# Convolutional Autoencoder for Vision-Based Human Activity Recognition

Surbhi Jain[1,3], Aishvarya Garg[2,3], Swati Nigam[1,3], Rajiv Singh[1,3(✉)], Anshuman Shastri[3], and Irish Singh[4]

[1] Department of Computer Science, Banasthali Vidyapith, Rajasthan 304022, India
jkrajivsingh@gmail.com
[2] Department of Physical Science, Banasthali Vidyapith, Rajasthan 304022, India
[3] Centre for Artificial Intelligence, Banasthali Vidyapith, Rajasthan 304022, India
anshumanshastri@banasthali.in
[4] Departmenet of Computer Science & Engineering, Oregon Institute of Technology, Oregon, USA
irish.singh@oit.edu

**Abstract.** Human activity recognition (HAR) is a crucial component for many current applications, including those in the healthcare, security, and entertainment sectors. At the current state of the art, deep learning outperforms machine learning with its ability to automatically extract features. Autoencoders (AE) and convolutional neural networks (CNN) are the types of neural networks that are known for their good performance in dimensionality reduction and image classification, respectively. As most of the methods introduced for classification purposes are limited to sensor based methods. This paper mainly focuses on vision based HAR where we present a combination of AE and CNN for the classification of labeled data, in which convolutional AE (conv-AE) is utilized for two functions: dimensionality reduction and feature extraction and CNN is employed for classifying the activities. For the proposed model's implementation, public benchmark datasets KTH and Weizmann are considered, on which we have attained a recognition rate of 96.3%, 94.89% for both, respectively. Comparative analysis is provided for the proposed model for the above-mentioned datasets.

**Keywords:** Human Activity Recognition · Deep Learning · Convolutional Neural Networks · Autoencoders · Convolutional Autoencoders

## 1 Introduction

Over the decades, there has been an increase in the demand for a system that can perform human activity recognition has attracted a significant amount of attention towards HAR. Human activity refers to the movement of one or several parts of the person's body and it is basically determined by kinetic states of an object at different times [1]. HAR is the system that can automatically identify and analyze different human activities from the different body movements by using machine learning and deep learning techniques. HAR provides applications not only in the field of healthcare and surveillance


B. J. Choi et al. (Eds.): IHCI 2023, LNCS 14532, pp. 93–104, 2024.
https://doi.org/10.1007/978-3-031-53830-8_10

but also for human computer interaction. Recent advancements in the field of neural networks allow for automatic feature extraction instead of using hand-crafted features to train HAR models. These models are primarily based on convolutional neural network (CNN), autoencoders (AE), and long short-term memory (LSTM). These methods extract important features from an image without any human supervision and provide a better feature representation of images to recognize objects, classes, and categories.

Generally, most of them follow the steps as shown in Fig. 1. Firstly, there is a requirement of raw data can be collected using sensors and vision-based tools. In sensor based, data is gathered from variety of sensors such as accelerometer, gyroscope, barometer, compass sensor, and other wearable sensors. For vision-based datasets, it uses visual sensing facilities, for example single camera or stereo and infrared to capture activities, and the collected data can be in the form of either videos or images [6]. Then the collected data goes through a pre-processing stage where processes such as segmentation, cropping, resizing is applied, unwanted distortions are suppressed so that the quality of the image is improved. Pre-processed data is then fed to the feature extractor; the process of feature extraction is attained by using either machine learning or deep learning techniques. In machine learning, features are extracted manually which are known as hand crafted features with the help of descriptors such as local binary pattern (LBP), histogram of oriented gradient (HOG), Scale-invariant feature transform (SIFT). In deep learning, features are extracted by using neural networks such as in CNN it is done at convolution layer and pooling layer. And lastly, activities are classified with the help of support vector machines in ML and in DL by utilizing a fully connected layer of CNN.

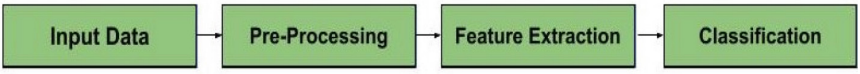



**Fig. 1.** Framework of Human Activity Recognition.

Many approaches have been implemented for recognizing human activities using deep learning model. From the surveys provided by [2–5], we have realized that most work in the field of activity recognition is done on sensor-based models. Hence, one of our objectives is to generate a model for vision based HAR. Going through different types of deep learning algorithms in depth, we came to know that AE can extract unique features along with dimensionality reduction property. So, we employed AE for the classification of activities. We have implemented a small framework in which AE was unable to predict the activities such that we can say that the efficiency of this algorithm degraded when the labeled data as input was given and the output obtained from it had some loss but, on the contradictory, the extracted features were far better than other techniques resulted from the above-mentioned property.

As we know, CNN is known for the best results in classification. Keeping this point in mind, to utilize AE's functionality for labeled data, we proposed a hybrid model for the classification of activities in which we employed two models i.e., Conv-AE and CNN models. In this proposed model, Conv-AE is utilized for feature extraction and dimensional reduction purposes and CNN for classification of activities from two

standard public vision-based datasets: KTH and Weizmann. For the performance analysis on both datasets, we have used a confusion matrix and accuracy for our proposed model.

In this paper, Sect. 2 reviews the existing literature including different deep learning approaches used for vision based HAR. Section 3 provides a brief introduction to the methods used and describes the methodology and tools and techniques that are utilized in the proposed system. Section 4 provides the complete overview of the result and comparative analysis of the system. It also includes the work plan of the proposed model. And lastly Sect. 5 represents the conclusion and the future scope based on the study done for the paper.

## 2 Literature Review

This section reviews the existing literature including different deep learning approaches utilized for vision based HAR for KTH and Weizmann datasets. Several vision-based activity recognition approaches were published in the last two decades. A considerable amount of research has been done on HAR and different techniques of deep learning have been published by the researchers over the decades to determine human activities. It is reviewed on the basis of extracted features which include spatial features, temporal features, and both spatial-temporal features.

Kiruba et al. [7] proposed a work that is based on three different stages. To reduce the computational time and cost, discriminative frames were considered, Volumetric Local Binary Pattern (VLBP) was utilized in the second stage. They discovered that the hexagonal volume local binary pattern (H-VLBP) descriptor outperforms all other novel geometric shape-based neighborhood selection techniques for the identification of human action. To achieve multi-class recognition and to reduce the dimensions, the decoder layer is replaced with the softmax layer of the deep stacked autoencoder (SAE). They were able to obtain an accuracy of 97.6% 98.6% for the KTH and Weizmann datasets, respectively. Nigam et al. [9] proposed a method utilizing background subtraction for human detection and employed it with LBP (local binary pattern) and multiclass-SVM for classification of human activities. Their method was able to achieve an accuracy of 100% for Weizmann dataset.

Ramya et al. [11] proposed a method for HAR utilizing the distance transformation and entropy features of human silhouettes. They tested their method for KTH and Weizmann dataset and were able to obtain an accuracy of 91.4% for KTH and 92.5% for Weizmann dataset. Karuppannan et al. [13] proposed the HAR method in three orthogonal planes (TOP) pyramidal histogram of orientation gradient-TOP (PHOG-TOP) and dense optical flow-TOP (DOF-TOP) were also utilized with a SAE for reducing the dimensions and lastly the output from SAE is fed to LSTM for classification. They were able to achieve an accuracy of 96.50% and 98.70% for the KTH dataset and Weizmann datasets, respectively.

For characteristics representation Gnouma et al. [8] took human actions as a series of silhouettes. Furthermore, a stacked sparse autoencoder (SSA) system is provided for automated human action detection. The updated history of binary motion image (HBMI) is utilized as the first input to softmax classifier (SMC). They evaluated their method on the KTH and Weizmann dataset and obtained an accuracy of 97.83% and 97.66%

respectively. Song et al. [10] proposed the method of feature extraction by using the recurrent neural network (RNN) and the AE. The features of RNN effectively express the behavior characteristics in a complex environment, and it is merged with AE via feature similarity to generate a new feature with greater feature description. The experiments using KTH and Weizmann datasets demonstrate that the technique suggested in the research has a high recognition rate as they achieved an accuracy of 96% and 92.2% for the KTH and Weizmann datasets, respectively.

Mahmoud et al. [12] proposed an end-to-end deep gesture recognition process (E2E-DGRP) which is used for large scale continuous hand gesture recognition using RGB, depth and grey-scale video sequences. They achieved 97.5% and 98.7% accuracy for the KTH and Weizmann dataset, respectively. Garg et al. [14] proposed a model where they used a hybrid model of CNN-LSTM for feature extraction and classification of human actions. On testing their model for KTH and Weizmann datasets they were able to achieve an accuracy of 96.24% and 93.39% on KTH and Weizmann datasets, respectively.

Singh et al. [15] analyzed the real world HAR problem. They proposed a model utilizing discrete wavelet transform and multi-class support vector machine classifier to recognize activities. They used KTH and Weizmann datasets to test their model and obtained a recognition rate of 97% for both the datasets. A deep NN-based approach is suggested by Dwivedi et al. [16] for identifying suspicious human activity. The deep Inception V3 model in this instance extracts the key features for discrimination from images. They feed features into LSTM. The proposed system is tested against eleven benchmark databases, including the KTH action database and the Weizmann datasets. The suggested method had a 98.87% recognition rate.

To extract features, Badhagouni et al. [17] suggested that a CNN classifier be combined with the Honey Badger Algorithm (HBA) and CNN. The projected classifier is used to recognize human behaviors like bending, strolling, and other similar ones. HBA can be used to improve CNN performance by optimizing the weighting parameters. The proposed method was tested using the Weizmann and KTH databases. Saif et al. [18] proposed a convolutional long short-term deep network (CLSTDN) consisting of CNN and RNN. In this method CNN uses Inception-ResNet-v2 to classify by extracting spatial features and lastly RNN uses Long Short-Term Memory (LSTM) for prediction based on temporal features. This method achieved an accuracy of 90.1% on KTH dataset.

## 3 Proposed Methodology

We have proposed a hybrid model using convolutional type of autoencoder (Conv-AE) and CNN in which Conv-AE is utilized for two objectives- one is for dimensionality reduction and the other is for extracting features, and CNN model is utilized for the classification of activities by using vision-based datasets i.e., KTH and Weizmann.

### 3.1 Convolutional Autoencoder (Conv-AE)

An autoencoder is a feed-forward neural network that has the same output as input and is based on unsupervised learning technique. As shown in Fig. 2 input layer's size is the same as the size of output layer. They consist of three main layers- an encoder, a code,

and lastly a decoder. Latent space representation is the output obtained from the encoder layer after compression. It encodes the input image in a compressed form into a reduced dimension. The compressed image obtained from the encoder layer will be a distorted form of the original input image. The second layer is known as the code layer which is the compressed input going to be fed to the decoder.

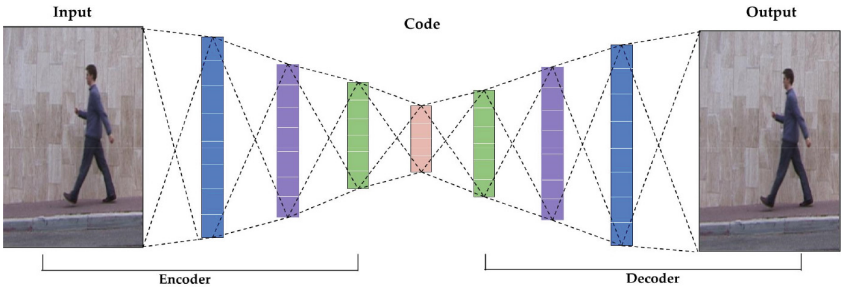



**Fig. 2.** Basic Architecture for the Autoencoder (AE).

And lastly, the decoder layer decodes the input compressed image back to its original dimensions. The decoded form of image is regenerated from latent space representation, and it's the reconstruction of the original image consisting of some loss. There are different types of AE and for our proposed method we are going to use Conv-AE which is same as the simple AE, as it is used for dimensionality reduction and it also has same number of input and output layers, the only difference is that the network of encoder layers converts into a network of convolutional layers and the decoder layer changes into a transpose of convolutional layers.

### 3.2 Convolutional Neural Network (CNN)

Convolutional neural network is a type of feed-forward neural network commonly utilized for extracting features and classifying images. As you can see in Fig. 3 it is made up of three layers namely convolutional layer, pooling layer, and fully connected layer. At the convolutional layer, convolution operations are performed on the passed image. The output of this layer gives us information about corners and edges of an image. This operation reduces the size of the numerical presentation so that only the important features are sent further for image classification which helps in improving accuracy. Then they obtained feature map is provided to a pooling layer as input which further helps in reducing the size of the feature map which forces the network to identify key features in the image. The extracted key features from the pooling layer are then passed through a flattening layer which converts the pooled feature map into a single column known as vectors. The column of vectors is then passed to the fully connected layer which consists of weights, biases, and neurons. Here, the process of classification takes place as the vectors go through a few more fully connected layers. And to resolve the problem of over-fitting batch normalization is used. And, lastly the output from fully connected

layers passes through a softmax activation function which computes the probability distribution and gives the most probabilistic output as to which input image belongs to which class [19].

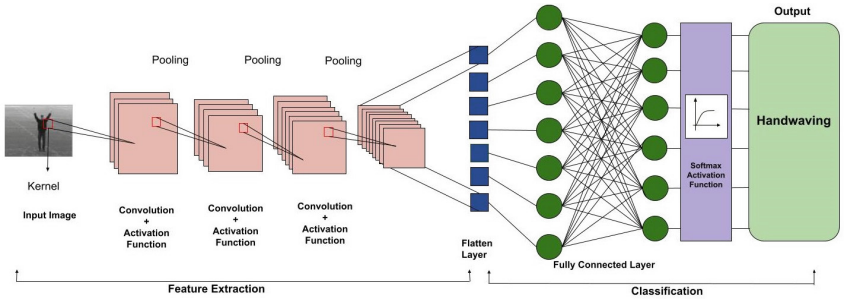



**Fig. 3.** General Architecture of the Convolutional Neural Network (CNN).

As we know that, Conv-AE works best for classifying the images for unlabeled data which makes it difficult to predict the data for labeled classes but because of its ability of dimensionality reduction, we are going to employ it with CNN where Conv-AE is utilized for feature extraction and dimensionality reduction purposes and 2D-CNN is used for classification.

### 3.3 Pre-processing

For the input to the model, pre-processing is done on the datasets. Operations such as segmentation is performed to convert into frames from videos at the given rate of frames per second, normalization is implemented to transform the dataset's value into a common scale, resizing converts the input shape of the image from $120 \times 260 \times 3$ to $100 \times 140 \times 1$ for the KTH dataset and $144 \times 180 \times 1$ for the Weizmann dataset.

### 3.4 Feature Representation

To perform the process of feature extraction, we have taken 3 convolutional layers for the encoder and the number of filters for the same are 64, 32, 16. The kernel's size is the same for all the layers that is (3, 3) with strides of size (1, 1) and the padding used is kept valid for the model. We have also added a pooling layer, more specifically a max-pooling layer of size (2, 2) in all Conv2D layers, as you can see from Fig. 4. Increasing the depth by adding number of layers not only helps with better representation of features but also causes the problem of over-fitting, to overcome that a layer of batch normalization is applied to the first and third layer of Conv2D of the encoder and the loss continues decreasing.

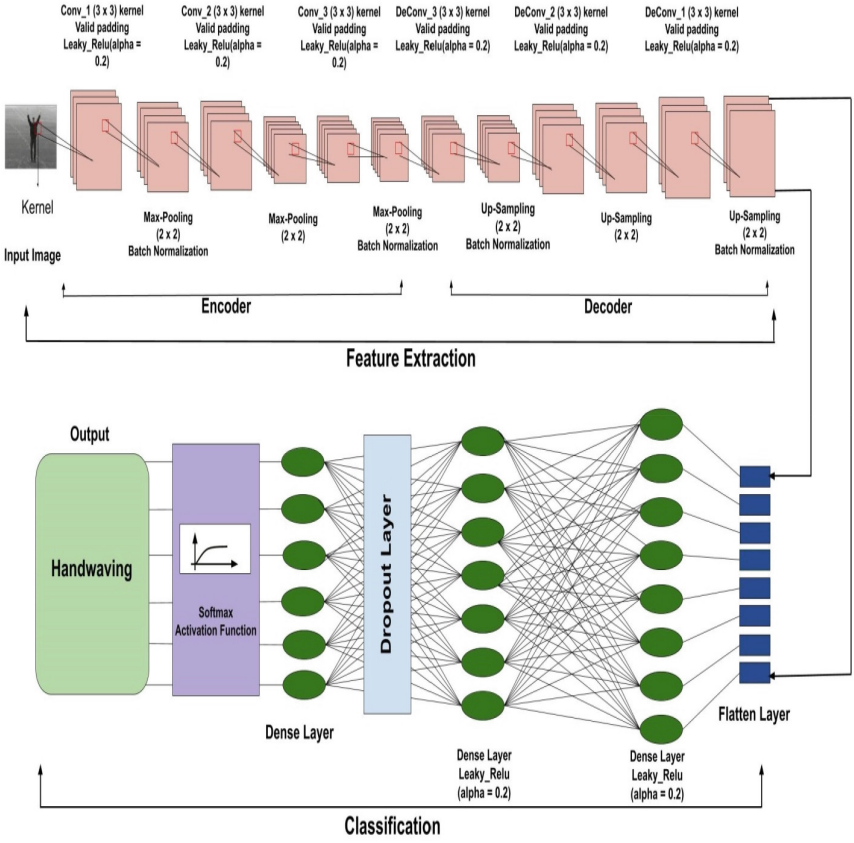



**Fig. 4.** Framework of the proposed method for Human Activity Recognition.

For the decoder, the same number of layers with exact number of filters are applied, but in transposed order of encoder, as the decoder is just the transpose of the encoder, the only difference is that instead of using a max-pooling layer we have used an up-sampling layer of size (2,2) between the three Conv2D layers. Lastly the output from the decoder is passed through a Conv2D layer of filter size 1 which represents present channels in the image. For all the layers an activation function is used known as leaky relu with the slope value of 0.2.

### 3.5 Classification

The 2-dimensional output from the decoder can't be a direct input to the fully connected layer, it is then transformed into a column of vectors by passing through a flatten layer so that it can be fed to the CNN model. For fully connected layer we have employed 3 dense layers with filter size of 32, 16 and last one has the number of filters same as the number of classes of dataset, in between we have also utilized a dropout layer to reduce over-fitting, dropping 20% of the neurons. And lastly there's a softmax activation function

which provides us with the output based on probability and classifies the activity based on their classes. As you can see from diagram 4 an input image of KTH dataset is given to the model, and it provides hand-waving as the highest probability.

## 4 Experimental Results

### 4.1 Datasets

For data processing, initially the frames are extracted from the videos to provide to the model as input. Our proposed model is implemented on the vision based standard public benchmark datasets i.e., KTH and Weizmann. KTH dataset is a recorded video database consists of videos which are 600 in number of 6 activities named as boxing, handclapping, hand-waving, jogging, running and lastly walking that is performed by 25 people in four different environments with different angle, illumination conditions, clothing and gender and having frame rate of 25 frames per second (fps). The dimension of each video in the database is 120 x 160 x 3. We have used all 6 activities for training and classification [20].


**Fig. 5.** Still images consisting of different actions and viewpoints from the KTH and Weizmann dataset.

Weizmann dataset consists of 90 videos containing 10 different activities such as bend, gallop sideways, jump, jump in place, jumping jack, one hand wave, two hands wave, run, skip, walk performed by 9 different subjects with 30 frames per second frame rate and having dimension of each video is 144 x 180 x 3, all 10 activities are considered for classification purposes [21] as one can see in Fig. 5.

For the implementation of the model, we have utilized the Jupyter notebook environment of Google Colaboratory with Keras Deep Learning API. The model was compiled with the help of categorical cross-entropy and for optimizer RMS-prop was used. Dividing the data in the ratio of 2:8 for testing and training for both the datasets where the random state is kept zero and shuffle is set to true. Table 1 gives the idea about different hyper-parameters used for the KTH and Weizmann dataset.

**Table 1.** Hyper-parameters used in the model for KTH and Weizmann dataset.

| Hyper-parameters | KTH dataset | Weizmann Dataset |
|---|---|---|
| Videos | 600 | 90 |
| Epochs | 30 | 35 |
| Batch size | 15 | 5 |
| Train-test ratio | 8:2 | 8:2 |
| Frames per class | 50 | 50 |
| Total number of frames | 30000 | 4500 |
| Dimension of video | 120 x 160 x 3 | 144 x 180 x 3 |

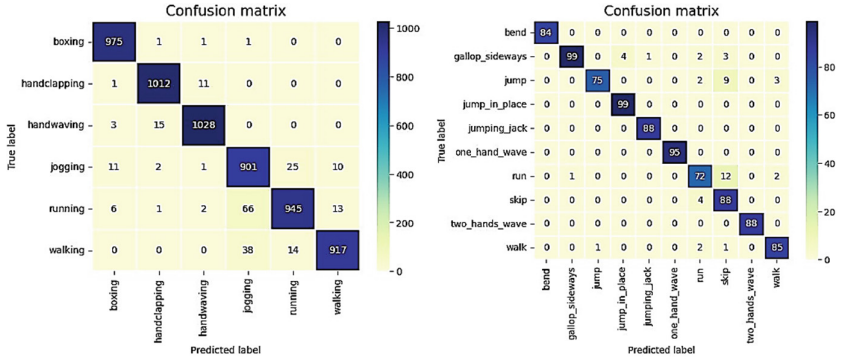
### 4.2 Results and Discussions

During implementation on Google Colaboratory, we faced the problem of crashing of model during the execution time when 60 frames were taken from each video in case of KTH dataset. The accuracy varied when the model was trained on different compute engine backend such as TPU and GPU of Google Colaboratory and more time was taken for the implementation. But TPU obtained the best results with foremost training accuracy and prediction rate. We have implemented this model with varying hyper-parameters such as train and test ratio, batch-size, epochs, resizing the frames, number of layers, optimizers, and filter size to achieve the best prediction accuracy rate. When the data was divided into 9:1 or 7:3 train test ratio it had a visible effect on the accuracy as the model was unable to predict correctly. We also trained our model between a range of 5 to 40 epochs for both datasets and with different batch sizes. But the most accurate result was obtained when we set the epochs at 30 and 35 with batch size 15 and 5 for the KTH and Weizmann datasets, respectively.

Relu activation function was considered to employ in the model as the time taken to learn by relu is less and is computationally less expensive than other activation functions, but it kept causing a problem of dying relu that is when many relu neurons only output values of 0. That's why instead of utilizing relu we implemented leaky relu activation function.

Initially we implemented the model containing only two Conv-2D layers, but the results were better with three layers and only two batch normalization layers in between to avoid over-fitting, as we know the increasing number of layers is directly proportional to the better classification of activities if the problem of over-fitting is kept in consideration. RMS-prop was decided to use as an optimizer when implementing the model with Adam and Stochastic Gradient Descent (SGD) optimizers didn't have any effect on the model's accuracy. As the amount of data given as input to the model also has an effect on the prediction accuracy, we provided a varying range of data to the model and tested it, we came to know that increasing the data lead to an increase of 2.5% in the accuracy, resizing was also done on the frames to make it easier for the model to extract features and to load and execute the program faster. Confusion matrix and accuracy were utilized for the evaluation of the proposed model. We have also provided comparative analysis

for the KTH and the Weizmann datasets. The confusion matrix is also formulated with the graph as the performance matrix for both datasets as shown in Fig. 6 (a), (b). Our proposed model's effectiveness is analyzed by comparing the highest accuracy of our model with the other methodologies on both datasets as shown in Table 2.




(a) (b)

**Fig. 6.** Confusion matrix of (a) KTH dataset and (b) Weizmann dataset for proposed model

**Table 2.** Comparative analysis between obtained accuracy (%) for KTH and Weizmann dataset.

| Author | Year | Methodology | KTH | Weizmann |
|---|---|---|---|---|
| [22] | 2016 | MI-ULBP | 77.16 | 75.66 |
| [23] | 2019 | Fusion of heterogeneous features + SMO + SVM | 92.28 | 91.69 |
| [24] | 2020 | SVM + ANN | 87.57 | 86 |
| [11] | 2021 | Distance Transform + Entropy Features + ANN | 91.4 | 92.5 |
| [25] | 2022 | Fuzzy Clustering Model | 92.90 | 91.41 |
| [26] | 2022 | CAE-DMD | - | 91.1 |
| [14] | 2022 | CNN + LSTM | 96.24 | 93.39 |
| [18] | 2023 | CLSTDN | 90.1 | - |
| **-** | **2023** | **Conv-AE + CNN** | **96.3** | **94.89** |

## 5 Conclusions and Future Scope

This work mainly focuses on vision based HAR with the help of deep learning algorithms. We have employed deep learning through convolutional autoencoders to extract features from the input data. And then a convolutional neural network is implemented to classify the activities according to their respective classes. The proposed model improves the quality of extracted features which then provides better classification. On implementing these algorithms separately we realized that CNN performed quite efficiently in predicting the labeled dataset whereas AE provided better features. Therefore, we proposed

a hybrid HAR model of AE and CNN algorithms for classifying the human activities. In this paper, Conv-AE is employed for the purposes of dimensionality reduction and feature extraction whereas CNN was used for the classification of the human activities. For the testing of the proposed model, vision based public benchmark datasets namely KTH and Weizmann were utilized. We have implemented proposed methodology with varying hyper-parameters from which we were able to obtain an accuracy of 96.3% and 94.89%for the KTH dataset and Weizmann dataset, respectively. From the given comparative analysis, our model was able to perform well in comparison to other state of the art methods and enhances the performances of HAR methods. A HAR system using deep learning techniques needs better feature representation, so fusion techniques for getting hybrid features with better resolution images can give us better results. Other deep learning techniques instead of CNN can be combined with the various feature descriptors to explore better accuracy results.

## References


1. Basly, H., Ouarda, W., Sayadi, F.E., Ouni, B., Alimi, A.M.: CNN-SVM Learning Approach based Human Activity Recognition, pp. 271–281. ICISP, Springer (2020)
2. Bouchabou, D., Nguyen, S.M., Lohr, C., LeDuc, B., Kanellos, I.: A survey of human activity recognition in smart homes based on IoT sensors algorithms: taxonomies, challenges, and opportunities with deep learning. Sensors, MDPI **21**, 6037 (2021)
3. Zhang, S., et al.: Deep learning in human activity recognition with wearable sensors: a review on advances. Sensors, MDPI **4**, 1476 (2022)
4. Alo, U.R., Nweke, H.F., The, Y.W., Murtaza, G.: Smartphone motion sensor-based complex human activity identification using deep stacked autoencoder algorithm for enhanced smart healthcare system. Sensors, MDPI **20**, 6300 (2020)
5. Gu, F., Khoshelham, K., Valaee, S., Shang, J., Zhang, R.: Locomotion activity recognition using stacked denoising autoencoders. IEEE Internet of Things Journal, IEEE **5**, 2085–2093 (2018)
6. Sunny, J.T., et al.: Applications and challenges of human activity recognition using sensors in a smart environment. IJIRST Int. J. Innov. Res. Sci. Technol **2**, 50–57 (2015)
7. Kiruba, K., Shiloah, E.D., Sunil, R.R.C.: Hexagonal Volume Local Binary Pattern (H-VLBP) with Deep Stacked Autoencoder for Human Action Recognition. Cognitive Systems Research, Elsevier **58**, 71–93 (2019)
8. Gnouma, M., Ladjailia, A., Ejbali, R., Zaied, M.: Stacked sparse autoencoder and history of binary motion image for human activity recognition. Multimedia Tools and Applications, Springer **78**, 2157–2179 (2019)
9. Nigam, S., Singh, R., Singh, M.K., Singh, V.K.: Multiview human activity recognition using uniform rotation invariant local binary patterns. J. Ambient Intell. Humani. Comp. Springer, 1–19 (2022)
10. Song, X., Zhou, H., Liu, G.: Human behavior recognition based on multi-feature fusion of image. Cluster Computing, Springer **22**, 9113–9121 (2019)
11. Ramya, P., Rajeswari, R.: Human action recognition using distance transform and entropy based features. Multimedia Tools and Applications, Springer **80**, 8147–8173 (2021)
12. Mahmoud, R., Belgacem, S., Omri, M.N.: Towards an end-to-end Isolated and continuous deep gesture recognition process. Neural Computing and Applications, Springer **34**, 13713–13732 (2022)

13. Karuppannan, K., Darmanayagam, S.E., Cyril, S.R.R.: Human action recognition using fusion-based discriminative features and long short term memory classification. Concurrency and Computation: Practice and Experience, Wiley Online Library **34**, e7250 (2022)
14. Garg, A., Nigam, S., Singh, R.: Vision based Human Activity Recognition using Hybrid Deep Learning. CSI, IEEE, 1–6 (2022)
15. Singh, R., Nigam, S., Singh, A.K., Elhoseny, M.: Wavelets for Activity Recognition. Intelligent Wavelet Based Techniques for Advanced Multimedia Applications, Springer **10**, 109–121 (2020)
16. Dwivedi, N., Singh, D.K., Kushwaha, D.S.: A Novel Approach for Suspicious Activity Detection with Deep Learning. Multimedia Tools and Applications, pp. 1–24. Springer (2023)
17. Badhagouni, S.K., ViswanadhaRaju, S.: HBA optimized Efficient CNN in Human Activity Recognition. The Imaging Science Journal, Taylor & Francis **71**, 66–81 (2023)
18. Saif, A.S., Wollega, E.D., Kalevela, S.A.: Spatio-temporal features based human action recognition using convolutional long short-term deep neural network. Int. J. Adv. Comp. Sci. Appl. Sci. Info. (SAI) Organization Limited **14**, 66–81 (2023)
19. https://towardsdatascience.com/acomprehensive-guide-to-convolutional-neural-networks-the-eli5-way3bd2b1164a53/
20. Schuldt, C., Laptev, I., Caputo, B.: Recognizing Human Actions: A Local SVM Approach. ICPR, IEEE **3**, 32–36 (2004)
21. Blank, M., Gorelick, L., Shechtman, E., Irani, M., Basri, R.: Actions as Space-time Shapes. ICCV, IEEE **2**, 1395–1402 (2005)
22. Nigam, S., Khare, A.: Integration of moment invariants and uniform local binary patterns for human activity recognition in video sequences. Multimedia Tools and Applications, Springer **75**, 17303–17332 (2016)
23. Naveed, H., Khan, G.A.U., Siddiqi, A., Khan, M.U.G.: Human activity recognition using mixture of heterogeneous features and sequential minimal optimization. International Journal of Machine Learning and Cybernetics, Springer **10**, 2329–2340 (2019)
24. Nadeem, A., Jalal, A., Kim, K.: Human Actions Tracking and Recognition based on Body Parts Detection via Artificial Neural Network. ICACS, IEEE, pp. 1–6 (2020)
25. Song, B.: Application of Fuzzy Clustering Model in the Classification of Sports Training Movements. Computational Intelligence and Neuroscience, Hindawi, 2022 (2022)
26. Haq, I.U., Iwata, T., Kawahara, Y.: Dynamic mode decomposition via convolutional autoencoders for dynamics modeling in videos. Comput. Vis. Image Underst. **216**, 103355 (2022)