IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.102 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 7, July 2024

DOI: 10.17148/IJARCCE.2024.13721

Comparative Study of Deep Learning Based Methods for Human Activity Recognition

Shabina Shaik[1], Hari Prasad Chandika[2], Dr Ramachandran Vedantam[3]

PG Student, Department of Computer Science and Engineering, VVIT, Guntur, India[1]

Assistant Professor, Department of Computer Science and Engineering, VVIT, Guntur, India[2]

Professor, Department of Computer Science and Engineering, VVIT, Guntur, India[3]

Abstract: Recognizing human activity is essential to improving interpersonal relationships and interactions between people because it provides important insights about a person's identity, personality, and psychological condition. The difficulties in the field are shown by how difficult it is to appropriately extract this information. Research in artificial intelligence and machine learning is primarily focused on the ability of humans to perceive actions, which is propelling developments in a wide range of applications. Robust activity recognition systems are required for a variety of applications, including robots, human-computer interaction, and video surveillance. Recognition of human activity has become an important area of research in image and video analysis. Numerous research has examined this subject over time, emphasizing both its importance and the continuous search for better approaches. In this regard, we suggest a unique method to improve the accuracy of human action and activity recognition using OpenCV, Convolutional Neural Networks, and Graph Neural Networks based on deep learning.

Our approach uses CNN and GCN algorithms, which are excellent at finding features and patterns in pictures and video sequences, to train a large dataset. This is enhanced by OpenCV, a potent real-time computer vision technology, which makes the identification system's implementation easier. Our method seeks to achieve high precision in identifying and categorizing different human actions by combining CNN, GCN and OpenCV, advancing fields like security, UI design, and autonomous systems.

Keywords: CNN, GCN, deep learning, OpenCV, human action/activity recognition.

I. INTRODUCTION

Understanding human behaviour is essential to improving interactions and relationships with others because it provides important insights about a person's identity, personality, and psychological condition. Accurately extracting such data is a difficult task, highlighting the complex nature of this sector. Artificial intelligence and machine learning research is primarily focused on the way humans understand and interpret behaviours, which has led to significant developments in a variety of applications. Strong activity detection systems are essential for many different applications, such as video surveillance, robotics, and human-computer interaction. As a result, in picture and video analysis, recognition of human activity has become an important research field.

A great deal of research has been done on this subject throughout the years, highlighting both its significance and the continuous search for better techniques. The pursuit of novel approaches has resulted from the necessity for enhanced precision and effectiveness in identifying human actions. Here, we provide a novel approach that improves the precision of human action and activity recognition by utilizing OpenCV's features and the deep learning power of Convolutional Neural Networks (CNN). Our method is to train a large dataset using a CNN algorithm, which is well known for its ability to recognize features and patterns in pictures and video sequences. CNNs are the best at extracting hierarchical characteristics, which makes them perfect for challenging jobs like activity recognition. OpenCV is a powerful real-time computer vision toolkit that works in tandem with deep learning to make the recognition system's implementation easier.

Efficient identification of human activities depends on image processing, feature detection, and performance in real time, all of which are made easier with OpenCV's array of tools and functions. Our approach seeks to achieve high precision in identifying and categorizing different human actions by combining CNN with OpenCV. This method boosts the system's efficiency and real-time performance in addition to accuracy. When these technologies are combined, they provide a potent solution for applications in autonomous systems, security, and UI design where accurate and dependable human activity identification is crucial.

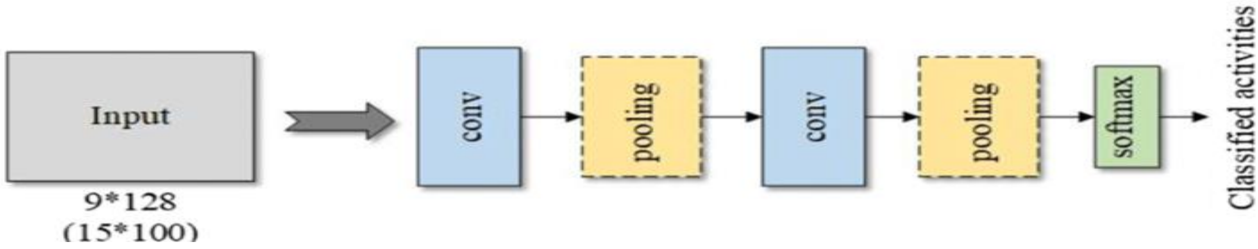

Fig. 1 CNN block diagram for Human activity

In conclusion, our project's main goal is to combine the best aspects of real-time computer vision and deep learning to create a cutting-edge system for recognizing human activities. This novel strategy should greatly progress the discipline by offering insightful information and useful applications.

A. OBJECTIVE

The goal of the research is to create strong deep learning models that are excellent at correctly identifying human actions, which will improve computer vision capabilities. This entails building models that can accurately identify a broad variety of human actions from video or image data. The goal of the project is to improve practical applications across a range of domains by utilizing the strength of Convolutional Neural Networks (CNN) and the real-time capabilities of OpenCV. These models have the potential to greatly increase security in surveillance by instantly identifying questionable activity. They can support patient safety, rehabilitation, and activity monitoring in the healthcare industry. Accurate action recognition can improve user experience by enabling more responsive and intuitive interfaces for human-computer interaction.

B. SCOPE

Designing and executing deep learning models for human action recognition is part of the project's scope. It includes gathering data, pre-processing, and developing models, all with the goal of maximizing efficiency and accuracy. The project's objectives are to assess the models on various datasets and investigate real-world uses such human-computer interaction, surveillance, and healthcare monitoring. Keeping up with new developments in deep learning is also necessary to maximize the project's efficiency.

C. PROBLEM STATEMENT

Deep learning-based human recognition is a major challenge in the field of computer science and artificial intelligence. It requires the creation of algorithms and models that can identify and classify various aspects of human movements and behaviour from images or video data. This function is important for the development of technology throughout the industry as it has many applications such as navigation control, video surveillance and healthy consumption tracking.

II. RELATED WORK

1. A. V. Savakis, M. F. Ong, and J. E. Davies' publication "Deep Learning for Human Action Recognition: A Resource Guide" is a noteworthy addition to the field of deep learning-based human action identification. This comprehensive guide, which was published in the esteemed IEEE Transactions on Circuits and Systems for Video Technology in 2019, is a valuable tool for scholars and professionals involved in this field of study. The writers have put together an extensive manual that covers the depth of understanding and most recent developments in the field of human action recognition. Because of this, the publication is a great resource for both seasoned academics looking to refresh their knowledge and novices in the subject. The guide probably discusses several deep learning architectures and how they are applied to the recognition of human actions, providing information on cutting edge techniques and the difficulties that this field faces.

2. “Interactive Learning-Based Human Recognition: A Review” by S. Wang, L. Liu, and J. Wu was published in the Journal of Sensors in 2019. Deep learning for same-person recognition: a review, S. Wang, L. Liu, and J. Wu provide a detailed overview of the use of deep learning in human performance recognition. This article explores the state-of-the-art, methods, and challenges in using deep learning techniques to analyse human behaviour from visual data, specifically system videos. CNN) and Recurrent Neural Networks (RNN) address the importance of deep learning models used in cognition. They can create many designs to capture the sparks and connections in the heart connection, which is important for understanding human behaviour. The review can discuss data frequently used to test and train these models and highlight their importance for performance evaluation.

ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.102 ※ Peer-reviewed & Refereed journal ※ Vol. 13, Issue 7, July 2024

DOI: 10.17148/IJARCCE.2024.13721

3. Paper "Real-time Human Action Recognition with Convolutional Neural Networks," by K. Simonyan and A. Zisserman, describes a CNN-based method for real-time human action recognition from video data. Their study, which was presented at CVPR 2014, tackles the difficulty of properly and effectively identifying human actions in video sequences. The authors stress that in order to capture intricate patterns and dynamics in video frames, CNN architectures must make use of both spatial and temporal data. This method is critical for applications where accurate and timely action recognition is required, like robotics, human-computer interface, and surveillance systems.

4. Chen, Xie, and Xu's paper, "Action Recognition in Videos Using Advanced Deep Learning Techniques: A Review," published in IEEE Access in 2019, offers a comprehensive examination of the latest developments in action recognition through advanced deep learning methods. The review addresses the intricate challenges associated with video data, such as temporal dynamics and spatial complexities in human actions. It extensively explores various deep learning approaches employed for action recognition, notably Convolutional Neural Networks (CNNs) and recurrent models, detailing their strengths and limitations. The authors also discuss pivotal datasets used for training and evaluating these models, stressing the significance of standardized benchmarks in this evolving field. Moreover, the paper explores practical applications of action recognition systems, spanning from surveillance to advancements in human-computer interaction.

5. Human activity recognition research using deep learning: Velliangiri Sarveshwaran, Iwin Thankumar Joseph, Maravarman M, Karthikeyan P Identifying human behaviour based on the reading process is the goal of Human Activity Recognition (HAR). There are two categories of describing human activities: non-economic activities and economic activities. Income is generated by business type. A non-commercial type of business is used to satisfy the soul. Applications of human recognition include performance evaluation, elderly care, rehabilitation, public theft prevention, smart home, smart transportation, etc. takes place. This study includes deep learning models, their advantages, disadvantages, and data for recognizing human activities. Finally, we have used deep supervised and deep unsupervised learning models to characterize key difficulties in HAR

III. PROPOSED SOLUTION

In this work, we adopt convolutional neural network (CNN), a well-known deep learning algorithm, to build a powerful human action/knowledge system. Master data is created by dividing the video into single frames, which are then analysed to identify various human behaviours. Segmented frames are presented as input to the CNN, which is trained to classify different tasks based on the patterns found in the frames. The training process involves feeding a large number of labels, which allows the CNN to learn complex features associated with each functional group. OpenCV facilitates the processing and analysis of videos by capturing frames and using trained CNN models to predict the actions taken. The system records videos in real time, processes them frame by frame, and analyses human behaviour such as walking, running, jumping, and other actions. This approach has important applications in many fields such as surveillance, human-computer interaction, motion analysis, and healthcare; Interpreting input pressure and human behaviour can provide better insights and improve automation and security measures.

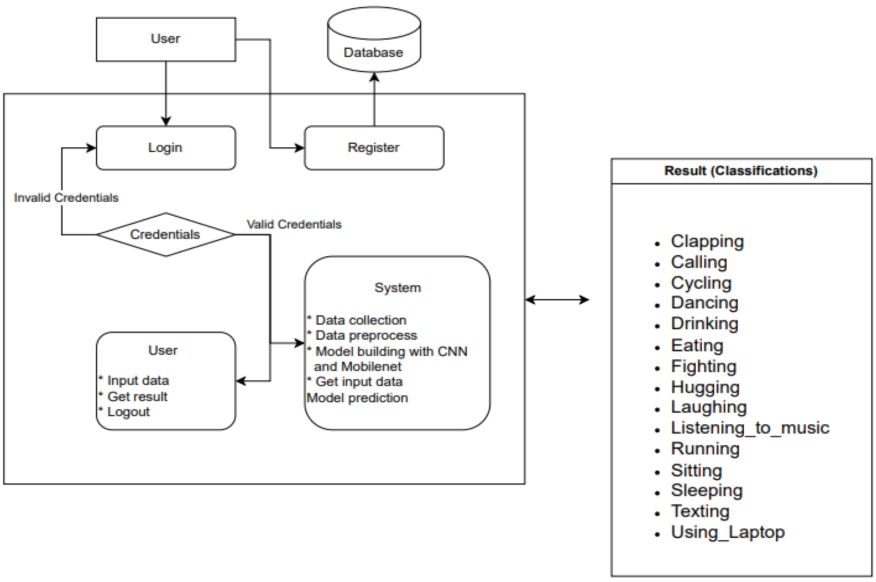

Fig. 2 Block diagram of proposed method

DOI: 10.17148/IJARCCE.2024.13721

1. ***Collection of data:*** This block is a collection of pictures or movies that show different human behaviors. The deep learning models require these datasets in order to be trained and tested.
2. ***Pre- processing:*** Preparing the raw data for testing and training is the task of this step. It consists of tasks including scaling, reshaping, enhancing, and transforming images into a format that the neural network can understand.
3. ***Testing:*** The training and testing portions of the pre-processed dataset are separated. The efficacy and precision of the trained models are assessed using the testing subset.
4. ***Training:*** This subset of the pre-processed data is used to train the Convolutional Neural Network (CNN), Graph Convolution Network (GCN) and MobileNet. During this phase, the models learn to recognize patterns and features associated with different actions.
5. ***CNN, GCN and MobileNet:*** These are the deep learning algorithms used for training the models. CNN, GCN are known for their effectiveness in image recognition tasks, while MobileNet is a lightweight model optimized for mobile and real-time applications.
6. ***Classification***: Once trained, the models can classify new input data into different action categories. This block represents the stage where the trained models are used to make predictions on unseen data.
7. ***OpenCV:*** OpenCV is a computer vision library used for real-time image and video processing. In this block, it captures video input from the web camera and processes it for action recognition.
8. ***Web Camera:*** The web camera captures real-time video, which is fed into the system for action recognition.
9. ***Activities Recognition:*** This final block represents the output of the system, where the recognized actions and activities are displayed or utilized for further applications. It shows the real-time recognition results based on the input from the web camera.

IV. METHODOLOGIES

A. CNN

Convolutional Neural Networks (CNNs) are extensively employed in computer vision applications, such as recognizing human activities. CNNs use unprocessed pixel data from pictures or video frames as their input representation. In order to improve contrast or colour uniformity and standardize proportions, these frames are usually pre-processed.

An outline of possible uses for CNNs is provided below.

1.Convolutional Layers: Convolutional layers are the fundamental components of CNNs. The network can learn hierarchical features thanks to these layers, which apply learnable filters to small input data patches. These characteristics could be forms, textures, or motion patterns in the context of human activity recognition.

$$(I * K)(x, y) = \sum_m \sum_n I(x - m, y - n) K(m, n)$$

Where I is the input image, K is the kernel, and (x, y) are the coordinates of the output feature map.

2. Pooling Layers: Following convolutional layers, pooling layers (like MaxPooling) minimize the feature maps' spatial dimensions while preserving crucial data. This contributes to strengthening the model's ability to vary the input.

3. Fully Connected Layers and Flattening: Fully connected layers receive the output that has been flattened from the last convolutional or pooling layer. These tiers incorporate advanced characteristics and assimilate the connections among them.

4. Output Layer: SoftMax activation is usually used in the last layer of the CNN for multiclass classification, in which each class is associated with a distinct human activity (e.g., walking, running, sitting).

Convolutional Neural Networks (CNNs) are trained using labelled datasets that include recordings of people engaging in various daily activities. The training process involves optimizing the network's weights to minimize a loss function, commonly categorical cross-entropy, which measures the difference between the network's predictions and the ground truth labels. During training, the CNN learns to recognize patterns and features within the data, allowing it to make accurate predictions. This optimization is typically achieved through backpropagation and gradient descent algorithms, iteratively adjusting the weights to improve the network's performance on the task it is being trained for.

DOI: 10.17148/IJARCCE.2024.13721

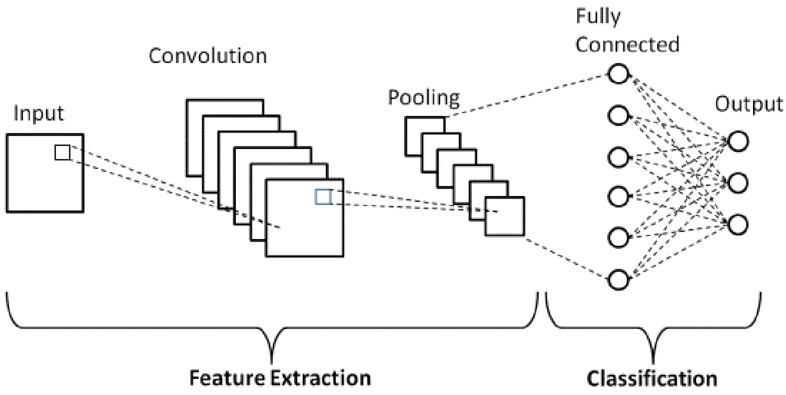

Fig. 3 CNN Architecture

CNN Implementation:

- We are using the pre-processed training dataset to train our model using CNN model.
- CNN algorithm consists of 4 layers: Input layer, Convolution Layer, pooling layer, Flatten layer and dense layer.
- In input layer we consider images as input.
- In Convolution layer, we convert image into matric format. Here matrix size is 224 X 224 (rows X columns).
- In the pooling layer the numerical values will be stored. To change the numerical data to binary data, we use SoftMax layer. In SoftMax layer we will convert the numerical data to binary.
- In flatten layer and dense the classes of total dataset (15 types) is stored which will be in the binary data format.
- We use fit method for saving the data in the form of .h5. Here model is a format for storing the binary data.

B.GCN (Graph Convolutional Network)

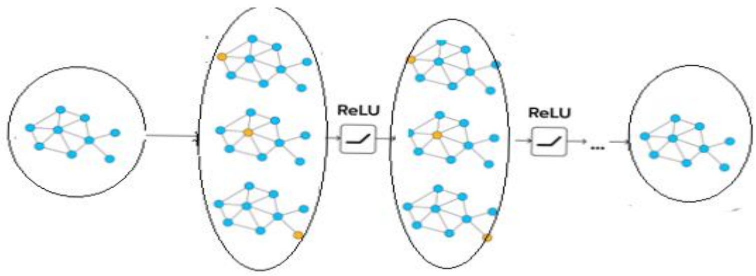

Fig.4 GCN Architecture

Graph Neural Networks (GNNs) represent one of the most captivating and rapidly evolving architectures within the deep learning landscape. As deep learning models designed to process data structured as graphs, GNNs bring remarkable versatility and powerful learning capabilities. Among the various types of GNNs, the Graph Convolutional Networks (GCNs) have emerged as the most prevalent and broadly applied model. GCNs are innovative due to their ability to leverage both the features of a node and its locality to make predictions, providing an effective way to handle graph-structured data.

Graph Neural Networks (GNNs) have shown great potential in human activity recognition (HAR), especially when dealing with data that can be naturally represented as graphs, such as skeletal data where joints are nodes and connections between joints are edges. Here's a step-by-step algorithm for using GNNs in HAR:

1. Data Collection and Preprocessing:

Data Collection: Collect raw skeletal data from various sources such as motion capture systems or depth cameras. Each frame consists of 3D coordinates of human body joints.

Graph Construction: Construct a graph for each frame or segment of data. Nodes represent joints, and edges represent the physical or logical connections between joints (e.g., bones or kinematic chains).

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.102 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 7, July 2024

DOI: 10.17148/IJARCCE.2024.13721

Data Preprocessing: It has two stages: normalization and segmentation. Here the process is to normalize the joint coordinates to have zero mean and unit variance and to divide the continuous data into fixed-size segments or windows, each represented as a graph.

Labelling: Assign activity labels to each graph based on the corresponding activity performed during that time window.

2.GNN Architecture Design:

Input Layer: The input layer takes the graph data, typically represented as an adjacency matrix (A) and a feature matrix (X). The feature matrix contains the coordinates or other attributes of the joints.

Graph Convolutional Layers: the process is to apply multiple graph convolutional layers to extract features from the graph structure. Each layer aggregates information from neighbouring nodes.

Pooling Layers: Introduce graph pooling layers to reduce the graph size while retaining important features. Techniques like Top-K pooling or hierarchical pooling can be used.

Flattening Layer: Flatten the output of the final graph convolutional or pooling layer to convert it into a 1D vector, which can be fed into fully connected layers.

Fully Connected Layers: the process here is to add one or more fully connected (dense) layers to learn higher-level representations of the features. An Example is Dense layers with ReLU activation.

Output Layer: The output layer uses a SoftMax activation function for multi-class classification, with the number of units equal to the number of activity classes.

3. Model Compilation:

Loss Function: Use a suitable loss function like categorical cross-entropy for multi-class classification tasks.

Optimizer: Choose an optimizer such as Adam, SGD, or RMSprop to minimize the loss function during training.

Metrics: Select evaluation metrics like accuracy, precision, recall, and F1-score to monitor the performance of the model.

C. MobileNet

We use MobileNet in this work, which is a simplified form of Convolutional Neural Networks (CNNs) made especially for resource-constrained settings, such embedded and mobile vision applications. Depth wise separable convolutions, which split a regular convolution into a depth wise convolution and a pointwise convolution, are the key component of MobileNet's architecture. With a great deal less computing complexity and parameter count thanks to this novel strategy, MobileNet becomes extremely accurate without compromising on efficiency. MobileNet is optimized for performance on smartphones with low resources, in contrast to typical CNNs, which demand significant computational power and memory. This ensures quick and efficient image processing and recognition tasks. MobileNet's effective design allows us to achieve high accuracy with minimal computing overhead in our applications, which makes it a perfect option for integrating sophisticated computer vision features into embedded and mobile devices. This work demonstrates MobileNet's ability to strike a balance between efficiency and performance, which is important in real-world circumstances where resources are scarce.

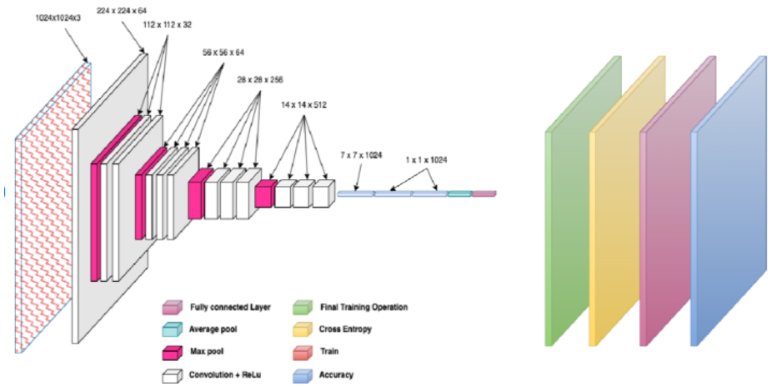

Fig. 5 MobileNet Architecture

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.102 ⁘ Peer-reviewed & Refereed journal ⁘ Vol. 13, Issue 7, July 2024
DOI: 10.17148/IJARCCE.2024.13721

Base Model:

- ***Shape Input:*** The picture of the shape (224, 224, 3) is used by the model to represent the height, width, and three RGB colour channels.

- ***Pre-trained Weights:*** Weights that have been pre-trained on the ImageNet dataset are used to initialize the base model. This makes it possible for the model to employ features that have already been learnt, which is very helpful for transfer learning.

- ***Include Top:*** By excluding the top (completely connected) layers of the original MobileNet (include top=False), we are able to modify the classification head to suit our particular needs.

$$(I * K_d)(x,y,c) = \sum_{i=0}^{k-1}\sum_{j=0}^{k-1} I(x+i, y+j, c)K_d(i,j,c)$$

Where Kd is the depth wise kernel and c is the channel index.

Worldwide Average Pooling: A GlobalAveragePooling2D layer is added after the base model. By calculating the average of each feature map, this layer minimizes the spatial dimensions of the feature maps and produces a vector of features that effectively represents the core of the input image.

Dense Layer: 64 neurons and ReLU activation are added to this completely connected layer. This layer facilitates the extraction of intricate patterns from the characteristics.

Batch Normalization: This layer serves to stabilize and expedite the training process by normalizing the activations of the preceding layer.

Dropout: To avoid overfitting, a dropout layer with a 0.2 dropout rate is employed, which involves randomly changing a portion of the input units to zero during training.

Layer of Output: The last layer has 15 neurons, is dense, and exhibits sigmoid activity. The probabilities for each of the 15 classes in our dataset are generated by this layer.

D. OpenCV:

For processing and analysing photos and videos, OpenCV (Open-Source Computer Vision Library) offers a number of tools and techniques that are necessary. The following methods are applied in the identification of human activity:

1. ***Video Preprocessing and Capture:*** OpenCV makes it simple to capture video from a variety of sources, including webcams and video files. OpenCV functions can be used to efficiently perform preprocessing tasks like frame normalization, scaling, and colour space conversion (e.g., from BGR to RGB).

2. ***Optical Flow:*** Optical flow techniques, such as the Lucas-Kanade approach, monitor the passage of dense flow or key points between successive frames. This aids in comprehending the dynamics of motion in human activity.

3. ***Backdrop Subtraction:*** Methods such as OpenCV's BackgroundSubtractorMOG2 assist in separating foreground objects—that is, people—from the backdrop. This process is essential for identifying and detecting human activity in films.

4. ***Feature extraction:*** OpenCV offers tools to extract different features from pictures or video frames. These tools include Local Binary Patterns (LBP) for texture analysis and Histogram of Oriented Gradients (HOG) for capturing shape and edge information.

5. ***Activity Classification:*** Features can be placed into the CNN model previously mentioned for activity classification after they are extracted. For smooth model training and inference, OpenCV makes it easier to integrate these features with deep learning frameworks like TensorFlow or PyTorch.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.102 ⋇ Peer-reviewed & Refereed journal ⋇ Vol. 13, Issue 7, July 2024

DOI: 10.17148/IJARCCE.2024.13721

6. ***Real-time Processing***: OpenCV is optimized for real-time applications, making it suitable for scenarios like real-time human activity recognition in surveillance systems or interactive human-computer interfaces.

V. DATASET DESCRIPTION

The deep learning-based human activity recognition dataset is made up of a wide range of photos that depict different human behaviours. It encompasses a variety of actions, including calling, clapping, jogging, sitting, drinking, dancing, and cycling.

A wealth of information for developing and accessing machine learning models is made available by the labelling of each image with the associated task. The purpose of the dataset is to facilitate the creation of reliable algorithms that can recognize and categorize human activity in real-world contexts. With the use of vast and varied visual data, this thorough gathering helps to improve the efficiency and dependability of activity detection systems.

TABLE I DATASET SPECIFICATIONS

Source	Data from Kaggle
Image Type	RGB
Image Extension	Jpg
Image Dimension	224,224
Image Quality	Medium to high


Calling


Clapping


Cycling


Drinking


Laughing


Running

TABLE II HARDWARE REQUIREMENTS

Processor	I3/Intel Processor
Hard Disk	160GB
Key Board	Standard Windows Keyboard
Mouse	Two or Three Button Mouse
Monitor	SVGA
RAM	8GB

TABLE III SOFTWARE REQUIREMENTS

Operating System	Windows 7/8/10
Server-side Script	HTML, CSS, Bootstrap & JS
Programming Language	Python
Libraries	Flask, Pandas, MySQL. Connector, Os, Smtplib, NumPy
IDE/Workbench	PyCharm
Technology	Python 3.6+
Server Deployment	Xampp Server
Database	MySQL

DOI: 10.17148/IJARCCE.2024.13721

VI. RESULTS

The CNN model's comparative performance is shown in the figure below, which displays graphs for accuracy, precision, recall, and loss. By carefully examining important performance measures, these visualizations show the model's strengths and shortcomings in terms of its ability to accurately classify human behaviours.

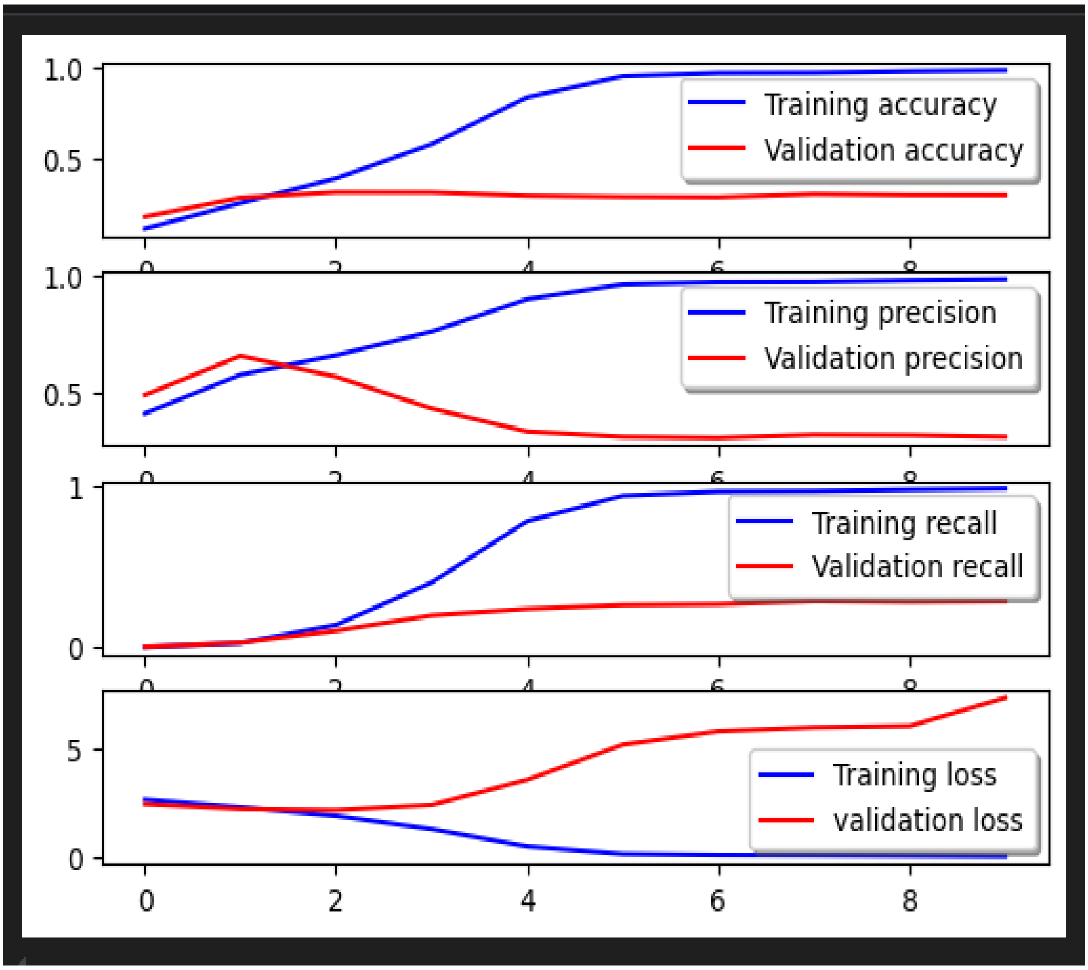

Fig. 6 Comparison graphs for CNN algorithm

TABLE IV IMAGES USED FOR TRAINING AND TESTING

Number of classifiers	15
Classifiers used	15(Calling, Clapping, Cycling, Dancing, Drinking, Eating, Fighting, Hugging, Laughing Listening_to_music, Running, Sitting, Sleeping, Texting, Using_Laptop)
Total number of Input Images	12,600
Training images	8820
Testing images	3780

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.102 ⪿ Peer-reviewed & Refereed journal ⪿ Vol. 13, Issue 7, July 2024

DOI: 10.17148/IJARCCE.2024.13721

TABLE V COMPARISION TABLE FOR ALGORITHMS

Algorithm	Accuracy	Precession	Recall
CNN	98	98	98
GNN	96	96	96
MobileNet	92	92	92

VII. CONCLUSION

In our proposed model, we employ Convolutional Neural Networks (CNN) and MobileNet for recognizing human actions and activities. We utilize a dataset of images depicting various actions, which is used to train the CNN and MobileNet algorithms. After completing the training phase, we integrate OpenCV to capture video input and recognize actions in real time. This combination of deep learning models and real-time computer vision enables precise and efficient identification of human activities, enhancing applications in areas such as surveillance, healthcare, and human-computer interaction. The model ensures robust and accurate performance in recognizing diverse human actions.

REFERENCES

[1]. J.E. Davis, M.F. Ong and A.V. Savakis, “Deep Learning for Human Vision: An Introduction,” IEEE Transactions on Circuits and Systems for Video Technology, 2019, DOI: 10.1109/TCSVT.2018.2846724.
[2]. S. Wang, L. Liu, thiab J. Wu, “Deep learning-based human action recognition: A review,” Journal of Sensors, 2019, DOI: 10.1155/2019/8131971.
[3]. K. Simonyan thiab A. Zisserman, “Instant human action recognition using convolutional neural networks,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, DOI: 10.1109/CVPR.2014.214.
[4]. K. Simonyan and A. Zisserman, “Two-stream convolutional networks for video recognition,” Advances in Neural Information Processing Systems (NeurIPS), 2014, arXiv: 1406.2199.
[5]. S. Chen, X. Xie, and C. Xu, "Action Recognition in Videos Using Advanced Deep Learning Techniques: A Review," IEEE Access, 2019, DOI: 10.1109/ACCESS.2019.2890656.
[6]. L. Wang et al., "Temporal Segment Networks: Towards Good Practices for Deep Action Recognition," Proceedings of the European Conference on Computer Vision (ECCV), 2016, DOI: 10.1007/978-3-319-46487-9_2.
[7]. K. Soomro, A. R. Zamir, and M. Shah, "UCF101: A Dataset of 101 Human Actions Classes from Videos in The Wild," CRCV-TR-12-01, 2012, arXiv:1212.0402.
[8]. G. Bradski and A. Kaehler, "OpenCV," O'Reilly Media, 2008.
[9]. G. Cao et al., "OpenPose: Real-time Multi-Person KeyPoint Detection Library," IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 2019, DOI: 10.1109/TPAMI.2019.2912942.
[10]. Enea Cippitell, Ennio Gambi, Susanna Spinsante, Francisco Florez-Revuelta (2016). Human Action Recognition Based on Temporal Pyramid of Key Poses Using RGB-D Sensors. Springer International, Conference, Advanced Concepts for Intelligent Vision Systems.PP 510-521.vol (10016)
[11]. Qingzhong Liu, Zhaoxian Zhou, SarbagyaRatnaShakya, PrathyushaUduthalapally, MengyuQiao, and Andrew H.ung (2018). “Smartphone Sensor-Based Activity Recognition by Using Machine Learning and Deep Learning Algorithms” International Journal of Machine Learning and Computing, Vol. 8, No. 2.
[12]. Ms. S. Roobini, Ms. J. Fenila Naomi (2019). Smartphone sensor based human activity recognition using deep learning models. International Journal of State-of-the-Art Technology and Engineering. ISSN: 2277-3878, Volume 8, Chapter 1.
[13]. Ye, G. Stevenson, thiab S. Dobson, “Kcar: A knowledge-driven approach to concurrent action identification,” Pervasive and Mobile Computing, vol. Nplooj 19. 47-70, 2015.
[14]. O.D. Lara ve M.A. Labrador, “A survey of human experience using portable devices,” IEEE Communications Surveys and Tutorials, vol. 15. No. 3, issues 1192-1209, 2013.
[15]. M. Ziaeefard thiab R. Bergevin, “Semantic human activity recognition: A literature review,” Pattern Recognition, vol. 48, No. 8, s. 2329-2345, 2015.