained from both the sections are compared to produce the best accuracy for the automatic generation of captions.

Image Acquisition

The dataset containing videos and suitable messages used for this work is retrieved from MSVD (Microsoft Research Video Description Corpus). The data set contains nearly 120K sentences used for short films, videos, and entertainment media. Each scene is described by a single sentence for each frame. The data set contains 2000 video extracts and solo sentences representing each frame. From the dataset, the videos needed for this work are recovered.

4.1 Machine Learning Method

Preprocessing of Images

The videos selected undergo preprocessing stage where there are two steps to make the videos ready for handling. They are

- The videos are converted into frames
- The frames are resized into a standard size.

The videos selected from the dataset are converted into frames using the Stop Motion Effect method which is a distinctive procedure that introduces minute gaps after each frame in the video to create action. The following are the steps for conversion. They are

1. Choose the file as input stored in the memory.
2. Change the settings for conversion of frames by selecting the speed and clip rate. Speed can be slow, medium, or fast and the clip rate differs between 0.2 and 1.5 seconds.
3. Next, select the output format. Once all are ready, operate Stop Motion and save the result.

After the frames have been converted they may be of various sizes which cannot be used for further experiments. So the frames are rescaled into standard size usually 200 x 200 using the pixel relation method called the inter-area method. Resizing is done by determining the values of the pixels in the new image from the old image. This method shrinks the images or frames to the standard size by calculating the values of the neighboring pixels both length and breadth-wise so that the original picture size is the multiple of the new image. Thus the images are preprocessed using the above steps. The preprocessed images are treated further to extract features used for classification.

Feature Extraction

Feature extraction is the dimension reduction technique used to select the essential number of features from the vast data set that represents the entire set of images used for classification. The finest features are selected using the basic techniques to classify the given data set into the given number of classes. The feature extraction methods used are Statistical methods which include Mean, Variance, and Standard deviation. The other two methods are Hu moments and the GLCM method which include contrast, correlation, energy, homogeneity, ASM, and dissimilarity. The above features are global features that describe the entire image used for classification.

Statistical features

The first-order statistical features called descriptive statistics like mean, variance, and standard deviation are extracted as features from the given images.

Mean is used for measuring the central tendency which is the middle point of the image which gives the brightness of the image. It shows the dissimilarity of one image with other images. The identical features can be calculated from the images which are used for classification. The formula is,

$$\text{Mean } \mu = \sum_{i=1}^{n} \frac{x_i}{n} \qquad (1)$$

In Equation (1), X represents the pixel values and n is the count of pixels.

Variance is defined as the dissimilarity from the mean value and calculated by considering the discrepancy between each pixel point and the mean value then squaring the differences and finally dividing the square sums by the data points. Variance measures all deviations from the mean in all directions. The formula is

$$\text{Variance } S_2 = \sum_{i=1}^{n} (x^i - \mu)\frac{2}{n} \qquad (2)$$

Standard Deviation is the Square root of variance. It is used to find the dispersion within the local region that is the edges of the image. SD is a measure of the variance of the data points in the image with the mean. If the value is big, there is more variation and vice versa. The formula is

$$\text{Standard Deviation } s = \sqrt{S_2} \qquad (3)$$

All the three values namely Mean, Variance, and Standard deviation are calculated for sample frames and given in Table 1.

Table 1: Three statistical measures

Frames	*Mean*	*Variance*	*Standard deviation*
1	1.15	2.49	4.99
2	6.47	1.84	4.28
3	1.41	3.82	6.18
4	3.00	1.04	1.02
5	9.74	1.88	4.34

Hu Moments

Hu moments or invariants are called the shape descriptors containing a set of 7 numbers namely H1 to H7 that are similar to image translation, scale, and rotation. A moment is defined as a number that describes the image properties like edges, lines, curves, and regions. These characters are used to differentiate the dissimilar images while show similarities in the same images. In the handling of images, moments are used to find the average of the intensities of image pixels. Hu created some descriptors that are scale-invariant from the geometric moments called raw moments. From raw moments, central moments are calculated and these values are used to calculate the 7 moments based on scale invariance. The key points are the seven values calculated due to rotational invariance. The value gives the location and position of the shapes in the images. The basic calculations consist of translation, scale, and rotational invariance values. They are

Translation invariance

$$M_{pq} = \sum_{x=0}^{n} \sum_{y=0}^{n} (x-\bar{x})^p (y-\bar{y})^q I(x,y) \quad (4)$$

In Equation (4), I(x,y) give basic 2d geometric moments of order (p+q) of the image, xP, yQ give the basics of the moments, and p and q are weights of horizontal and vertical dimensions.

Scale Invariance

$$N_{pq} = \frac{M_{pq}}{M_{00}^{\frac{1+p+q}{2}}} \quad (5)$$

In Equation (5), M00 denotes the total mass of the image. Translation and scale invariants are simple and can be calculated using the above single formula but the difficulty is finding the rotational invariants and Hu defined the seven rotational invariants which are calculated from the above two formulas. They are,

$$h_1 = \eta_{20} + \eta_{02} \quad (6)$$

$$h_2 = (\eta_{20} - \eta_{02})^2 + 4(\eta_{11})^2 \quad (7)$$

$$h_3 = (\eta_{30} - 3\eta_{12})^2 + 3(\eta_{03} - 3\eta_{21})^2 \quad (8)$$

$$h_4 = (\eta_{30} - \eta_{12})^2 + (\eta_{03} - \eta_{21})^2 \quad (9)$$

$$h_5 = (\eta_{30} - \eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{03} + \eta_{21})^2] + (3\eta_{21} - \eta_{03})(\eta_{03} + \eta_{21})[3(\eta_{30} + \eta_{12})^2 - (\eta_{03} + \eta_{21})^2] \quad (10)$$

$$h_6 = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - 7(\eta_{03} + \eta_{21})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{03} + \eta_{21}) \quad (11)$$

$$h_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{03} + \eta_{21})^2] + (\eta_{30} - 3\eta_{12})(\eta_{03} + \eta_{21} + \eta_{21})[3(\eta_{30} + \eta_{12})^2 - (\eta_{03} + \eta_{21})^2] \quad (12)$$

From Equation (6) to (12), the values are calculated and seven features are derived from the images. These features are used as input to the classifier for classification. The seven values are calculated for sample frames and displayed as Table 2 below.

Table 2: Sample Hu Moments

Frames	***H1***	***H2***	***H3***	***H4***	***H5***	***H6***	***H7***
1	1.39	3.68	2.39	5.26	6.99	-8.45	1.73
2	2.25	1.14	1.35	7.23	-1.85	-2.41	6.92
3	1.20	1.18	1.80	1.51	-2.39	6.29	2.49
4	5.12	1.16	6.04	1.58	-8.73	1.70	1.54
5	1.69	6.07	1.18	3.31	-1.15	-4.94	-1.73

GLCM features

Gray level Co-Occurrence Matrix (GLCM) also called spatial dependence matrix is the statistical method used for feature extraction from the image. This method is used to survey texture properties with the spatial association between pixels in the image. The calculation is based on the frequent occurrence of a group of pixels with precise values forming the matrix format to produce statistical measures. Using these calculations matrix is composed of various combinations of the gray level (intensity of the pixel) to provide the measure of variation in intensity levels.

The second-order statistical measures can be calculated using the GLCM method [9] through a matrix with rows and columns equal to the intensity values of the pixels in the image. Out of 13 features, six important features like contrast, correlation, energy, homogeneity, dissimilarity, and ASM (Angular Second Moment) are calculated from the GLCM matrix and the formulas are given as follows.

Contrast- Gives the intensity dissimilarity measures between adjacent pixels in the image. The formula is,

$$\sum_{i,j=0}^{N-1} P_{ij}(i-j)^2 \quad (13)$$

In Equation (13), i and j denotes the pixel positions in the image

Energy- Provides the sum of squared elements in this method. Equation (14) is used to calculate Energy.

$$\sum_{i,j=0}^{N-1} P^2(i,j) \quad (14)$$

Correlation- Reflects the similarity occurrence of the particular group of pixels and the relationship between two or more variables. The formula is,

$$\sum \frac{(\bar{x}-x)(\bar{y}-y)}{\sqrt{(\bar{x}-x)^2 \quad (\bar{y}-y)^2}} \quad (15)$$

Homogeneity- Returns the measurement value of the adjacency of the pixel positions in the image. The formula is,

$$\sum_{i=1}^{n} \frac{(O_i - E_i)^2}{E_i} \tag{16}$$

In Equation (16), Oi denotes the observed value and Ei is the expected value.

ASM- Angular Second Moment is the measure of homogeneity of the image. ASM gives the regularity of grey level distribution in the image. The formula is,

$$\sum_i \sum_j P(i,j)^2 \tag{17}$$

Dissimilarity- Measures the distance between pairs of pixels in the image. The formula is,

$$\sum_{i=1}^{n} \frac{|X_{1(i)} - X_{2(I)}|}{n} \tag{18}$$

Using the above equation, the values are calculated for sample images and the results are compared. The calculated values for samples are given in Table 3 below. GLCM values represent the texture features that are extracted using the equations discussed above. The values are tabulated for easy comparison of these values and the vital features are only taken for the classification method. The six features are considered for classifying the images into prescribed classes.

Table 3: Sample GLCM values

Frames	*Contrast*	*Corre*	*Energy*	*Homog*	*ASM*	*Dissimilarity*
1	3.80	9.25	3.32	8.67	1.10	2.83
2	5.61	8.44	4.50	8.89	2.03	2.73
3	1.81	9.73	3.97	9.33	1.58	1.41
4	8.53	8.20	7.48	9.59	5.59	8.25
5	4.68	8.76	3.55	8.41	1.26	3.41

All three methods in feature extraction are discussed in detail and the features extracted are given as input to the classifier. The number of features extracted per image is given in Table 4 below. There are 16 features extracted from the single image.

Table 4: Feature Extraction Details

S. No	*Method*	*Number of Features*	*Feature Details*
1	Statistical Method	3	Brightness, edges
2	Hu Moments	7	Shape
3	GLCM Method	6	Texture

4.1.2 Classification Method

After the successful extraction of features using the methods mentioned above the next step is the classification of images into different classes. Each class represents unique actions and based upon the class the image belongs to, the caption is generated. Classification of images is mainly based on the accurate features taken from the feature extraction methods. The classification method utilized here is the Random Forest method which is discussed in the next section.

Random Forest

Random Forest is the classification method from supervised learning in which many decision trees are built called forests. [10]. The forest contains trees where the interior node represents the trial attribute, each outlet gives the output of the tree, and the leaf node gives the result. The sixteen features calculated are taken as the root node and internal node. With that tree is constructed and the features are used for testing to decide the class of the input image. The majority result of the trees is considered the final result. This is called the voting method used as basic in the RF method.

The bootstrap aggregation method is utilized in the Random Forest technique where the input image is classified into 50 classes. The number of trees constructed is 200 and each image is sampled with the decision trees with the replacement of features so producing different results. The final decision is based on the majority result.

4.2 Deep Learning Method

Another classification method used for classifying images is the deep learning method based on Convolutional Neural Network architecture. Based on deep learning models and data set collections, constructing novel techniques for caption generator automatically is a trouble-free task. CNN is the subcategory of deep learning networks used especially for image identification and classification. CNN analyzes the images to extract features and uses these features for the organization. This is similar to the human brain where the eyes and brain coordinate to capture the structures in the image through the sight which is transferred to the brain for detecting the object in the scene. This method is repeated through numbers in computers as features to classify the image for caption generation.

CNN architecture is built with many layers of neurons where the order is input, hidden, and output layers. The foremost hidden layer acquires basic features like lines, and splines and the next layer learns objects, and text, and the other layers extract similar features used for classifying. So the layers are arranged in the correct order and the basic building blocks of CNN architecture are four layers with one activation function. They are,

1. Convolution Layer
2. Pooling Layer
3. Activation function
4. Flattening
5. Fully Connected layer

The layers are explained in brief to understand the architecture of the proposed model

Convolution layer

The top layer in the architecture that accepts the input video frame is the convolution layer where the input image can be detected by the features already stored, with that output image is directed to the next layer. Convolution is the mathematical operation on two functions like f and g that alters one function with another, producing a new function. The definition is given as,

$$(f * g)_t = \int_{-\infty}^{\infty} f(T_g)(t - T)_{dT} \quad (19)$$

The input image is converted into matrix format and the filter or kernel is another matrix in odd numbers like 3x3, and 5x5 to convolve over the input image to produce a feature map with the same size as the filter. Each feature maps produce features to identify the objects in the image. So this block is used for the feature extraction process avoiding irrelevant structures. Filters renovate the number of pixels into activation maps.

Pooling layer

The layer implanted between convolution layers is the pooling layer to reduce the size of the image thereby reducing the parameters and avoiding unwanted features. This layer is used to identify edges, and small features like ears, and nose by using several filters. The size of the filter is usually 2x2 where the image size is reduced by half of its dimension so the overfitting problem is solved. Usually, the Max pooling method is used for reduction purposes. Other important factors needed are

- Filter size describes the size of filters like 2x2 or 3x.3
- Stride provides the filter while navigating, the number of positions it leaps in the matrix.
- Padding gives the border effect to increase the size of the image.

Activation function

It is the nonlinear transformation triggering on the input layer before it proceeds to the next layer. The transfer function takes the input signal and changes it into positive form if the signal has negative values before passing it to the succeeding layer. So this function transfers relevant data to the next layer. The activation function utilized here is the Relu function, a linear function that converts the input values directly to positive values otherwise generates zero.The function is defined by,

$$F(X) = X = max(0, X) \quad (20)$$

In Equation (20), x is the input to a function.

Flattening

The next layer to pooling is the flattening method which converts the matrix into single column values, by considering the values row-wise and storing the values in the single column. After conversion, the values are given to the artificial neural network for classification. Flattening changes the values in the matrix to single dimensional array and passes to the next layer to make a long feature vector. So all the pixel values are stored in a particular line and delivered to the last layer.

Fully Connected layer

The pixel values are considered as individual neurons for further classification in the preceding layer. The layer contains the number of neurons as the class divisions to be detected. The final layer contains the classification result of the labels for the problem and assigns it to the data set. Softmax activation is used in the last layer which is best suited for real-life applications. The other function used is dropout to reduce the parameters of the image to avoid redundant features.

Modified CNN Architecture

The Convolutional Neural Network architecture is constructed for the proposed method to classify the given images into prescribed 50 classes. The basic steps used for this method are,

- First, download the data set either online or from the repository, and then preprocess the image

- Then prepare the data for classification by dividing the data into training and testing phases. For this model, 80 percent of the data is selected for training and the remaining 20 percent is used for testing.

- Then construct the CNN architecture using the fundamental design to restore the basic model while freezing the bottom layer to be suitable for the current problem.

- The model is constructed with building blocks like a convolutional layer and a pooling layer.

- Finally, the Global max-pooling layer is used to classify the given image into any of the fifty classes.

- The whole network is trained with 20 epochs. After the 20th epoch, there is no improvement in the network.

The block diagram for the proposed CNN model is given in Fig.3 below.

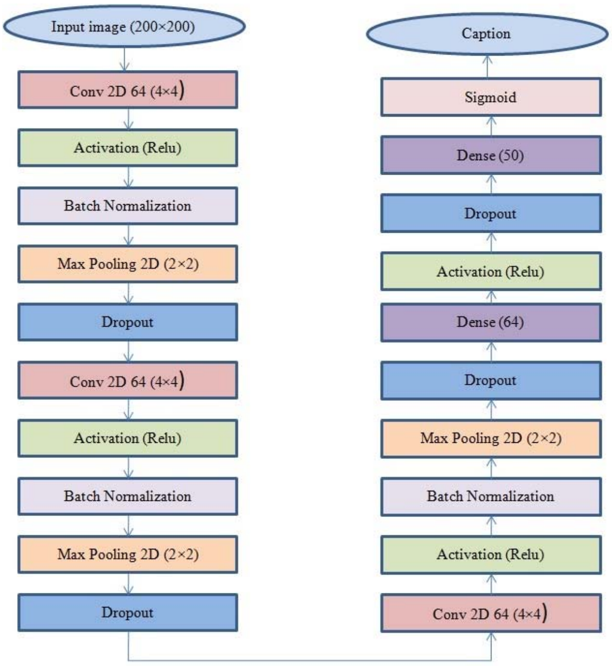


Fig.3 Block diagram of CNN model.

The CNN model is constructed with 3 convolutional layers and 3 max-pooling layers to extract the features from the given image. The input image is of size 200 x 200 and the activation function used is the Relu function. Batch normalization is used to normalize the values and the dropout factor is used to avoid irrelevant features. The global max pooling function is used to get compressed values from the input layers to the dense layer which acts as a fully connected layer. Two dense layers convert the features into attributes of the class the input image belongs to. The final layer is the classifier which is used to classify the images based on the features extracted into 50 different classes. The design is briefly explained in the architecture of the CNN model. Table 5. shows the architecture of the CNN model.

Table 5: Architecture of the CNN model

S. No	*Layer*	*Output size*	*Parameter*
1	conv2d_7 (Conv2D)	(None, 197, 197, 64)	3136
2	batch_normalization_6 (Batch Normalization)	(None, 197, 197, 64)	256
3	max_pooling2d_6 (MaxPooling2D (2x2))	(None, 98, 98, 64)	0
4	dropout_8 (Dropout)	(None, 98, 98, 64)	0
5	conv2d_8 (Conv2D)	(None, 95, 95, 64)	65600
6	batch_normalization_7 (Batch Normalization)	(None, 95, 95, 64)	256
7	max_pooling2d_7 (MaxPooling2D (2x2))	(None, 47, 47, 64)	0
8	dropout_9 (Dropout)	(None, 47, 47, 64)	0
9	conv2d_9 (Conv2D)	(None, 44, 44, 64)	65600
10	batch_normalization_8 (Batch Normalization)	(None, 44, 44, 64)	256
11	max_pooling2d_8 (MaxPooling2D (2x2))	(None, 22, 22, 64)	0
12	dropout_10 (Dropout)	(None, 22, 22, 64)	0
13	global_max_pooling2d_2 (Global pooling)	(None, 64)	0
14	dense_4 (Dense)	(None, 64)	4160
15	dropout_11 (Dropout)	(None, 64)	0
16	dense_5 (Dense)	(None, 50)	3250
Total Parameters		142,514	
Trainable		142,130	
Non Trainable		384	

The architecture of the CNN model as explained above. The total parameters calculated are given as 1, 42,514 where the trainable parameters are 1, 42,130 and the remaining 384 are declared as non-trainable ones.

5. Experimental results

The video pool used is collected from the MSVD database where the videos are converted into frames. A single video contains up to 300 frames and one frame belongs to one category. There are 50 categories or actions considered for this method and the input image is classified into any of the classes. Based on the classes, the captions are generated. The accuracy of the RF method is calculated by how suitable the captions are generated for the frames. For further clarification, performance measures are calculated. They are,

Accuracy

Accuracy gives the correct prediction of the result. It is calculated by the equation (21),

$$\frac{(TP+TN)}{(TP+FP+TN+FN)} \tag{21}$$

Precision

It represents how many true positive results are precise.

$$\frac{TP}{(TP+FP)} \tag{22}$$

Recall

It is calculated based on the ratio of true positive from sum of true positive and false negative.

$$\frac{TP}{(TP+FN)} \tag{23}$$

F1-Score

It uses precision and recall to measure the accuracy of the results.

$$\frac{(2*Recall*Precision)}{(Recall+Precision)} \tag{24}$$

Using the intended value from the formula above, the performance measures are calculated and given in Table 5 below. True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) are the measures used for estimation.

Table 6: Performance evaluation of RF method

Performance Metrics	*Values in %*
Accuracy	82.23
Precision	78.38
Recall	74.26
F1-Score	71.15

The values namely Accuracy, Precision, Recall, and F1-score are calculated and the accuracy value gives the performance of the RF method.

The chart also shows the performance of the Random Forest method where all four values are compared and given in Fig.4.

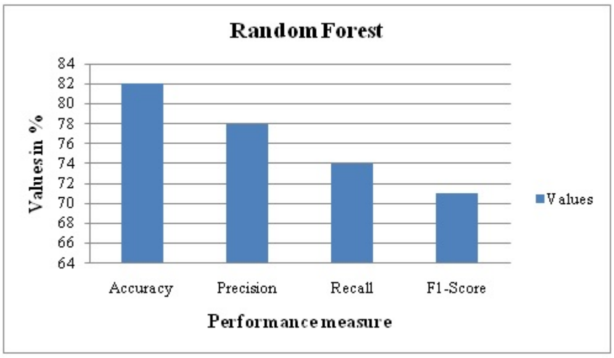


Fig.4 Performance measure of the RF model.

The CNN architecture is built and the images are used for training purposes to classify the images into different fifty classes. After the model is trained well, the images are tested with this model to find how accurate the prediction is given by this model. The performance of this model is evaluated using the metrics and a bar chart is constructed for accurate comparison of measures like accuracy, precision, recall, and F1 score.

The CNN parameters used are the convolution layer, filters used are 64 of size 4 x 4 and 2 x 2, the stride is 2 and the dropout factor is given as 0.2. Relu is the activation function used with batch normalization and finally, 2 dense layers are used to filter the features.

The performance measures of the CNN model are given in Table 7.

Table 6: Performance evaluation of CNN method

Performance Metrics	*Values in %*
Accuracy	96.10
Precision	95.51
Recall	95.32
F1-Score	91.08

The performances of the CNN Model are evaluated using Accuracy, Precision, Recall, and F1-Score which are explained in the above Table and the diagrammatic comparison of the performance measures of the CNN Model is shown in Fig. 5.

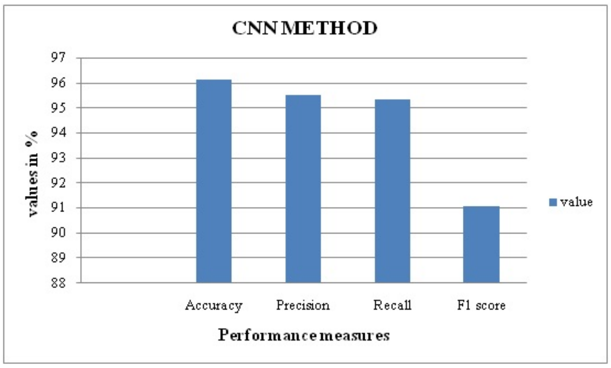


Fig.5 Performance measure of the CNN model.

6. Comparative Study

Comparing both the classification methods namely Random forest and CNN architecture for classifying the images into 50 different classes where each class is assigned unique actions. CNN model performs well by showing the highest accuracy of 96.10% compared with the RF method which shows accuracy up to 82.23%. Fig. 6. Shows the comparative result of both Machine learning and CNN model. With this result, it is clear that the deep learning method performs well than the machine learning techniques used in this method. So the captions generated are more suitable when used with the deep learning method. The two models are tested with MSVD data set and the results are clearly shown in Fig.7. Training and testing phases are introduced so the model is sound skilled and tested for the images.

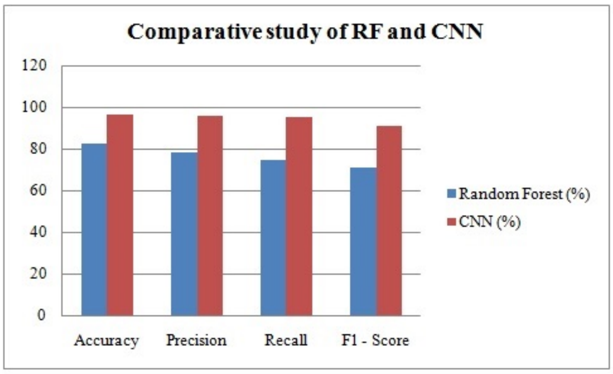
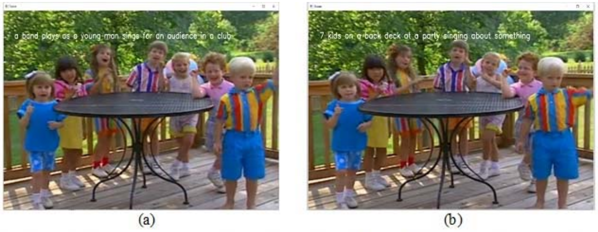


Fig.6 Comparative result of Random forest and CNN model.


a band plays as a young man sings for an audience in a club
7 kids on a back deck at a party singing about something
(a)
(b)

I.e. A band plays as a young man sings for an audience in a club.

I.e. 7 kids on a back deck at a party singing about something

Fig. 7. Shows the results of two different models.

(a). Result of Machine learning technique (ie. Random Forest) (b). Result of CNN.

From the above picture, figure (a) is the output of the machine learning method where the caption is general and the figure (b) shows the CNN result predicting an accurate caption.

7. Conclusion

This paper discussed the automatic generation of video captions while playing scenes in the video either online or in educational and entertainment media. MSVD dataset is utilized to demonstrate the proposed models. The videos from the dataset are converted into frames and preprocessed where the size is rescaled to standard size. The preprocessed images are used for feature extraction in the first part and the images are classified using the RF method by utilizing the extracted features. The second part classifies the images directly after preprocessing using the standard CNN method. Both the models are compared and finally deep learning model performed more excellently than the other model. This model can be further modified for audio channeling and future research can be based on this novel architecture.

References

[1] Zhong, Yu., Hongjiang Zhang., Anil K. Jain,.: Automatic Caption Localization in Compressed Video. In: IEEE transactions on pattern analysis and machine intelligence, Vol. 22, No. 4 (2000).

[2] Study of Video Captioning Problem, Jiaqi Su Princeton University jiaqis@princeton.edu, (2018).

[3] Jain, A.K., Yu, B.: Automatic text location in images and video frames. In: Pattern Recognition, vol. 31, pp. 2055-2076 (1998).

[4] Chien Cheng Lee., Yu-Chun Chiang., Hau-Ming Huang., Chun-Li Tsai.: A Fast Caption Localization and Detection for News Videos. In: Second International Conference on Innovative Computing, Information and Control (2007).

[5] Xingqi Wang., Guang Dong,.: A Novel Approach for Captions Detection in Video Sequences. In: International Conference on Fuzzy Systems and Knowledge Discovery, (2009).

[6] Watanabe, K., Sugiyama, M.: Automatic caption generation for video data. Time alignment between caption and acoustic signal. In: IEEE Third Workshop on Multimedia Signal Processing (1999).

[7] Suzuki, T., Kitazume, T., Sugiyama, M,.: The latest achievement of VC project for automatic video caption generation. In: IEEE Workshop on Multimedia Signal Processing (2002).

[8] Liu, Y., Dey, S., Lu, Y,.: Enhancing Video Encoding for Cloud Gaming Using Rendering Information. In: IEEE Transactions on Circuits and Systems for Video Technology, 25(12) (2015).

[9] Akshada, A., Gade., Arati, J., Vyavahare,.: Feature Extraction using GLCM for Dietary Assessment Application, In: International Journal Multimedia and Image Processing (IJMIP), Vol 8, Issue 2 (2018).

[10] Kanimozhi, P., Sathiya, S., Balasubramanian, M., Sivaguru, P., Sivaraj, P,.: Evaluation of Machine Learning And Deep Learning Approaches To Classify Breast Cancer Using Thermography, In: International Journal of Psychology and Education, Vol 58, Number 2 (2021).

[11] Xu, K., Ba, J., Kiros, R., Cho, K., Courvile, A., Salakhutdinov, R., Zemel, R., Bengio, Y,.: Show attend and tell: Neural image caption generation with visual attention. In: arXiv: 1502, 03044, 2(3):5 (2015).

[12] Vinyals, O., Toshev, A., Samy, B., Erhan, D,.: Show and tell: A neural image caption generator. In: CVPR, (2015).

[13] Donahue, J., Anne, L., Rohrbach, M., Venugopalan, S., Guadarrama, S., Saenko, K., Darrell, T,.: Long-term recurrent convolutional networks for visual recognition and description. In: CVPR, (2015).

[14] Anne, L., Hendricks, S. Venugopalan, Rohrbach, M., Mooney, R., Saenko, K., Darrell, T,.: Deep compositional captioning: Describing novel object categories without paired training data. In: CVPR, (2016).

[15] Fang, H., Gupta, S., Landola, F., Srivastava, K., Deng, L., Dollar, P., Jianferg, G.: From captions to visual concepts and back. In: CVPR, (2015).

[16] Mao, J., Xu, W., Yang, Y., Wang, J., Huang, Z., Yuille, A.: Deep captioning with multimodal recurrent neural networks (m-RNN). In: arXiv:1412.6632 (2014).

[17] S. Venugopalan, Anne, L., Hendricks, Rohrbach, M., Mooney, R., Saenko, K., Darrell, T,.: Captioning images with diverse objects. In: arXiv:1606.07770 (2016).

[18] Karpathy, A., Fei-Fei, L.: Deep visual-semantic alignments for generating image descriptions. In: CVPR, (2015).

[19] Johnson, J., Karpathy, A., Fei-Fei, L.: Densecap: Fully convolutional localization networks for dense captioning. In: CVPR, (2016).

[20] Hochreiter, S., Schmidhuber, J.: Long short-term memory. In: Neural computation, 9(8) (1997).

[21] Venugopalan, S., Xu, H., Donahue, J., Rohrbach, M., Mooney, R., Saenko, K.: Translating videos to natural language using deep recurrent neural networks. In: NAACL, (2015).

[22] Yao, L., Torabi, A., Cho, K., Ballas, N., Pal, C., Larochelle, H., Courville, A.: Describing Videos by Exploiting Temporal Structure. In: IEEE International Conference on Computer Vision (ICCV), (2015).

[23] Pan, Y., Mei, T., Yao, T., Li, H., Rui, Y.: Jointly modeling embedding and translation to bridge video and language. In: CVPR, (2016).

[24] Venugopalan, S., Rohrbach, M., Donahue, J., Mooney, R., Darrell, T., Saenko, K.: Sequence to sequence-video to text. In: ICCV, (2015).

[25] Pan, P., Xu, Z., Yang, Y., Wu, F., Zhuang, Y.: Hierarchical recurrent neural encoder for video representation with application to captioning. In: CVPR, (2016).

[26] Yang, Y., Zhou, J., Ai, J., Bin, Y., Hanjalic, A., Shen, H.T., Ji, Y.: Video captioning by adversarial LSTM. In: IEEE Image Processing, 27, 5600–5611, (2018).

[27] Donahue, J., Hendricks, L.A., Rohrbach, M., Venugopalan, S., Guadarrama, S., Saenko, K., Darrell, T.: Long-term recurrent convolutional networks for visual recognition and description. In: IEEE Trans. Pattern Anal. Mach. Intell., 39, 677–691(2017).

[28] Yan, C., Tu, Y., Wang, X., Zhang, Y., Hao, X., Zhang, Y., Dai, Q.: STAT: Spatial-temporal attention mechanism for video captioning. In: IEEE Trans. Multimed. 22, 229–241 (2019).

[29] Pan, P., Xu, Z., Yang, Y., Wu, F., Zhuang, Y.: Hierarchical recurrent neural encoder for video representation with application to captioning. In: IEEE Conference on Computer Vision and Pattern Recognition, (2016).

[30] Yu, H., Wang, J., Huang, Z., Yang, Y., Xu, W.: Video paragraph captioning using hierarchical recurrent neural networks. In: IEEE Conference on Computer Vision and Pattern Recognition, (2016).

[31] Anne Hendricks, L., Venugopalan, S., Rohrbach, M., Mooney, R., Saenko, K., Darrell, T.: Deep compositional captioning: Describing novel object categories without paired training data. In: IEEE Conference on Computer Vision and Pattern Recognition. (2016).

[32] Krizhevsky, A., Sutskever, I., Hinton, GE.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. (2012).

[33] Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: arXiv preprint arXiv:1409.1556, (2014).

[34] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucka, V., Rabinovich, A.: Going deeper with convolutions. In: IEEE Conference on Computer Vision and Pattern Recognition, (2015).

[35] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition, (2016).

[36] Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: IEEE Conference on Computer Vision and Pattern Recognition, (2016).

[37] Ren, S., He, K., Girshick, R., Sun, J.: Faster r-CNN: Towards real-time object detection with region proposal networks. In: Advances in neural information processing systems, (2015).

[38] Badrinarayanan, V., Handa, A., Cipolla, R.: Segnet: A deep convolutional encoder-decoder architecture for robust semantic pixel-wise labeling. In: arXiv preprint arXiv:1505.07293, (2015).

[39] Chen, L.-C., Papandreou, G., Kokkinos, L., Murphy, K., Yuille, L.: Deeplab: Semantic image segmentation with

deep convolutional nets, atrous convolution, and fully connected CRFs. In: arXiv preprint arXiv:1606.00915, (2016).

[40] Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition, (2015).

Vaishnavi J received M.Sc degree in Computer Science from PSGR Krishnammal College, Bharathiyar University, India in 2018, and M.Phil degree from Bharathidasan University, India in 2019. She has Cleared National Eligibility Test in 2020, currently working as the research scholar in the Department of Computer and Information Science, Faculty of Science, Annamalai University. Her research interests include Computer vision, Deep learning and Machine learning. She has published one paper in Scopus Journal and presented three International conferences.

Narmatha. V received the M.Sc degree in Computer Science from Madurai Kamaraj University, Madurai, India in 1999, and M.Phil degree from Alagappa University, Karaikudi, India. She has completed her PhD in Computer Science in Annamalai University, India in 2020 in the field of Medical Image Processing. She is working as an Assistant Professor in the department of computer and information science, Faculty of Science, Annamalai University, India. Her research interests include Image Analysis and Understanding, Computer vision and Machine learning. She has more than 22 years of teaching experience. She has published six papers in Scopus Journal, two papers in web of science, two in UGC Journal and presented ten International conferences also.