Journal of Harbin Engineering University
ISSN: 1006-7043

Vol 45 No. 8
August 2024

Advanced Human Activity Recognition With Enhanced Convolutional Neural Networks

Vani E, Supriya[1], Narayanadas[2] Arpitha[3], Arupulla Mamatha[4]

[1]Department of Information Science and Engineering, Nitte Meenakshi Institute of Technology, Bengaluru, INDIA

[2]Department of Information Science and Engineering, Nitte Meenakshi Institute of Technology, Bengaluru, INDIA

[3]Department of Computer Science and Engineering, Sree Dattha Institute Of Engineering And Science, Hyderabad, INDIA

[4]Department of Computer Science and Engineering, Sree Dattha Institute Of Engineering And Science, Hyderabad, INDIA

Abstract: Human Activity Recognition (HAR) serves as a pivotal function with implications spanning from healthcare monitoring to security systems. Recent advancements in Machine Learning (ML) alongside Computer Vision techniques have demonstrated considerable progress in automating this task. This paper offers a detailed review and analysis of diverse ML algorithms and Computer Vision methods used in HAR systems. We explore the challenges encountered in this field, such as variability in human actions, occlusion, and changes in perspective, and examine how various methodologies mitigate these issues. Moreover, we spotlight principal datasets employed for training and evaluation. Through a thorough empirical analysis, we assess the performance of various ML models in precisely identifying human activities from sensor data or video feeds. Our observations affirm the effectiveness of deep learning frameworks, especially Convolutional Neural Networks (CNNs), in detecting complex spatiotemporal patterns essential for HAR endeavors. Additionally, we explore forthcoming trends, ongoing challenges, and future avenues for research in this evolving area, highlighting the promise for continued progress through joint efforts among the ML and Computer Vision communities.

Keywords: Human Activity, Machine Learning, Deep Learning CNN,VGG16

I Introduction

Human Activity Recognition (HAR) is a complex task in the realm of time series classification that entails predicting a person's movements based on sensor inputs. Traditionally requiring deep expertise in signal processing, this task involves the meticulous crafting of features from raw data to develop a suitable machine learning model. However, advancements in deep learning, particularly the use of CNN and recurrent neural networks (RNNs) like long short-term memory networks (LSTMs), have proven effective. These methods can autonomously extract features from raw sensor data and achieve state-of-the-art results in activity prediction. HAR often occurs in indoor settings and involves common actions such as walking, standing, or more specific activities like those in kitchens or industrial environments. The sensors, which might be embedded in smartphones or worn as part of fitness trackers, capture data like acceleration and rotation. In contemporary times, the widespread availability of smartphones and fitness devices has made collecting such sensor data more feasible and cost-effective, enhancing the study of HAR as a prevalent area of research. The task itself is framed as either a univariate or multivariate time series classification challenge, where the primary goal is to determine the activity from a snapshot of sensor data. This is often achieved by segmenting continuous data streams into smaller chunks or windows using a sliding window technique, which are then classified into broad activity categories. The challenge lies in the significant variability in how different individuals perform the same activity, which can affect the consistency of the sensor data, making accurate activity recognition a demanding yet critical task.

Ii. State Of The Art

Transfer learning is indeed a fascinating aspect of machine learning. It's like leveraging previously

acquired knowledge or expertise to tackle new, similar tasks. By transferring knowledge from one domain to another, it often reduces the need for extensive labeled data or computational resources, making the learning process more efficient. It's akin to how humans learn; we use our past experiences to tackle new challenges or learn new skills. This concept has parallels in the psychological concept of learning transfer, though direct connections between these fields are minimal. The reuse or transfer of knowledge from previously mastered tasks to novel tasks can significantly enhance learning efficiency. It involves the adaptation of a pre-trained model to a new, albeit similar, problem. This approach has become exceedingly popular in deep learning, as it allows for the training of robust neural networks with relatively limited data—a common scenario in many real-world applications where massive labeled datasets are scarce.

In this process, the knowledge from a machine learning model that has been trained on one task is applied to a different, albeit related, task. For example, a model trained to classify images as containing a backpack could be repurposed to recognize other items like sunglasses, utilizing the learned features from the original training. Transfer learning essentially leverages the learnings from one task to enhance generalization in another, by transferring the learned weights of a network from "task A" to "task B."

The principle behind this is to harness the knowledge a model has acquired from a task with abundant labeled data and apply it to a new task that lacks extensive data. Instead of initiating the learning process from zero, transfer learning starts with patterns that have already been learned from solving a related task.

Predominantly utilized in fields like computer vision and natural language processing, where computational demands are high, transfer learning is more a methodological framework within machine learning rather than a standalone technique. It is often paired with neural networks, which require substantial amounts of data .

In the context of Human Activity Recognition (HAR), transfer learning is employed by leveraging pre-trained deep learning models to recognize and classify human activities based on sensor data. This allows researchers to benefit from knowledge initially derived from domains like image classification and apply it to activity recognition, even when there is limited labeled data available in the target domain. To focus on addressing the Human Activity Recognition (HAR) problem using hybrid deep learning (DL) models, combining Convolutional Neural Networks (CNNs) with various types of Recurrent Neural Networks (RNNs). The results suggest that the hybrid models outperform both CNNs and RNNs individually in terms of accuracy. Moreover, the evaluation considers other metrics such as precision, recall, F-score, sensitivity, and specificity, providing a comprehensive assessment of classification performance.

Saedeh Abbaspour et al.[1] have apply four hybrid DL models to the HAR problem. Each hybrid model integrates a CNN with a variant of RNNs. A well-known and publicly available dataset (i.e., PAMAP2) is used to evaluate the performance of the proposed hybrid models. The analysis results indicate a high level of accuracy for each model, which is higher than the accuracy achieved by using either CNNs or RNNs individually. In addition to accuracy, precision, recall, F-score, sensitivity, and specificity are other measures that are evaluated on the classification results. Overall, the results indicate that the models including Bi-directional RNNs perform better than the ones based on uni-directional RNNs. This outcome is reasonable due to the fact that in the former, the data are processed both from past to future and from future to past. However, this advantage comes with the cost of more computational time.

J.K. Aggarwal et al[2]. summarizes the major techniques in human activity recognition from 3D data with a focus on techniques that use depth data. Broad categories of algorithms are identified based upon the use of different features. The pros and cons of the algorithms in each category are analyzed and the possible direction of future research is indicated. As the recent development of range sensing technology progresses, we have easy access to the 3D data as a complement to the traditional RGB imagery. Acquiring 3D data from depth sensors is more convenient than estimating it from stereo images or using motion capture systems. This is an important cornerstone in computer vision, as the information lost in

projection from 3D to 2D in the traditional intensity image may partially be recovered from the sensor. Luay Alawneh et al. [3] investigate the benefits of time series data augmentation in improving the accuracy of several deep learning models on human activity data gathered from mobile phone accelerometers. More specifically, we compare the perfor-mance of the Vanilla, Long-Short Term Memory, and Gated Recurrent Units neural network models on three open-source datasets. We use two time series data augmentation techniques and study their impact on the accuracy of the target models. The experiments show that using gated recurrent units achieves the best results in terms of accuracy and training time fol-lowed by the long-short term memory technique. Furthermore, the results show that using data augmentation significantly enhances recognition quality. Mohammad Abu Alsheikh et al.[4] shows that deep activity recognition models (a) provide better recognition accuracy of human activities, (b) avoid the expensive design of handcrafted features in existing systems, and (c) utilize the massive unlabeled acceleration samples for unsupervised feature extraction. Moreover, a hybrid approach of deep learning and hidden Markov models (DL-HMM) is presented for sequential activity recognition. This hybrid approach integrates the hierarchical representations of deep activity recognition models with the stochastic modeling of temporal sequences in the hidden Markov models. We show substantial recognition improvement on real world datasets over state-of-the-art methods of human activity recognition using triaxial accelerometers.

Iii. The Proposed Method

Problem statement

The main aim of the project is to prediction the human activity from the input videos using Machine Learning with Computer vision. Human Activity Recognition (HAR) aims to automatically identify and classify human actions from sensor data or video streams. While traditional approaches have made significant strides, they often struggle with the inherent complexity and variability of human movements, especially in real-world scenarios characterized by occlusion, viewpoint changes, and environmental clutter. To address these challenges, this research focuses on leveraging the power of Machine Learning (ML) techniques, particularly those rooted in Computer Vision, to develop robust and accurate HAR systems.

The objectives of the project are:

1) Study and apply the needed tools namely:

a) Video should be downloaded or locally saved.

b) Flask Deployed Server with python 3.11 Community

c) Algorithms for computer vision and machine learning.

2) Develop a front-end website to upload video to process.

3) Video gets process and generate the result

4) Recognition of human activity.

Human Activity Recognition (HAR) is indeed a vital field with wide-ranging applications. By automatically identifying and categorizing human actions or behaviors, HAR systems enable numerous practical applications across different domains. In healthcare monitoring, HAR can assist in tracking patient movements and activities, helping healthcare providers monitor patient well-being and adherence to treatment plans. In fitness tracking, HAR algorithms can analyze exercise routines and provide feedback to users, facilitating personalized training programs. Overall, HAR plays a crucial role in improving efficiency, safety, and user experience across various domains by automatically analyzing and understanding human activities. The block diagram human activity recognition is as shown below figure.

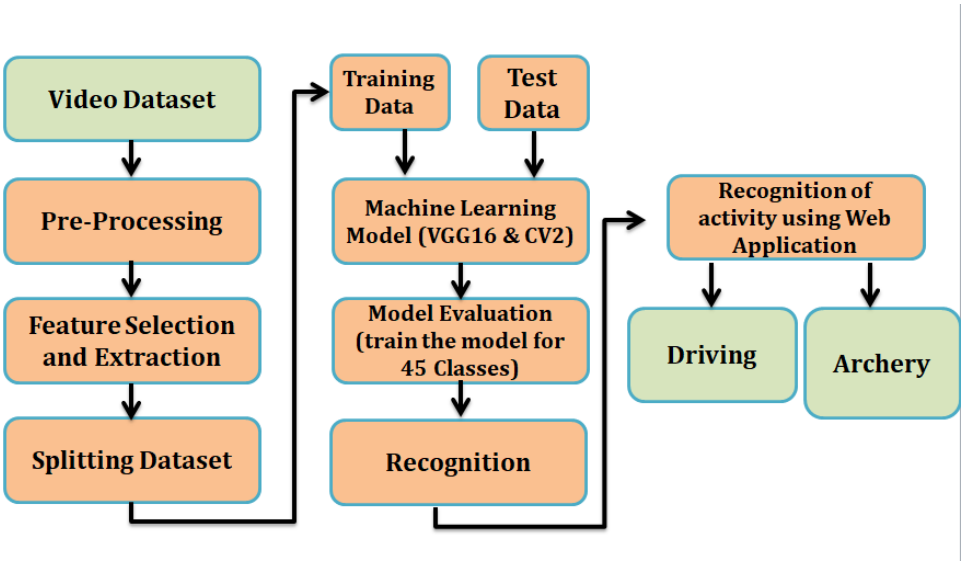

Figure 1: Human Activity Recognition Architecture Diagram

Data Collection: Here, we are collected the video dataset of different human activities. Later, using our deep learning model video is converted into number of frame images. This essential initial step involves gathering sensor readings or video footage that capture human movements across various settings and environments. This phase is vital as high-quality, representative data sets are crucial for developing precise and robust models for activity recognition.

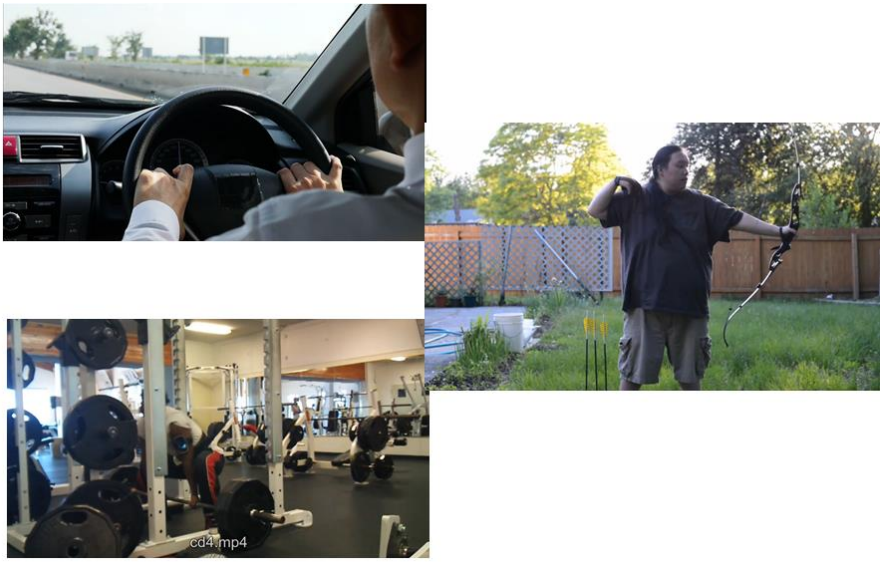

Figure 2: Video sample Dataset

Data Preprocessing: This stage focuses on cleaning raw sensor data by eliminating noise, filtering anomalies, and normalizing data formats. The cleaned data is then ready for the next phase of feature extraction.

Feature Extraction: In this phase, pertinent features are derived from the pre-processed sensor data. This might involve applying algorithms to calculate statistical, temporal, spatial, or frequency-based attributes, tailored to the data's specific properties.

Machine Learning: This component is responsible for the classification of activities using algorithms that have been trained on datasets enriched with features. It encompasses techniques for supervised, unsupervised, or semi-supervised learning that can identify various activities from the features extracted. This stage also includes the

integration of machine learning models with computer vision capabilities.

User Interface (UI): This element of the system provides an intuitive interface for users to interact with the HAR system. It may feature tools for visualizing data, monitoring activities, and adjusting system settings.

Practical Implementations

- **Importing Packages:** Import the required libraries for the project. Here, we require Numpy, Tensorflow, keras, OpenCV and so on
- **Upload the Model H5 File :** In this project, we are VGG16 and ImageNet Model. Based on the performance of the algorithm. ImageNet achieved 95% of the accuracy. So we saved the best model on H5 file. This file, will be used for deployment in the web application.
- **Function: Create Number of Classes:** the numbers of classes or physical activity labels are created. The goal of this project is to recognize the human physical activity, so we are 45 labels of different human activity.
- **Read the Video:** Here we using OpenCV to read the input video from the given path.
- **Covert the Video and Image :**In this section, the model covert the input video into number of image frames. These frames will be stored in a separate folder i.e., data.
- **Result: Recognition of Human Activity:** All the prediction output will be saved in one folder called result.

Algorithms

a. VGG16

A convolutional neural network, also referred to as a ConvNet, is a form of artificial neural network used primarily for processing structured array data such as images. A typical convolutional neural network consists of an input layer, an output layer, and several hidden layers which may include convolutional layers, pooling layers, fully connected layers, and normalization layers. The VGG16 model, a specific type of CNN, is recognized as one of the most effective models for image recognition tasks. Developed by researchers, this model leverages an architecture that deepens the network through the use of very small (3x3) convolutional filters. This configuration allows for a significant enhancement over previous models, pushing the network to have between 16 and 19 layers of weights, summing up to approximately 138 million trainable parameters.

VGG16 Architecture : This is primarily a convolutional neural network (CNN) architecture designed for image classification rather than object detection. It was developed by the Visual Geometry Group (VGG) at the University of Oxford. The VGG16 model consists of 16 layers, including 13 convolutional layers and 3 fully connected layers. While VGG16 is indeed popular for image classification tasks due to its simplicity and effectiveness, achieving high accuracy on benchmark datasets like ImageNet, it is not specifically designed for object detection. Object detection involves not only classifying objects within an image but also locating their positions by drawing bounding boxes around them.

Weight Layers: In VGG16, the numeral "16" refers to the number of layers containing learnable parameters. VGG16 includes 13 convolutional layers, five max pooling layers, and three dense layers, adding up to 21 layers, but only 16 are layers with adjustable parameters.

Input Specifications: VGG16 accepts input images with dimensions of 224x224 pixels and three color channels (RGB).

Unique Characteristics: A notable aspect of VGG16 is its preference for simplicity, using small 3x3 filters with a stride of 1 throughout the convolutional layers, maintaining consistent padding and 2x2 filters with a stride of 2 in the max pooling layers.

Consistent Layer Arrangement: The convolutional and max pooling layers are systematically arranged across the network's architecture.

Filter Sizes: The convolutional layers in VGG16 vary in the number of filters: Conv-1 layer has 64 filters, Conv-2 contains 128 filters, Conv-3 consists of 256 filters, and both Conv-4 and Conv-5 layers have 512 filters each.

Fully Connected Layers: Following the convolutional layers, three fully connected (dense) layers are present. The first two dense layers each have 4096 neurons, while the third performs a 1000-class ILSVRC classification, thus containing 1000 output neurons (one for each class). The last layer in the network is a softmax layer for producing final probability distributions.

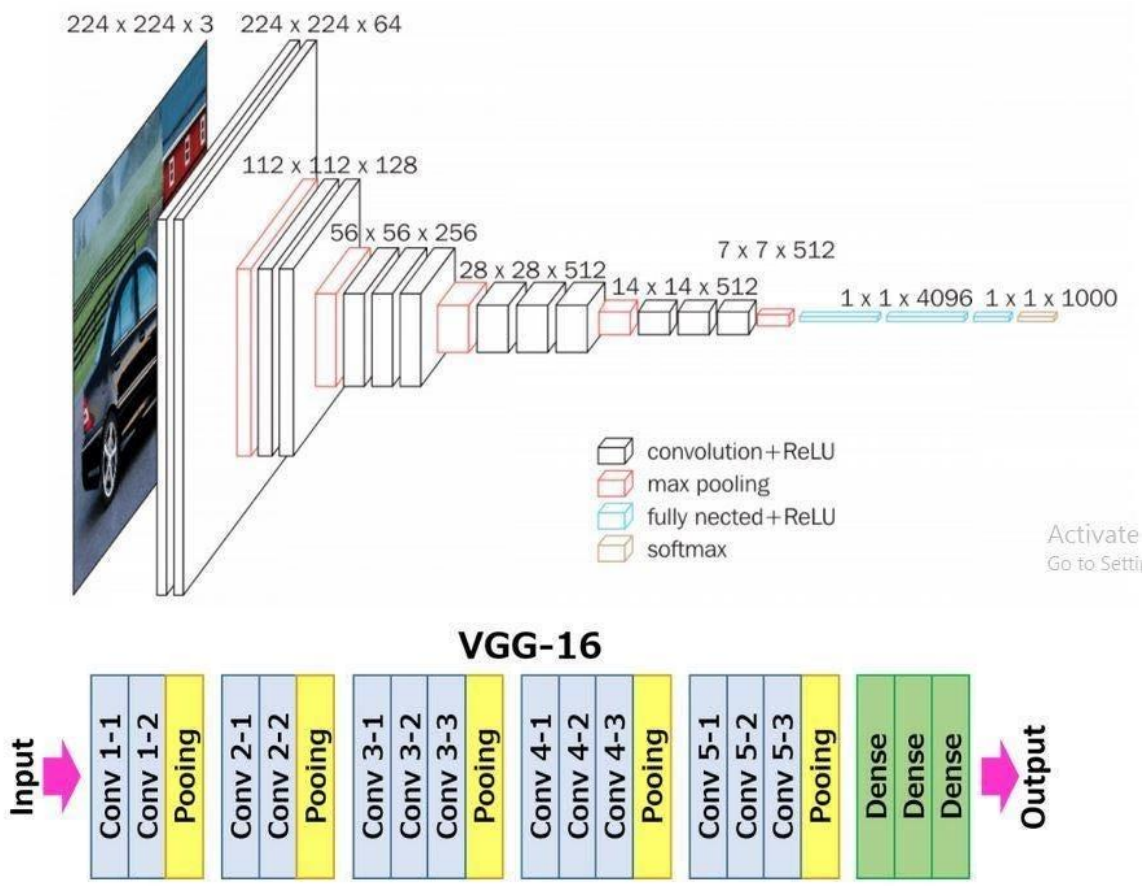

Figure 3: VGG16 Architecture

b. Open-CV

OpenCV (Open-Source Computer Vision) stands as an open-source collection of computer vision and machine learning algorithms. This library offers a diverse array of functionalities and utilities for processing images and videos, encompassing numerous techniques such as image manipulation, feature extraction, object detection, and beyond. While primarily implemented in C++, OpenCV also boasts interfaces for Python, Java, and several other programming languages.

The operation of OpenCV (Open-Source Computer Vision) involves a sequence of steps to execute diverse computer vision tasks. Here's a broad overview of OpenCV's functioning:

Library Integration: Incorporating OpenCV into your code necessitates importing the library into your project. This entails including the relevant header files and linking essential libraries depending on the programming language employed.

Image/Video Loading and Manipulation: OpenCV furnishes functions to read and load images or videos from files or capture devices like webcams. Post-loading, various operations can be performed on the data, encompassing pre-processing, resizing, cropping, and color space conversions.

Image Processing Operations: OpenCV presents a vast array of image processing functions. Filters such as blurring, sharpening, and edge detection can be applied. Additionally, transformations like rotation, scaling, and perspective correction are available. Histogram analysis, morphological operations, and feature detection are also supported.

Object Detection and Tracking: OpenCV encompasses pre-trained models and algorithms for object detection and tracking. Techniques like Haar cascades, HOG, and deep learning-based models (e.g., YOLO, SSD) can be employed. These algorithms identify objects in images or videos and track them across frames.

Feature Detection and Description: OpenCV includes functions for detecting and describing image features like corners, blobs, and edges. Well-known algorithms such as SIFT and SURF are implemented. These features facilitate tasks such

as image matching, object recognition, and 3D reconstruction.

Integration with Machine Learning and Deep Learning: OpenCV seamlessly integrates with machine learning and deep learning frameworks such as TensorFlow and PyTorch. Models can be trained for various computer vision tasks using these frameworks and deployed within OpenCV. This enables tasks like object classification, image segmentation, and image generation.

Iv. Experimental Results

In this experiment , we have used the concept of Human Activity Recognition (HAR) using Convolutional Neural Networks (CNNs) involves identifying and classifying physical activities from data, typically obtained from sensors or video. Initially Make a training file and validation file with 50 categories and 15 videos of each class such as paragliding', 'catching or throwing baseball', 'dancing macarena', 'laughing', 'eating cake', 'beatboxing', 'feeding birds', 'opening present', 'swinging on something', 'making pizza', 'kitesurfing', 'ice climbing', 'slacklining', 'chopping wood', 'cartwheeling', 'archery', 'welding', 'roller skating', 'playing flute', 'deadlifting', 'bouncing on trampoline', 'shovelling snow', 'playing keyboard', 'celebrating', 'high jump', 'capoeira', 'finger snapping', 'side kick', 'brushing hair', 'cooking sausages', 'throwing ball', 'hammer throw', 'dribbling basketball', 'stretching leg', 'breakdancing', 'snowmobiling', 'canoeing or kayaking', 'headbanging', 'exercising with an exercise ball', 'blowing out candles', 'watering plants', 'cheerleading', 'milking cow', 'snowboarding', and 'driving car'. In second stage, The video will be converted into the frames and each frame will be processed separately. Extract frames from the video at a fixed frame rate (e.g., 30 frames per second). Store these frames in folder for further processing. Ensure they are labeled according to the activity they represent. Resize frames to a fixed size (e.g., 224x224) to match the input dimensions of the CNN model. The sample video selected as in Figure 4 is for Driving Car and the results are stored in one of the temporary folder as shown in the figure 5.

Figure 4: Upload Videos

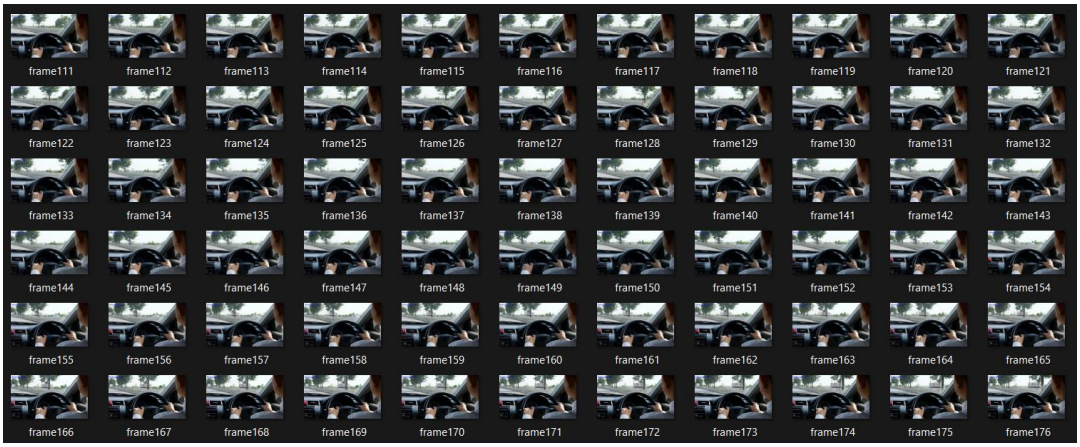

Figure 5 : Stored data of Video sample

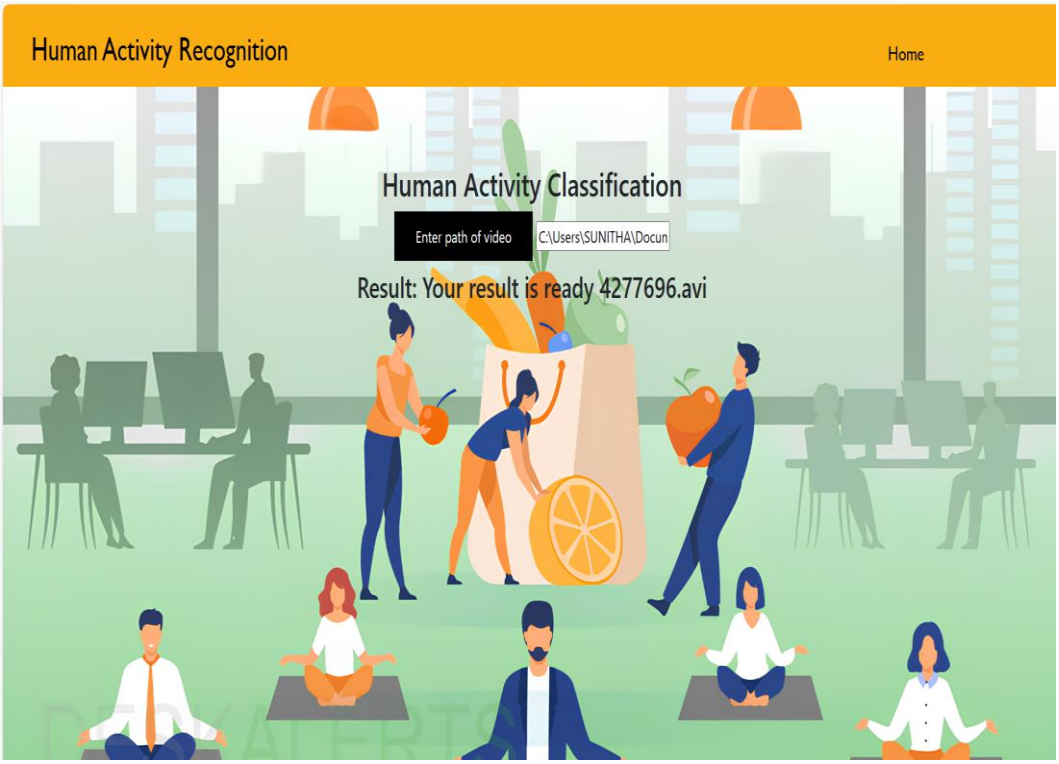

Figure 6: Videos get processed

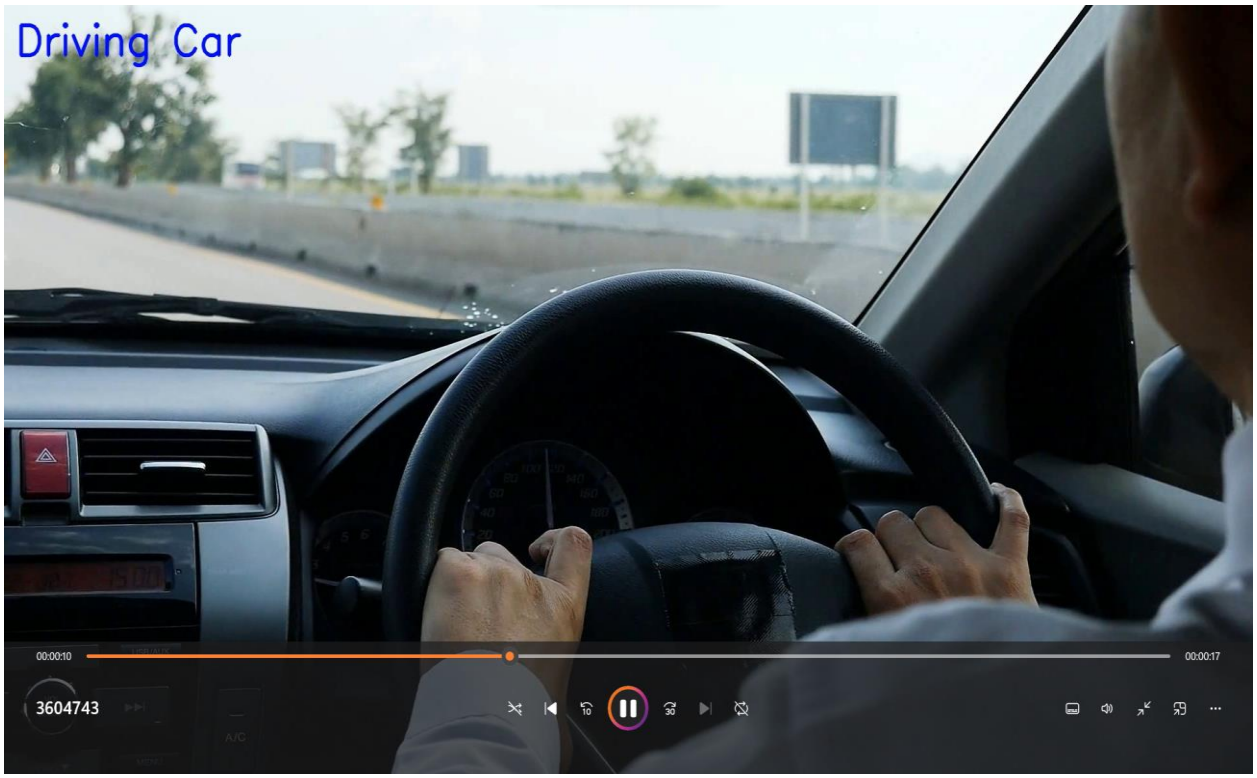

Figure 7: Prediction Result 1

Figure 8: Result 2

V. Conclusions And Future Work

The experimental results concentrated on creating a system for recognizing human activities utilizing OpenCV and exploiting the capabilities of the VGG16 & ImageNet16 model. The objective was to identify and scrutinize human activities in real-time, particularly within 45 different performance scenarios. OpenCV was employed for video processing, feature extraction, and activity recognition, while the ImageNet16 model offered pre-trained functionalities for activity classification. By amalgamating OpenCV with the ImageNet16 model, the project aimed to harness the potential of computer vision and machine learning for precise and effective activity recognition. In essence, the project aimed to showcase the practical application of human activity recognition through OpenCV, coupled with the integration of the ImageNet model, underscoring potential advantages in sports performance analysis and coaching. The amalgamation of these technologies facilitated real-time analysis, personalized feedback, and data visualization, enabling users to discern various activities, make informed decisions, and monitor progress over time.

Future Works

In Future, there are some alternative ways to express those points: Further refining the pre-trained ImageNet model specifically for human activity recognition. This entails training the model on a diverse dataset of labeled human activities to enhance its precision and sensitivity in identifying various actions. Presently, the project concentrates on analyzing individual frames for activity recognition. However, incorporating temporal context into the analysis can augment accuracy and resilience. Techniques like recurrent neural networks (RNNs) or long short-term memory (LSTM) networks can be applied to model temporal relationships and capture motion dynamics over time.

Expanding the system to accept and fuse multiple forms of input, including video, audio, and inertial sensor data. This holistic approach can yield a more comprehensive understanding of human activities by combining visual information with auditory cues or motion data obtained from wearable devices.Enhancing performance analysis capabilities by integrating advanced computer vision techniques. This may involve employing more sophisticated pose estimation algorithms, 3D reconstruction techniques, or joint angle

estimation methods, enabling detailed insights into body movements and kinetics.

These potential advancements can elevate the accuracy, adaptability, and user-friendliness of the human activity recognition system, rendering it suitable for a broad spectrum of applications, spanning sports performance analysis, healthcare monitoring, and interactive user interfaces.

References

[1] Abbaspour S, Fotouhi F, Sedaghatbaf A, Fotouhi H, Vahabi M, Linden M (2020) A comparative analysis of hybrid deeplearning models for human activity recognition. Sensors20(19):5707

[2] Aggarwal JK, Xia L (2014) Human activity recognition from 3Ddata: a review. Pattern Recogn Lett 48:70– 80

[3] Alakwaa W, Nassef M, Badr A (2017) Lung cancer detectionand classification with 3D convolutional neural network (3D-CNN). Lung Cancer 8(8):409

[4] Alawneh L, Alsarhan T, Al-Zinati M, Al-Ayyoub M, JararwehY, Hongtao L (2021) Enhancing human activity recognitionusing deep learning and time series augmented data. J AmbientIntell Humaniz Comput 12(12):10565–10580

[5] Almaslukh B, AlMuhtadi J, Artoli A (2017) An effective deep autoencoder approach for online smartphone-based human activity recognition. Int J Comput Sci Netw Secur 17(4):160–165 Alom MZ, Taha TM, Yakopcic C, Westberg S, Sidike P, Nasrin MS, Hasan M, Van Essen BC, Awwal AAS, Asari VK (2019) A state-of-the-art survey on deep learning theory and architectures. Electronics 8(3):292

[6] G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955.

[7] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[8] I. S. Jacobs and C. P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.

[9] K. Elissa, “Title of paper if known,” unpublished.

[10] R. Nicole, “Title of paper with only first word capitalized,” J. Name Stand. Abbrev., in press.

[11] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].