Video-Based Action Recognition Using Deep Learning Models

Aitore Issadykova
Nazarbayev University
Nur-Sultan, Kazakhstan
aitore.issadykova@nu.edu.kz

Assem Kussainova
Nazarbayev University
Nur-Sultan, Kazakhstan
assem.kussainova@nu.edu.kz

Zhaniya Koishybayeva
Nazarbayev University
Nur-Sultan, Kazakhstan
zhaniya.koishybayeva@nu.edu.kz

1. Abstract

Convolutional Neural Networks (CNN) are one of the most popular tools for video-based action recognition purposes and used in feature extraction part and in classification part. Several researches used CNN with mix of other methods for video classification purposes and achieved a superior accuracies with hybrid models. This project examined the performance of CNN in the video based action recognition for different sport activities from KTH and UCF Sports datasets.The different feature extraction methods such as Histogram of Gradients (HOG) and neural networks such as VGG16 and MobileNet were used and the overall performance of the CNN classifier compared with Random Forest classifier. The study provided the general CNN model for classifying different types of video datasets. With the general model and MobileNet feature extractor the accuracy of about 74% and 77% were achieved for well-known UCF Sports and KTH datasets, respectively . Moreover, the additional runtime analysis of each experimental setup was performed.

2. Introduction

Image processing has been developing exponentially during the last few decades because of the fast performance improvements in deep learning architectures. Convolutional Neural Networks (CNN) are considered to be the most popular tool for extracting the features from images. Great success in this field extended the idea to processing sequences of images, i.e. videos, using sequential models (Recurrent Neural Networks or RNN) such as Long Short-term Memory (LSTM). This fusion of image processing methods and sequential models is a very powerful method, used in a countless number of applications. State-of-art CNN models like VGG16, ResNet and GoogLeNet extract features from every frame of a video and feed this information to a RNN structure, which was designed to understand the sequential properties of these numbers. Deep learning models has been attracting the attention of many researches in recent years. This topic has a wide range of applications starting from enhancing safety measures through implementing this method to video surveillance to helping people with disabilities such as blindness, deafness in order to interact with the world more effectively.

This paper aims to work on video-based action recognition using deep learning models and compare performance with other classification method such as Random Forest. The current study will work with two well-known action datasets: KTH and UCF Sports action, and will use various feature extraction techniques to improve the results of the model. All methods will be compared by specified evaluation metrics and all limitation of such methods will be discussed.

3. Literature Review

Convolutional Neural Networks have proven being highly successful at static image recognition problems. By using feature pooling operations, CNNs are capable of automatically learning complex features required for visual object recognition tasks achieving superior performance to hand-crafted features. Encouraged by these positive results several approaches have been proposed recently to apply CNNs to video and action classification tasks.

Related to this field research has been done by Ng et al. [7] where authors use an approach of processing video files as frames using recurrent neural network with max-pooling that uses Long Short-Term Memory classifier which is connected to the output of the CNN. Authors used two CNN models for processing each individual frame of the input video files per second from Sports-1M and UCF-101 dataset: AlexNet and GoogleNet. In addition, for better capturing of motion information without loss an optical flow is added to the model. As a result, best achieved accuracies of action recognition for Sports-1M and UCF-101 dataset were 73.1% and 88.6% respectively.

Later Ullah et al. [12] improved previous approach performance in action recognition software by using features of the CNN and processing it through bidirectional LSTM algorithm with increasing the number of layers in the neural network models. Authors derived features from the UCF-101 dataset video frames with six frame jump, which are after sent to DB-LSTM in portions within time interval, where two layers are stacked on both forward and backward pass of the LSTM. In comparison with earlier work, current model is capable of learning long term complex sequences in videos and the proposed method improved the recognition rate on UCF-101 dataset from 88.6% to 91.21%.

Two most popular action datasets that are used for video-based classification purposes are KTH and UCF Sports. There are a lot of studies that used two datasets and built several distinctive models with high accuracies. One of such studies was conducted by Sargano and others [9]. Authors presented human action recognition method based on transfer learning using a CNN with SVM-KNN architecture which deals with overfitting problem. As baseline architecture for CNN feature extractor was used AlexNet trained on ImageNet dataset, consisting of 5 convolutional and 3 fully connected layers. After that, hybrid classification model on SVM and KNN algorithms was applied for action recognition on video. The proposed model achieved 98.15% and 91.47% accuracies for KTH and UCF Sports datasets, respectively, with Leave-One-Out cross validation.

The research conducted by Beikmohammadi and others [2] used improved 14 layers CNN - MobileNet, which was trained on ImageNet databases, for feature extraction. Since the classifiers such as SVM, LSTM, KNN, CNN and others are claimed to be costly for this research, authors used simple logistic regression for the classification purposes. The models used only 7 selected frames from the video for training and testing and achieved 98.81% and 96.47% accuracies for KTH and UCF-Sports respectively.

Ravanbakhsh et al. [8] proposed method which captures human motion through a hierarchical structure. Using video files from KTH, UCF Sport and UCF-11 Human Action datasets, authors extract frames from each video and features are computed for each frame using CNN, which are later mapped into a short binary code space. Key-frames are selected using the changes in each bit of the binary codes in given time and fed into hierarchical decomposition. For dimension reduction PCA method is applied for all levels, which are saved as vectors for a video snippet. Finally, the histogram of temporal words for all the videos is built and SVM classifier is used to predict action labels.

Charalampous, K. and A. Gasteratos [3] described an unsupervised on-line deep learning algorithm for action recognition in video sequences. Authors used publicly available datasets, including KTH and UCF Sports. The proposed method applies clustering algorithm and forms representation vectors from learned space. Video files were divided into frames, from which the spatio-temporal features were extracted suing Viterbi Algorithm, which is used for assignment of indices according to similarity between representation vectors and transition probability. In addition, for described feature extraction from files were utilized essential ART-2 clustering algorithm and L1-norm minimization methods.

The study of Shamsipour, Ghazal and Pirasteh, Saied [10] aims to contribute a method of artificial intelligence and CNN for recognition of human interaction by video collected from drone. Authors used SVM method to classify the stationary features obtained from pre-trained CNN with ImageNet along with application of PCA to every window from frame, and tested obtained model on the UCF Sports Action data with sufficient accuracy score which has indicated improvement of more than 3.17% as compared to the previous methods.

Basha et al. [1] introduced method which aims to classify human actions in a video, where the videos are captured at a distance from the performer. Researchers trained and evaluated the proposed 3D CNN model on KTH and WEIZMANN datasets. In this method frames from video were inputted to the 3D CNN model where spatio-temporal features were extracted. CNN model generates feature vector, which transferred to LSTM model, consisting of 1 hidden layer, which gathers decisions of 4 neighboring frames to classify an image according to sport actions from videos.

Previous researchers successfully used CNN for the feature extractions of the video based action dataset. Studies conducted with KTH and UCF-Sports dataset mostly used classic Machine Learning classifiers, such as SVM, KNN, etc. This project aims to build classifiers based on CNN and compare results with other studies.

4. Datasets

For this project we chose to work with two datasets suggested in the proposal: UCF sports action and KTH.

4.1. UCF sports action

UCF sports action dataset was downloaded from the official website of the Center for Research in Computer Vision [11]. The dataset represents video and image files of sports activities that are broadcasted through different media channels including ESPN and BBC. This collection of data includes 150 video sequences (almost 15 minutes), where each video file has a resolution of 720 x 480. In addition to the main viewpoints and perspectives of the objects in the video, additional sides and angles of the shooting were provided for some of the sports activities. The dataset has been used in research and projects, which are directed to the fields of action recognition, action localization, and saliency detection. The dataset includes 10 sports actions: Diving (14 videos), Golf Swing (18 videos), Kicking (20 videos), Lifting (6 videos), Riding Horse (12 videos), Running (13 videos), SkateBoarding (12 videos), Swing-Bench (20 videos), Swing-Side (13 videos), Walking (22 videos).

4.2. KTH

KTH is an open access dataset, publicly available on the website [5]. The dataset was first introduced in 2004, currently it consists of 600 .avi videos of all combinations of 25 subjects performing six actions (walking, jogging, running, boxing, hand waving and hand clapping) in four different scenarios (outdoors, outdoors with scale variation, outdoors with different clothes and indoors). The database consists of 2391 sequences shot on a static camera with 25 fps and 160x120 pixels resolution.

5. Methodology

5.1. Data Preparation

Before starting to work with building a deep learning model, the data from the selected datasets went through a preparation stage. Here it was required to transform data according to the goals and requirements of neural network models.

From the previously studied training data, the main KTH dataset was not changed, while for the UCF Sports Action dataset, it was necessary to combine data from some sports classes, since these classes contained repeating sport types, but shooting was made from different angles. For example, 3 perspectives of golf video files (front, side and back) have been grouped into one general class Golf-Swing. In addition, since video files were missing in some class folders (ex. Diving-Side/014), but contained ready-made frames obtained from the video, they were used as an input for training data.

5.2. Data Preprocessing

For experiments each video from datasets was stored as sequence of the particular number of frames. The number of frames was determined as minimum number of frames in all video within one dataset or was taken as maximum amount which RAM can processed as for KTH dataset. Moreover, in order to obtain some dynamic changes in the frames, the defined number of frames was skipped. The final list of frames was normalized and reshaped into 64*64 size.

Additionally, a list of videos from KTH dataset contains empty frames for active sports type such as walking, running and jogging. Empty frames are frames without human on the particular frame. Such frames were detected with Canny Edge detection method and were not included in the final data.

5.3. Feature Extraction

To determine the features of incoming frames from the video, it was required to find a solution for the model to be simultaneously fast, high-quality and economical in resources when processing a large number of images. The MobileNetV2 neural network met these criteria, the additional advantage of which is improved performance, compared to older models of neural networks which are also used to extract image features [6].

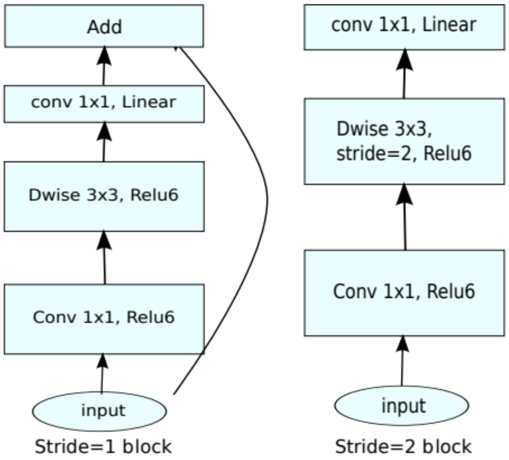

Figure 1. MobileNetV2 architecture

On the Fig.1 shown that the MobileNetV2 block called the expansion convolution block consists of three layers. First comes the pointwise convolution with more channels, called the expansion layer. It is followed by depthwise convolution with ReLU6 activation. This layer, together with the previous one, essentially forms the already familiar building block of MobileNetV1. At the end there is a 1x1 convolution with a linear activation function that reduces the number of channels without losing useful information.

In addition to the main approach for extracting features, several methods were also chosen to gain features of video action frames for further comparison of their performance in combination with classifiers.

Firstly, the Histogram of Oriented Gradients method was implemented for the simple extraction of features from video frames. HOG works according to the following principle. In a previously transformed image, it is needed to look at each pixel to determine how dark the current pixel is compared to the pixels directly adjacent to it. An arrow is then drawn showing the direction in which the image be-

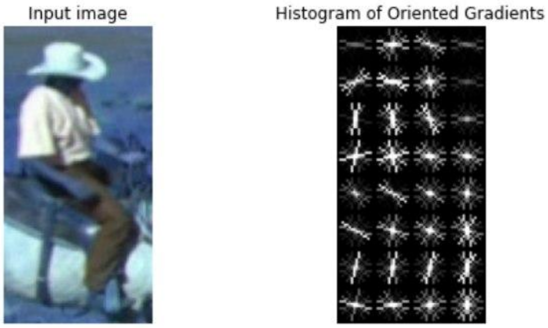

Figure 2. Feature extraction example with HOG method

comes darker. If this process is repeated for each individual pixel in the image, then eventually each pixel will be replaced by an arrow. These arrows are called gradients and they show the flow from light to dark throughout the image. Thus, we get the basic structure of the image through the streams of light and dark. The example of application of HOG feature extraction method is presented on Riding-Horse class image from UCF Sports action dataset (Fig. 2).

The second method extracts features of images dataset using a pretrained VGG16 neural network model. VGG16 is a convolutional neural network model that is an improved version of AlexNet, where large filters are replaced by several 3x3 filters one after another [4]. The VGG16 architecture is shown in Figure 3 below.

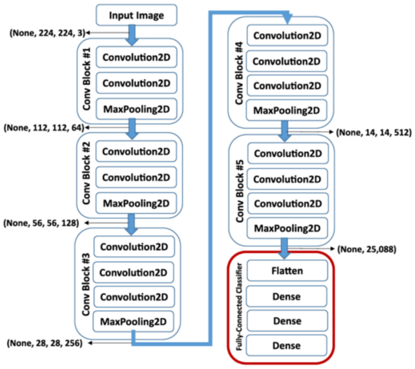

Figure 3. Architecture of VGG16 neural network

The input of the conv1 layer is 224x224 RGB images. The images then pass through a stack of convolutional layers that use filters with a very small receptive field of 3x3. There are five max-pooling layers in the network, which follow one at a time after some of the convolutional layers. The max-pooling operation is performed on a 2x2 pixel window. After the stack of convolutional layers are three fully connected Dense layers: the first two have 4096 channels, the third has 1000 channels. The last one is the soft-max layer. The configuration of fully connected layers is the same in all neural networks.

5.4. Model Architecture

To accomplish the task, several models of neural networks were considered, among which were various installations of Convolution 2D, RNN, LSTM and other approaches from previously conducted studies with selected datasets. From all tested models with basic settings, the highest accuracy was obtained using 3D CNN application. The baseline model for video action prediction purposes was constructed using 12-layers of Convolutional Neural Network reflected on Fig.4.

The architecture of the model starts with Batch Normalization at first layer for normalizing the input layer by adjusting and scaling the activations, followed by three 3D Convolution layers with "relu" activation function and with "channels first" data format. Next layers are Dropout with rate 0.1 and only one MaxPooling 3D with pool size = 2 due to small size of input data. Then the model adds another Dropout layer with 0.1 rate, one Flatten layer to flatten the input for next layer, and one more Dropout layer with rate = 0.1 to deal with overfitting issues. The last two layers are the Dense layer with kernel initializer "normal" and "softmax" Activation layer. The whole model is compiled with categorical cross-entropy loss because of multiclass classification problem, RMSprop optimizer was chosen due to relatively small size of batch, and accuracy was chosen as a main metrics for the model.

The model of the Random Forest was also chosen as the classifier of actions in the video for comparison with CNN model. This model uses all frames and is tuned using the default parameters: the number of trees in the forest is 100 and random_state equal to 0.

The purpose of applying these video classification methods is to compare the models' performance using both methods of extracting features from computer vision and deep learning with their subsequent application on selected classifiers to study the differences in their efficiency.

Note: Described CNN model has also been applied in the CSCI 594 Deep Learning project along with MobileNetV2 feature extraction. The aim of the work was to find a universal model for recognizing video for different input data from several datasets, and to test its performance on the RusLan dataset, consisting of Russian-language gestures. The prediction accuracy of the model was compared with previous state-of-the-art work with these datasets.The main purpose of CSCI 594 Deep Learning project is to provide baseline for RusLan dataset, which was not used in any studies before.

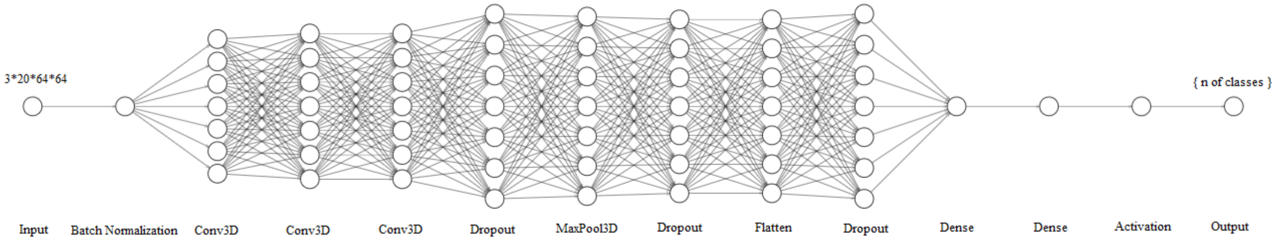

Figure 4. Proposed CNN model

6. Experimental Setup

6.1. KTH

For KTH dataset the final data contained 25 frames with skipping 2 frames in original data, which resulted in 3 seconds segment from the original video. After prepossessing and feature extraction the data was divided on the train and test with ratio 80% : 20%. For replication of the similar results the random state was set on 21. The best model obtained was trained with 20 epochs and with batch size equals to 32.

6.2. UCF Sports action

Initially 20 frames were stored as the final data and the number of frames skipped was defined for each class separately because of significant difference in the length of video for each class. The setup uses the obtained final data, which was preprocessed and split to the train and test datasets with ratio 85% to 15%. For obtaining each class in the test dataset the random state of the splitting was set to 19. The best model was obtained by training the model with 30 epochs and by declaring batch size to 32.

6.3. Random Forest classifier

For both datasets for the Random Forest classifier the number of frames were decreased to 10 and performed additional preprocessing of flattening image. Moreover, due to limited support on Conv3d layers on TPU runtime, the TPU on Google Colab application was used only for Random Forest classifier, while GPU was used on CNN classifier. Other experimental setup for this classifier is the same as it was described before.

7. Evaluation Metrics

Since the project's main target is a classification problem and the classes in the training sets are more or less balanced, accuracy will be taken as major evaluation metric. Additionally, the Results Section will include efficiency (runtime) in the evaluation and in the process of choosing optimal feature extraction and classifier models. Moreover, the weight difference between VGG16 and MobileNet feature extractors will be taken into account, since VGG16 weights almost 36 times more than MobileNet. For similar results from these two neural network feature extractors, the MobileNet will be chosen for its memory cost side for future works with large datasets.

8. Results

Table 1 presents the accuracy and runtime of each combination of feature extractors (HOG, MNetV2 and VGG16) and classifiers (RF and CNN). First, it is clear that CNN classifiers give greater or equal highest accuracy, than Random Forest: for KTH dataset all models with CNN works strictly better then RF for each feature extractor; for UCF dataset all accuracies are comparable except for the one under the MNetV2 feature extractor model, for which the highest achieved accuracy of CNN is better. The second observation is that on average MNetV2 is the fastest feature extractor, VGG16 is the second in the list and HOG takes the most time to finish. For example, for KTH dataset working with CNN, MNetV2 would wrap up in 91.54 seconds, VGG16 - in 137.18 seconds, and HOG - in 185.26 seconds. These results are easily explained analytically: MNetV2 occupies only 14 MB as a model, whereas VGG16 takes more than 500 MB, and HOG takes the longest because it is a slow feature extractor itself and on top of this it is applied to each frame separately. Additionally, RF is expectedly running faster than CNN, when comparing their performances holding everything else constant.

KTH dataset performed best under HOG+CNN model with 87% accuracy, but the fastest result was obtained by MNetV2+RF model in 35.44 seconds.

The MNetV2+CNN model showed the best accuracy for UCF Sports Action dataset - 74%. The least runtime is associated with MNetV2+RF model.

Even though for KTH dataset HOG+CNN gave the largest accuracy, it takes too much time to run, and for bigger datasets it would be more preferable to have a faster model, for example MNetV2+CNN which takes half as much time to finish. Apart from that, this model gives the

	HOG+RF	HOG+CNN	MNetV2+RF	MNetV2+CNN	VGG16+RF	VGG16+CNN
KTH Accuracy	67%	**87%**	55%	76%	55%	73%
KTH Runtime	71.08s	185.26s	**35.44s**	91.54s	58.08	137.18
UCF Sports Action Accuracy	57%	57%	65%	**74%**	65%	65%
UCF Sports Action Runtime	19.15s	43.39s	**9.19s**	23.52s	26.73s	29.94s

Table 1. Comparison of the results

best accuracy for UCF Sports Action dataset, and though it is not the fastest one, its runtime is optimal and any change does not worth sacrificing the accuracy. Therefore, MNetV2+CNN model is chosen for video-based action recognition in terms of accuracy, runtime and memory.

Table 2 describes the results obtained on two datasets and provides the results obtained by other researchers. Overall, the general model performed less accurate than existed studies. However, other studies used as classifiers LSTM, SVM and KNN models and added some additional feature such as transfer learning and spatio-temporal feature extraction. Transfer learning is not recommended to use for small datasets and spatio-temporal feature extraction requiers some additional data preprocessing for each dataset explicitly. This study focused on the providing general CNN model classification for different type of video datasets.

Previous approaches	KTH	UCF Sports Action
Proposed architecture	76.67%	73.91%
Ravanbakhsh et al.[8]	74.5%	88.1%
Charalampous, K. and A. Gasteratos [3]	91.99%	88.55%
Shamsipour, Ghazal and Pirasteh, Saied [10]	-	**93.67%**
Basha et al. [1]	95.27%	-
Sargano et al.[9]	**98.15%**	91.47%

Table 2. Comparison of results with existing studies

8.1. Error Analysis

8.1.1 KTH

Fig. 5-10 illustrate confusion matrices for different models for KTH dataset. It is easy to notice the visual clustering of the first three and the last three classes on each matrix. Taking into account the factors the nature of actions in KTH are same, i.e boxing, handclapping, handwaving and running, jogging and walking, it was expected that the models will misclassify the classes between each other. Additionally, boxing could be misclassified as running, jogging or walking due to the video instances when a subject performs boxing action while running in place. Almost all models commonly have the most accurate predictions on boxing, handclapping and walking classes, and the least accurate classes: handwaving, running and jogging.

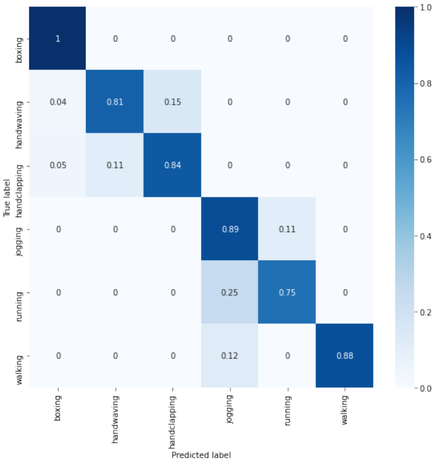

Figure 5. Confusion Matrix of HOG+CNN model (KTH dataset)

Fig. 7, represents the confusion matrix of model of choice (MNetV2+CNN). All classes have prediction accuracy not smaller than 62%. Boxing, handclapping and walking were predicted with 88-89% accuracy.

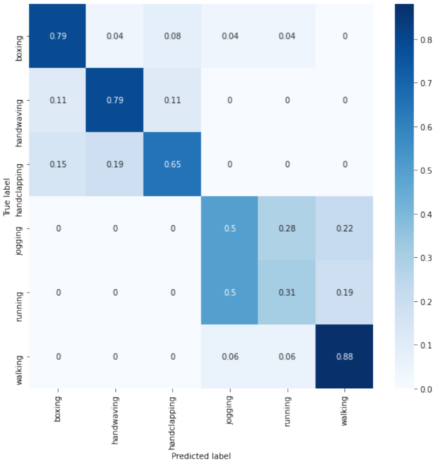

Figure 6. Confusion Matrix of HOG+RF model (KTH dataset)

8.1.2 UCF Sports Action

Fig. 11-16 illustrate confusion matrices for different models for UCF Sports Action dataset. For Random Forest classi-

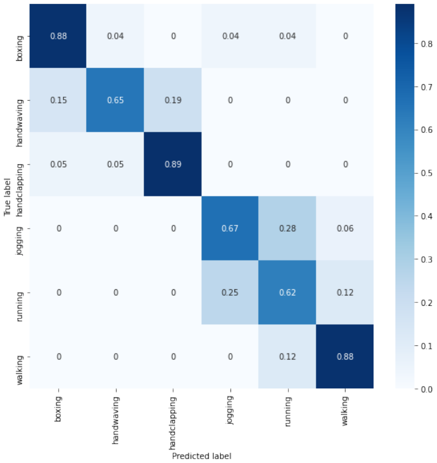

Figure 7. Confusion Matrix of MNetV2+CNN model (KTH dataset)

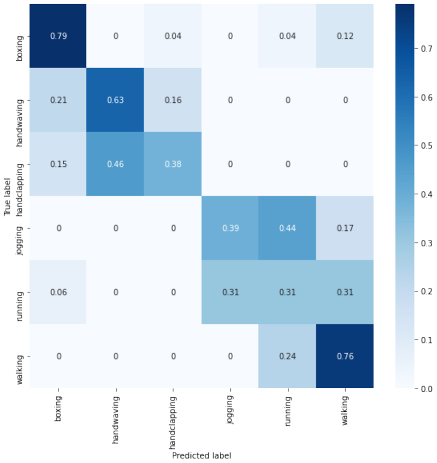

Figure 8. Confusion Matrix of MNetV2+RF model (KTH dataset)

fier it was common to misclassify Run-Side, Skateboarding-Front and Walking-Front the most (Fig. 12, 14, 16). For some reason all of the Walk-Front videos were classified as Swing when using RF.

Fig. 13, represents the confusion matrix of model of choice (MNetV2+CNN).Six classes achieved 100% accuracy, but Golf-Swing, Run-Side and Skateboarding-Front got only 40%, 33% and 67% of accuracy, respectively. The misclassification of data from these classes is influenced by the presence of other people on the video and their poses in the frames, which the model also perceives as equivalent objects. In a Skateboarding class, people walk alongside the subject during the video, which can also be classified as Walk class. The same tendency is observed with the Golf and Skateboarding classes.

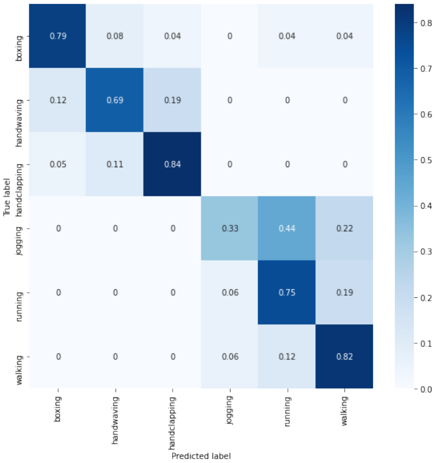

Figure 9. Confusion Matrix of VGG16+CNN model (KTH dataset)

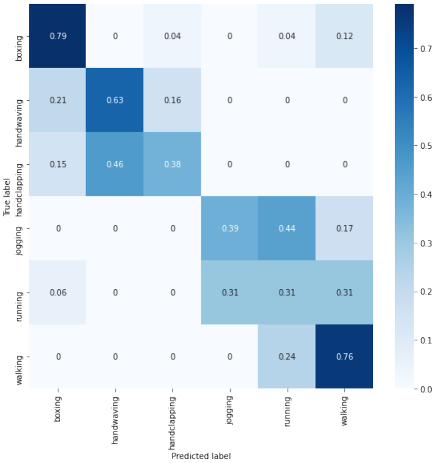

Figure 10. Confusion Matrix of VGG16+RF model (KTH dataset)

9. Limitations and Future Works

9.1. Limitations

One of the main limitation of the current study is memory, especially, RAM restriction on the Google Colab system. Because of this limitation, the current study wasn't able to perform PCA reduction and other dimension reduction techniques for Random Forest classifier preprocessing part and it was one of the reasons for limiting number of frames to 10.

The limited time is another limitation which was faced during this project. Due to heavy memory allocation of the HOG descriptor, it was costly to determine best optimal parameter for models with HOG feature extractor. In addition, it is still possible to find more accurate models with same experimental setup with having greater research time.

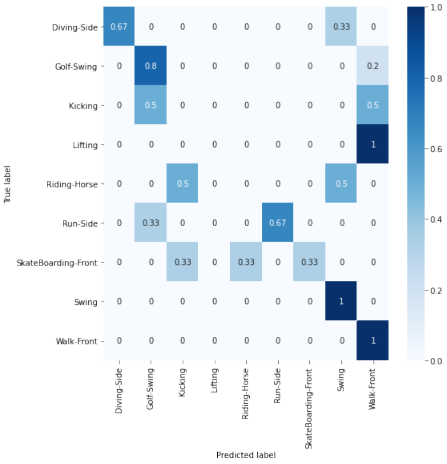

Figure 11. Confusion Matrix of HOG+CNN model (UCF dataset)

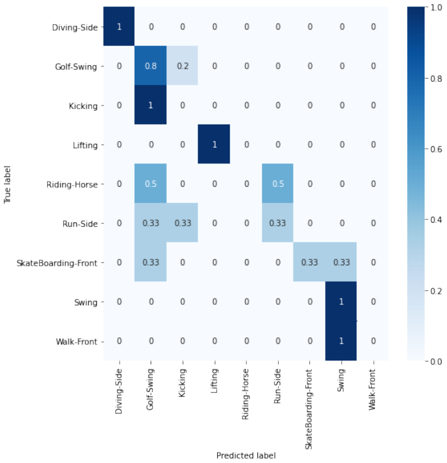

Figure 12. Confusion Matrix of HOG+RF model (UCF dataset)

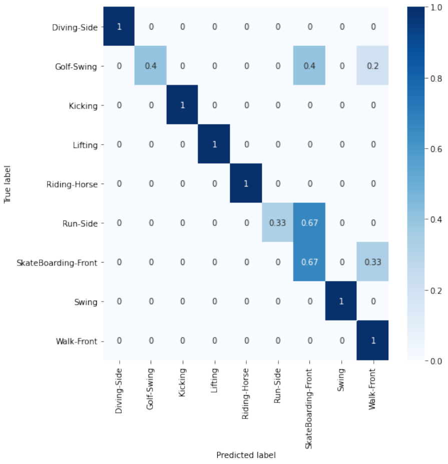

Figure 13. Confusion Matrix of MNetV2+CNN model (UCF dataset)

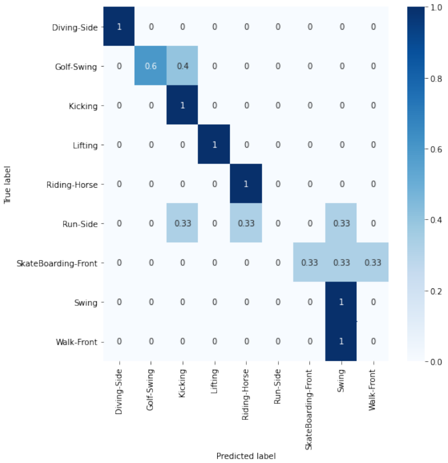

Figure 14. Confusion Matrix of MNetV2+RF model (UCF dataset)

9.2. Future Works

For the future researches it is recommended to compare performance of the optimal proposed model with MobileNet feature extractor and CNN classifier on the large datasets like UCF101 and others. In addition, it is planned to replicate all experimental setups on the different categories of the video such as video recognition of weather and other subjects of interest.

As a continuation of the current work, it is recommended to tune further the parameters of CNN model in order to outperform the achieved accuracies. Furthermore, some dimensional reduction techniques such as PCA or Autoencoder Neural Network should be performed for optimizing random forest classifier.

10. Conclusion

Video based classification is one of the most popular topics in the Machine Learning field. Object, action and speech recognition are used in different spheres including medicine, engineering and others. This study proposed the general model for video based classification using Convolutional Neural Networks and analyzed its performance using several feature extractors such as HOG, VGG16 and MobileNet and compared it with Random Forest Classifier. This paper analyzed the performance of the well-known action datasets KTH and UCF Sports, provided the optimal model for action recognition and compared it with existing studies with other researchers. The accuracy of proposed general model is lower than of existed papers; however, it is one of the minority papers that used CNN as classifiers for general model for both datasets. In addition, this work

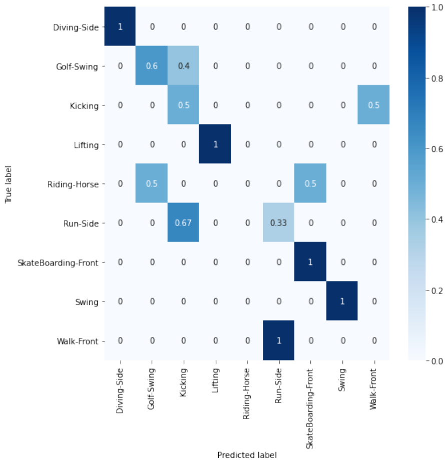

Figure 15. Confusion Matrix of VGG16+CNN model (UCF dataset)

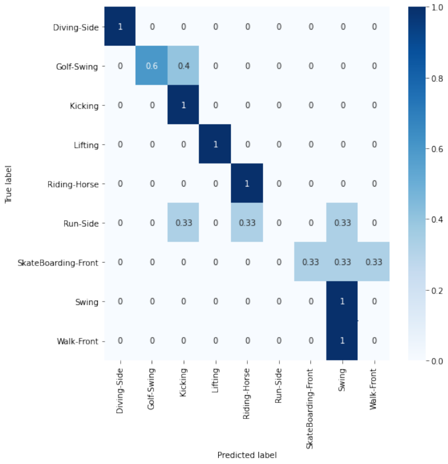

Figure 16. Confusion Matrix of VGG16+RF model (UCF dataset)

provides some runtime cost analysis of each experimental setup, which was not done in previous researches.

Possible future research was discussed for extending the current work. Limitations include both time and memory cost issues which should be taken into account for the future researches.

References

[1] Basha, Sh Shabbeer and Pulabaigari, Viswanath and Mukherjee, Snehasis. (2020). An Information-rich Sampling Technique over Spatio-Temporal CNN for Classification of Human Actions in Videos.

[2] Beikmohammadi, A., Faez, K., Mahmoodian, M. H., and Hamian, M. H. (2019, December). Mixture of Deep-Based Representation and Shallow Classifiers to Recognize Human Activities. In 2019 5th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS) (pp. 1-6). IEEE.

[3] Charalampous, K. and A. Gasteratos, On-line deep learning method for action recognition. Pattern Analysis and Applications, 2016.19(2): p. 337-354.

[4] Gopalakrishnan, Kasthurirangan and Khaitan, S.K. and Choudhary, Alok and Agrawal, Ankit. (2017). Deep Convolutional Neural Networks with transfer learning for computer vision-based data-driven pavement distress detection. Construction and Building Materials. 157. 322-330. 10.1016/j.conbuildmat.2017.09.110.

[5] KTH dataset. Available at: https://www.csc.kth.se/cvap/actions/.

[6] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov and L. Chen, ”MobileNetV2: Inverted Residuals and Linear Bottlenecks,” 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, 2018, pp. 4510-4520, doi: 10.1109/CVPR.2018.00474.

[7] Ng, Joe and Hausknecht, Matthew & Vijayanarasimhan, Sudheendra & Vinyals, Oriol & Monga, Rajat & Toderici, George. (2015). Beyond short snippets: Deep networks for video classification. 4694-4702. 10.1109/CVPR.2015.7299101.

[8] Ravanbakhsh, M., Mousavi, H., Rastegari, M., Murino, V., & Davis, L. (2015). Action Recognition with Image Based CNN Features. ArXiv, abs/1512.03980.

[9] Sargano, A. B., Wang, X., Angelov, P., & Habib, Z. (2017, May). Human action recognition using transfer learning with deep representations. In 2017 International joint conference on neural networks (IJCNN) (pp. 463-469). IEEE.

[10] Shamsipour, Ghazal and Pirasteh, Saied. (2019). Artificial Intelligence and Convolutional Neural Network for Recognition of Human Interaction by Video from Drone. 10.20944/preprints201908.0289.v1.

[11] UCF sports action dataset. Available at: https://www.crcv.ucf.edu/data/UCF_Sports_Action.php.

[12] Ullah, Amin & Ahmad, Jamil & Muhammad, Khan & Sajjad, Muhammad & Baik, Sung. (2017). Action Recognition in Video Sequences using Deep Bi-directional LSTM with CNN Features. IEEE Access. PP. 1-1. 10.1109/ACCESS.2017.2778011.