Human Praying Structures Classification With Transfer Learning

Md. Ekram Hossain[1], Md. Shohel Arman[1], Syeda Sumbul Hossain[1], Afia Hasan[1], Mahmuda Rawnak Jahan[1] and Md. Anwar Hossen[1]

[1]Department of Software Engineering, Daffodil International University, Dhaka-1207, Bangladesh
ekram35-1936@diu.edu.bd, sshuvo27@gmail.com, syeda.swe@diu.edu.bd, afiahasan.28@gmail.com, jahan.swe@diu.edu.bd, anwar.swe@diu.edu.bd

Abstract. Action Recognition is one of the most important fields in computer vision. Hence there is an open question of high accuracy of complex background of human activities. A deep learning approach has recently been used to increase recognition validity with different application areas such as video surveillance, entertainment, autonomous driving vehicles, and human-machine interactions, etc. The aim of this research is to recognize human religious actions that differ in different activities. In our study, we have created our dataset from religious praying videos collected from YouTube which has been classified in four different classes in terms of religion. We have applied a deep convolutional neural network using the Resnet-50 model for identifying Human Activity Recognition (HAR) and we have got 98.79% accuracy. This research will help to cover more human action recognition tasks of daily activities.

Keywords: Learning, Residual Neural Network, Frame Classification, Human Action Recognition, Human Praying Structure, Video Frames, Action Dataset

1 Introduction

Over the past decade, recognition of human actions (HAR) has become the subject of a more interesting study, which includes many applications such as video surveillance, virtual reality, intelligent humancomputer interaction, and more. There are several stages in HAR that describe the features. There are two steps, we can explore of human action from the sequence of images: 1) Extraction of complex features from input videos, 2) Create a classification that is typical for human activity such as Oriented Gradient (HOG) Histogram (HOG) [1], Histogram of Optical Flow (HOF), Motion Interchange Patterns (MIP), Space-Time Interest Points (STIP), Uses features [2] and dense trajectories [3]. However, these methods

are difficult and time consuming to extend features to other systems. Therefore, to reduce the need for hand-engineering features, it is necessary to propose generic feature extraction methods and reduce the scale of calculations. CNN [4] is an in-depth model that achieves complex classified features through alternate convolutional operations with sub-sampling operations in raw input images.

A deep convolutional neural network architecture [5] has been proposed to detect human activities in videos using the action bank features which is in the UCF-50 database. A novel dynamic neural network model has been proposed that can identify the types of dynamic visual images of human actions learning [6]. A new approach is proposed for By combining partbased models and in-depth learning through post-normalized CNN training a new approach is proposed in [7]. A CNN architecture using ResNet-50 is proposed to detect human actions in videos using the most famous database which is UCF-101 [8]

There are lots of human action-based open datasets available that can successfully identify human actions written in different literatures. However, we have not found any dataset that specifically can recognise human praying activities. The objective of this research is to make a specific dataset that can be used to identify human praying actions in terms of different religions. We have created a religious praying-based video dataset from YouTube and classify those videos in four categories as Muslim praying, Christian praying, Hindu praying and Buddha praying as in Bangladesh there are main four classes of religions.

2 Literature Review

Action recognition has been studied over the years. Initially the focus was on developing features designed to represent work like 3D-SIFT [9] and Dense-Trajectory [10]. With the development of deep convNets, introducing features for the automatic management of convNet, several recent convNetbased approaches have been proposed to act as recognition. Ji et al. [11] uses a 3D convent to detect actions in the video. Simonian and Zisserman [12] proposed a bi-stream framework that uses two convNets to extract and recognize the properties of two data streams (e.g., appearance and motion) respectively. Based on this structure, recent studies have further improved the functionality of ConvNet properties by incorporating additional data sources [13], unconformable spherical images [14] on convolutional neural network (CNN) based robots.

Much of the existing tasks are aimed at knowing features to give a direct description of the individual activities of the activities, while the features shared in the monsters of different activities are little studied. This prevents them from clearly differentiating between subtle and subtle activities. Yet, some methods [15] achieve generality at different levels by integrated features into multi-convNet layers. They

focus directly on representing individual action classes and do not consider the sharing features of different activity partners. In addition to the development of appropriate features, other studies have been focused on the appropriate combination of multiple data flows to improve function recognition effectiveness [16-18].

Abu-El-Haiza and others are considering the need for dedicated datasets for large-scale multi-label video classification [19], More than eight million videos with 500,000 hours of playtime and 4,800 visual qualities were labeled using the YouTube video annotation system. They divided the videos into several categories such as vehicles, sports, and concerts. They review both classical and deep-learning approaches applied to visual classification in sports videos. Minhas et al. [20].

A famous deep learning model [27], AlexNet CNN is commonly used for classifying frames from sport videos, Nvidia GTX580 Graphics Processing Unit (GPU) employs four classes for network training from scratch. Overall training and validation are encouraged using performance response normalization and dropout levels. That method showed 94.07% accuracy as compared to the baseline method in the subject dataset. Russo et al. [21]. The model demonstrates fair accuracy of 10% and 92% for ten sports classes. Ng et al. [22] are capable of handling full-length 2ideosu se of the AlexNet and GoogleNet models. While evaluating the proposed architecture, UCF-111 used 487 classes with 11.3 action classes and approximately 11.2M YouTube sports videos with 13,320 videos.

3 Methodology

3.1 Data Collection Process

There is various dataset available for human action recognition but those datasets have not prayed structure classes. Those are some small and some large datasets. KTH [23], Weizmann [24], UCF Sports [25], IXMAS [26] datasets include only 6, 9, 9, 11 classes respectively. Table 01 represents Action dataset examples. That of all dataset has a disadvantage of unrealistic video clips. We have tried to build a dataset of realistic video clips. We have downloaded 15 video clips for each class from YouTube and saved as .avi format. After that, we labeled those videos in four categories: Muslim praying, Hindu praying, Christian praying, and Buddha praying. Figure 01 shows some examples of collected data in different categories.

Figure 1. Collected Data Categories

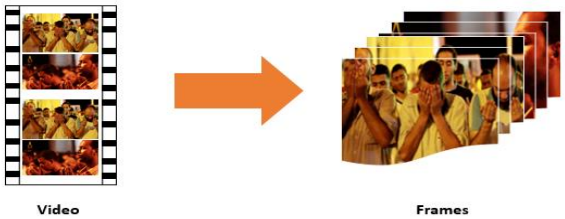
Table 1. Action Dataset Example

Datasets	*Classes*	*Clips*	*Background*	*Motion*	*Year*	*Resource*
KTH	6	600	Static	Slight	2004	Actor Staged
Weizmann	9	81	Static	No	2005	Actor Staged
UCF Sports	9	182	Dynamic	Yes	2009	TV, Movies
IXMAS	11	165	Static	No	2006	Actor Staged
UCF11	11	1164	Dynamic	Yes	2009	YouTube

3.2 Data Pre-Processing

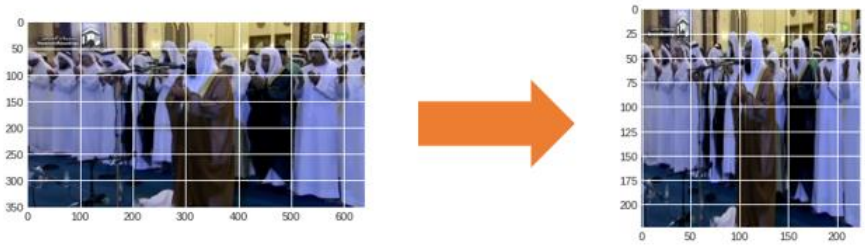
Our dataset contains four classes as video clips from which we need to create frames using OpenCV . Each video clip is divided into several parts in every second. We have five frames in one-second video. RGB color channel allocation is also done at this level. In our case, we have taken three channels. Figure 02 shows the extracted frames in one second video.

Figure 2. Frames Extraction



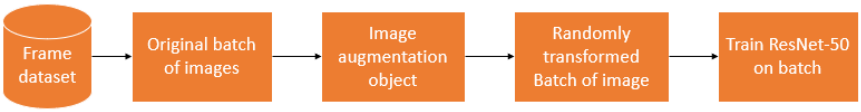
After that, we have stocked the path of the frame dataset as a variable then create a function to load the frames into an array. Video frames are different sized frames. For our model, we need to convert all the frames to base size. We have analyzed that large size images are very complex for our model but reduced the image size to (224,224,3) has benefited us. By doing so, we have managed to reduce the training time. Figure 03 shows the resized frames.

Figure 3. Frames Resizing



Then, we apply data augmentation to increase the generalizability of the model. It is the process of spreading the length of the training dataset by correcting the images in the dataset. This increases the accuracy of the model. Here, we use on-the-fly data augmentation to create great models. Figure 04 shows the data augmentation process we have used in this research.

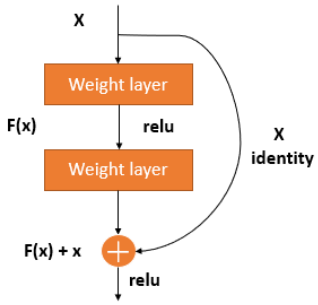
Figure 4. Data augmentation process



4 Model Architecture

Siamese Neural networks with deep understanding have made multiple breakthroughs for image classification. Characterized by the significance of the depth, a question arises: Is it easier to learn better networks like stacking more layers? One obstacle to answering this question is the infamous problem of extinction / explosion of gradients, which hinders expression from the very beginning. This problem, however, is generally solved by the initialization and intermediate normalization layers that enable networks with tens of layers to initiate the conversion of stochastic gradient descent (SGD) with backpropagation. This erosion problem can be solved by introducing deeply residual learning structures. Figure 05 shows the Residual Learning Building Block.

Figure 5. Residual Learning Building Block



Let us consider H (x) as the underlying mapping to fit by several stacked layers (not necessarily the whole net), x refers to the inputs in the first of these layers. If one assumes that multiple nonlinear layers can perform complex tasks approximately, it is equivalent to assuming that they can guess the remnants without shelter. H (x)- x (assuming that the input and output are at the same level). So, the arrayed layers let us approximate these layers to approximate a residual function F (x) = H (x)- x rather than the approximate H (x) expectation. The main function thus becomes F (x) + x. Although both forms should be able to infer asymptotically desired functions (as hypothyroid zed), the ease of learning may differ. Although both forms are remarkably capable of enabling desirable tasks (as expected), the ease of learning may differ. A typical resNet-50 has all CNN components [28].

The process of learning features has three parts. The Convolution level extracts the high-level properties of each image from the input images. After we remove the high-level properties from the input images, we immediately apply a ReLU (non-Linear the rectified unit) to each convolution layer (Quality and summary only according to the material.

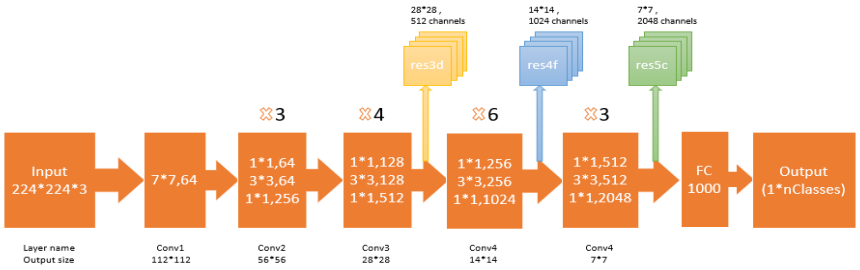
After ReLU we apply max pooling. The max pooling layer selects the best features from the initial features extracted by the convolutional layer. Max pooling gives us the best features that are multidimensional arrays. Since our fully connected layers are only learned in single-dimensional arrays, we need to plant multidimensional arrays before feeding the fully connected layers. Fully connected layers learn on flat inputs by applying backpropagation. For backpropagation and distribution of images, we have used the Stochastic Gradient Descent (SGD) function instead of ADAM. SGD is much more optimized and reduces compilation time for a long term training process [29]. Fully connected layers give an N dimensional vector output where the number of classes that the program has to choose from. Each number of these N dimensional vectors represents a certain class of probabilities.

5 Model Evaluation

5.1 Training Process

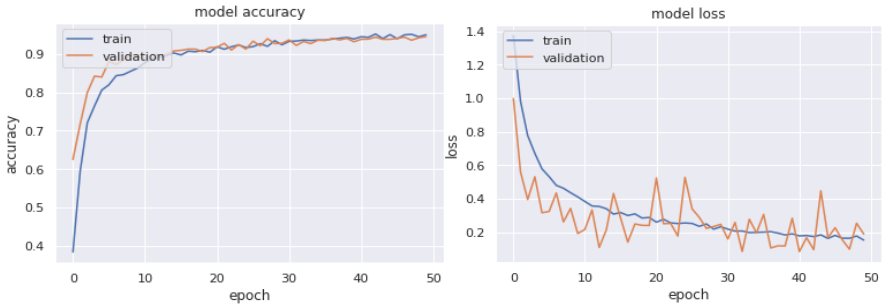
Figure 6 illustrates the overall training process and evaluation of the proposed method of model training. For training purposes, we have used the TensorFlow backend with Keras application that includes a dataset generator to manage large datasets of images with optimal memory usage.

Figure 6. ResNet-50 Architecture



We have split the dataset Between 70% and 30% for training and testing, respectively. We have used Adam optimizer, learning rate and training performance metrics. Figure 07 shows the progress of training over a period of more than 50 epochs and reflects sustainability and High success of our proposed model and the accuracy of training and validation are 98.79% and 97.85% respectively.

Figure 7. Model Accuracy and Loss



5.2 Optimizer Selection

During the experiments, various optimizers have tried to train and evaluate the model. The best optimizer has worked with SGD. We have applied six popular and best optimizers which are Nadam, Adam, Adadelta, SGD, RMSprop, and Adagrad.

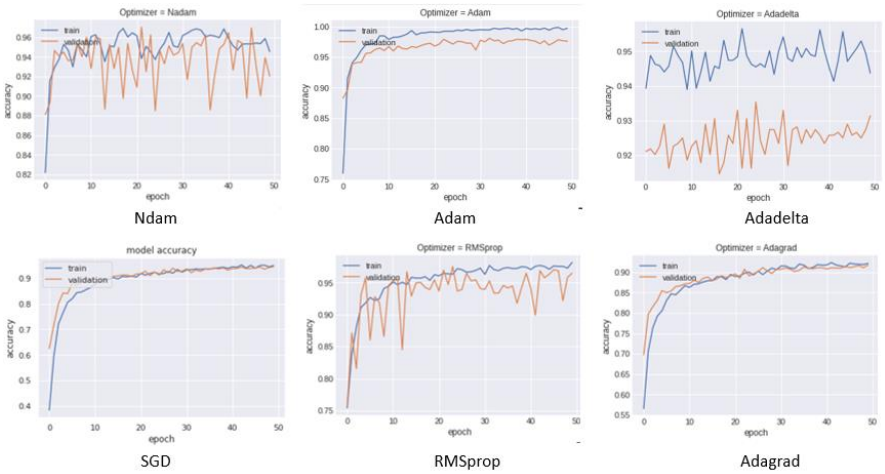
Adam has better accuracy for training but for validation and loss was poor from SGD optimizer. Table 02 shows the optimizer's accuracy.

Table 2. Accuracy with Optimizer

Optimizer	***Accuracy***	***Validation Accuracy***
SGD	98.79 %	97.85 %
Ndam	95.57%	92.36%
Adam	98.60%	97.83%
Adadelta	95.48%	93.50%
RMSprop	95.68%	95.36%
Adagrad	93.88%	92.88%

We can observe that the SGD optimizer is best for a long-time training procedure. Figure 08 presents the comparison among those optimizers.

Figure 8. Comparison of Optimizers



5.3 Simulation Parameter

Throughout our simulation using Python, we have used optimistic and realistic parameters for the video combination. We uploaded pre-trained weights with 1000 classes from ImageNet and used Adam Optimizer with optimal learning rate for training optimization. Table 03 represents the overview of simulation parameters

Table 3. Simulation Parameters

Method	***Parameter***
Language	Python
Pre-trained	ImageNet, 1000 classes
Optimizer	SGD optimizer
Learning Rate	0.0001
Loss Function	Categorical Cross-Entropy
Performance Matric	Accuracy
Total Class	04
Video file format	Avi
Augmentation	Scale, Rotate, Flip, Shift
Batch Size	32

5.4 Performance Evaluation

The Resnet-50 model evaluations expose the similarities and differences of inter classes. Table 4 presents the confusion matrix of those classes. The similarities of inter-class are Muslim praying structure and Christian praying structure. These two classes are significantly similar to each other and Muslim praying and Hindu praying structure are quite different from each other and also the same to Buddha praying structure.

Table 4. Confusion Matrics

Actual	***Buddha Praying***	***Christian Praying***	***Hindu Praying***	***Muslim Praying***
Buddha Praying	289	7	1	5
Christian Praying	0	323	7	2
Hindu Praying	2	9	308	2
Muslim Praying	1	1	6	322

The classification report of the ResNet-50 model is shown in Table 5. This report shows the quality of the prediction by evaluating Precision, Recall, and F1 score for each class.

Table 5. Classification Report

Class	*Precision*	*Recall*	*F1 Score*
Buddha Praying	99%	96%	97%
Christian Praying	95%	97%	96%
Hindu Praying	96%	96%	96%
Muslim Praying	97%	98%	97%

5.5 Prediction Result

After training our model, we tested on our test data set. We have created a video sample with a combination of our classes for a real-life test with this model. We saved our best model weight and loaded the model to fit our sample video. After prediction again we gather the predicted frames into video. Figure 09 presents some example of prediction results.

Figure 9. Prediction of Model

6 Conclusion And Future Work

Video can be categorized in different ways to identify human activities still there is a question of how it affects the overall performance. We have collected 15 videos from YouTube for each class and labeled those in terms of religions aspects. In our work, firstly we have used OpenCV to create five image-like frames from every second video and got multiple frame images for an acting class. Then we have split the train set and test set according to 70% and 30%. After that, we have trained our

model using ResNet-50 with evaluation matrices. F1 scores, precision, and recall metrics used for evaluation. The evaluation results show that the ResNet-50 model is great for recognizing human activities. We have also used a sliding window mechanism, which allows dealing with high-resolution textured images that can perform as expected with low-resolution images without changing the whole architecture. There are lots of human action datasets available and among them, UCF-101 is the most popular and large dataset. We have proposed a religious action new dataset which has praying structure-based videos. This work will help in planning and developing worship places in places by identifying most frequent religious activities.

In future work, we can increase more religious activities with realistic data in this dataset. Help of increasing the dataset there will add various levels of human religious activities that can be detected and cover more human activity recognition in this area

References

1. Huang Y, Yang H, Huang P. Action recognition using hog feature in different resolution video sequences [C]//Computer Distributed Control and Intelligent Environmental Monitoring (CDCIEM), 2012 International Conference on. IEEE, 2012: 85-88.
2. Sadanand S, Corso J. Action bank: A high-level representation of activity in video. In IEEE,2012: 1234-1241.
3. Wang H, Kläser A, Schmid C. et al. Dense trajectories and motion boundary descriptors for action recognition [J]. International journal of computer vision, 2013, 103(1): 60-79.
4. LeCun Y, Bottou L, Bengio Y. et al. Gradient-based learning applied to document recognition [J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.
5. Ijjina E P, Mohan C K. Human Action Recognition Based on Recognition of Linear Patterns inAction Bank Features Using Convolutional Neural Networks[C]//Machine Learning and Applications (ICMLA), 2014 13th International Conference on. IEEE, 2014: 178-182.
6. Jung M, Hwang J, Tani J. Multiple spatio-temporal scales neural network for contextual visual recognition of human actions[C]//Development and Learning and Epigenetic Robotics (ICDL-Epirob), 2014 Joint IEEE International Conferences on. IEEE, 2014: 235-241.
7. Zhang N, Paluri M, Ranzato M. A. et al. Panda: Pose aligned networks for deep
8. K. Soomro, A. R. Zamir and M. Shah, UCF101: A Dataset of 101 Human Action Classes From Videos in The Wild. CRCV-TR-12-01, November, 2012.
9. Ji, S., Xu, W., Yang, M., Yu, K.: 3d convolutional neural networks for human action recognition. IEEE T-PAMI 35(1) (2013) 221-231. .
10. Jue Wang, Anoop Cherian, FatihPorikli: Ordered Pooling of Optical Flow Sequences for Action Recognition. In WACV, 2017.
11. Song, S. Lan, C. Xing, J. Zeng and Liu J. 2017. An end-to-end spatial-temporal attention model for human action recognition from skeleton data. In AAAI.

12. Ran, L., Zhang, Y., Zhang, Q., Yang, T.: Convolutional neural network-based robot navigation using uncelebrated spherical images. Sensors 17(6) (2017).
13. Ji, S., Xu, W., Yang, M., Yu, K.: 3d convolutional neural networks for human action recognition. IEEE T-PAMI 35(1) (2013) 221-231.
14. Jue Wang, Anoop Cherian, FatihPorikli: Ordered Pooling of Optical Flow Sequences for Action Recognition. In WACV, 2017.
15. Song, S. Lan, C. Xing, J. Zeng and Liu J. 2017. An end-to-end spatial-temporal attention model for human action recognition from skeleton data. In AAAI.
16. Ran, L., Zhang, Y., Zhang, Q., Yang, T.: Convolutional neural network-based robot navigation using uncelebrated spherical images. Sensors 17(6) (2017).
17. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semanticsegmentation. In: CVPR. (2015) 3431-3440.
18. Feichtenhofer C., Pinz A., and Wildes R. 2016. Spatiotemporal residual networks for video action recognition. In NIPS.
19. Abu-El-Haija, S.; Kothari, N.; Lee, J.; Natsev, P.; Toderici, G.; Varadarajan, B.; Vijayanarasimhan, S. YouTube-8M: A Large-Scale Video Classification Benchmark. arXiv **2016**, arXiv:1609.08675.
20. Minhas, R.A.; Javed, A.; Irtaza, A.; Mahmood, M.T.; Joo, Y.B. Shot classification of field sports videos using AlexNet Convolutional Neural Network. Appl. Sci. **2019**, 9(3), 483, doi:10.3390/app9030483
21. Russo, M.A.; Kurnianggoro, L.; Jo, K.H. Classification of sports videos with combination of deep learning models and transfer learning. In Proceedings of the 2nd International Conference on Electrical, Computer and Communication Engineering, Cox'sBazar, Bangladesh, 7–9 February 2019, doi:10.1109/ECACE.2019.8679371. [CrossRef].
22. Ng, Y.H.J.; Hausknecht, M.; Vijayanarasimhan, S.; Vinyals, O.; Monga, R.; Toderici, G. Beyond short snippets: Deep networks for video classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015.
23. Baccouche, Moez, Franck Mamalet, Christian Wolf, Christophe Garcia, and Atilla Baskurt. "Sequential deep learning for human action recognition." In *International workshop on human behavior understanding*, pp. 29-39. Springer, Berlin, Heidelberg, 2011.
24. Ramanan, Deva. "Learning to parse images of articulated bodies." *Advances in neural information processing systems*. 2007.
25. Soomro, Khurram, and Amir R. Zamir. "Action recognition in realistic sports videos." In *Computer vision in sports*, pp. 181-208. Springer, Cham, 2014.
26. Chaquet, J. M., Carmona, E. J., & Fernández-Caballero, A. (2013). A survey of video datasets for human action and activity recognition. *Computer Vision and Image Understanding*, *117*(6), 633-659.
27. Rafiq, Muhammad, et al. "Scene Classification for Sports Video Summarization Using Transfer Learning." Sensors 20.6 (2020): 1702.
28. He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
29. Wu, Jiaxiang, et al. "Error compensated quantized SGD and its applications to largescale distributed optimization." arXiv preprint arXiv:1806.08054 (2018)