VISION-BASED HUMAN ACTIVITY RECOGNITION USING DEEP LEARNING
CS5720 Neural Network Deep Learning - Project

***Abstract*—This paper presents a novel approach for vision-based identification of human activities. HAR with the use of transfer learning and convolutional neural networks (CNN). As deep learning gained popularity, other concepts emerged to address the HAR's issues. Recognizing human behavior without disclosing the identity of the individual is one example. HAR has grown in importance across a wide range of industries, including security systems, anomaly detection, and senior citizen monitoring. Nonetheless, cutting-edge computer vision techniques are thought to be potential advancements for building video frame-by-frame classifications of human activity. People's daily lives are greatly impacted by human activity detection because it can gather enormous amounts of high-quality data about human activity from fixed or wearable sensors. within the film, many objects and individuals may be seen behaving in various locations within the image. For visual reasoning in an action detection challenge, multi-spatial unit interactions must be modeled.**

Deep learning models have surpassed traditional machine learning techniques by independently extracting characteristics with minimal computer resources. In our study, we extract human activity from image datasets using a CNN model. We can efficiently use deep picture features to train machine learning classifiers through transfer learning.Utilizing the VGG-16 framework, our trials yield a remarkable 96.95 percent accuracy. Our findings also demonstrate how much better VGG-16 performs than other CNN models that were employed in HAR. This work offers a viable method for vision-based action detection in real-world applications and advances the development of HAR approaches. The study intends to demonstrate how these methods have the ability to completely transform how we watch and interpret human activity by reviewing the most recent methods and applications of vision-based HAR using CNNs and transfer learning.

***Index Terms*—Deep learning, Convolutional Neural Networks, Human Activity Recognition, Transfer Learning,**

I. INTRODUCTION

The field of human activity recognition (HAR) has gained significant importance in recent times owing to its extensive applications in domains like human-computer interaction, sports analytics, healthcare, and surveillance.A more successful and efficient strategy has arisen as a result of the rapid growth of deep learning techniques. Traditional methods depended on hand-crafted features and intricate algorithms to detect activities. utilizing the power of deep learning and data from pre-trained models, video-based human activity identification utilizing convolutional neural networks (CNN) and transfer learning is a potential method to capture intricate patterns of human motions and activities.

The goal of the multidisciplinary discipline of HAR, which is connected to computer vision, is to examine how individuals move, balance, control their posture, and interact with their surroundings. Biomechanics, artificial intelligence, pattern recognition, machine vision, image processing, data analysis, and nonlinear modeling are all included in this. Two-dimensional, depth, or thermal images, motion, body-mounted sensors, or smartphones can all be used to analyze it. In this context, human activity is detected by human model-based techniques based on the locations and movements of body parts. However, researchers must examine the variety of human body sizes, postures, movements, appearances, clothing, camera movement, views, and lighting in order to create a suitable and efficient HRT system.

Numerous applications of HAR in a wide range of fields and complexity have been studied. These include security, environmental monitoring, video surveillance, robotics, training and hands-on courses, immediate response, health care, specific medical diagnosis, fitness monitoring, and bio-mechanical analyses, such as those involving data analysis. HAR's primary obstacles consist of: (i) analyzing unclear gestures and positions; (ii) certain persons may classify certain stances and movements differently, and there may be partial occlusion of things relevant to the body or scene. (iii) Videos with dubious quality, like fuzzy images and noisy data from poor sensors. (iv) Significant variations in operation times between one another. (v) Either excessive brightness or no lighting. (vi) Acquiring massive data sets is difficult.

One difficulty in video analysis is human action detection, which is the identification and categorization of human movements inside specific frames. A human action like kicking or pulling away is an illustration of this. It is possible to train a classifier on a subset of action cases (training set) and then test it on another subset of action cases (test set). The system's objective is to determine which activity category a given video frame belongs to, or more broadly, to recognize and assess human activity in video frames.

The goal of this study is to present a thorough overview of CNN-based learning and vision-based human action recognition. We start off by talking about the difficulties and possibilities involved in identifying human activity in video.

Github Link - https://github.com/yasaswini8777/NNDL-Final-Project

Next, we explore the fundamentals of CNNs and emphasize how they can automatically learn spatial feature hierarchies, which is crucial for efficient action detection. The idea of transfer learning is then discussed, which enables us to leverage pre-trained models to shorten training times and utilize less processing power to enhance the performance of our detection systems.

Vision-based technology classifies human activities using still and video pictures captured by infrared or depth cameras. Although they are non-intrusive, sensor-based HRT systems might not offer very high accuracy. As a result, there is currently a lot of interest in vision-based human activity detection systems. It's challenging to identify human activity in streaming video.

Based on mobilities, character-based and vision-based approaches to human activity detection can be divided into two categories. An optical handheld marker is used in the MoCap framework using the marker-based approach. Although this method may properly capture intricate human movements, it has certain limitations. Both the attachment of optical sensors to the subject and specific camera settings are needed. Alternatively, an RGB or depth image is used in the vision-based technique. The user is not required to wear any gear or affix any sensors to their body. Consequently, this approach is receiving increased attention these days, and as a result, the HAR framework is straightforward and simple to use in a variety of applications.

In conclusion, we examine cutting-edge methods and uses of CNNs and learning for vision-based human resource management. We present the potential of these technologies to fundamentally alter how we view and interpret human behavior. It is essential to embrace the synergy of CNNs and transfer learning as we continue to push the frontiers of HAR in order to build more precise and effective systems that can comprehend and interact with the environment around us more effectively.

II. MOTIVATION

A. Need of the project

The increasing requirement for intelligent systems that can comprehend and interpret human behavior across multiple domains has led to the necessity of a project centered on vision-based human recognition employing CNN and transfer learning. This problem can be solved quickly and effectively by combining CNNs with transfer learning, which offers significant advantages and insightful information for a wide range of applications. Our primary goals in initiating this initiative are:

Improved security and surveillance: These systems can greatly increase the efficacy of video surveillance and assist in the detection and prevention of possible threats, crimes, and accidents by precisely identifying and detecting human behavior in real time.

Health care and housing support: Health practitioners can receive vital information from tracking and evaluating the behaviors of older patients or patients with specific health issues that enables timely and customized therapy. People with impairments can live more independently thanks to this technology, which can help by sending out notifications or offering support when needed.

Enhanced Human-Computer Interaction: Enriching user experiences, a strong action recognition system can identify and understand human actions to facilitate easier and more natural interactions between platforms and apps.

Sports analytics and training support: Through the system's automatic detection and observation of athletes' movements, trainers can enhance training programs and achieve better overall results by receiving useful information on performance, technique, and injury risk.

Workplace safety and ergonomic optimization: Monitoring movements and activities at work can assist spot possible risks, guarantee adherence to safety guidelines, and enhance workplace ergonomics to lower the chance of accidents and boost output.

Immersive entertainment and gaming experience: For virtual reality, augmented reality, and gaming applications, real-time recognition and reaction to user inputs can result in a more engaging and dynamic experience.

Efficient use of resources: Through the use of transfer learning, this research can shorten the time and computational resources required to train complicated models, increasing its affordability and accessibility for a wide range of applications.

To put it briefly, the pressing requirement to construct sophisticated and effective systems capable of interpreting human activity in a variety of scenarios is met by the project that emphasizes the vision-based recognition of human action utilizing CNN and transfer learning. In today's increasingly linked and data-driven world, the technology's many potential uses and advantages highlight the significance and appropriateness of this endeavor.

III. MAIN CONTRIBUTIONS AND OBJECTIVES

- Creation of deep learning models to reliably identify human activities in video footage.
- Investigation of real-time recognition methods and robustness to changes in the environment.
- Attain a high degree of accuracy while identifying various human behaviors.
- For a useful deployment, make sure scalability and real-time performance are met.
- Boost resistance to outside influences including lighting and camera angles.
- Preserving Privacy
- Generalization and Transfer Learning

IV. RELATED WORK

Human activity recognition using vision has drawn a lot of interest lately. The majority of research has employed traditional classifiers and manually created features in photos and videos to detect activity. Conventional techniques demonstrated the best performance and produced the best results in many circumstances. However, customary methods are

Github Link - https://github.com/yasaswini8777/NNDL-Final-Project

unworkable in real-world scenarios because to the significant data dependence of hand-crafted functions and their limited flexibility in responding to changing circumstances.

The ability of HMM ("Hidden Markov Model") approaches to extract temporal patterns has made them popular detection methods in recent years. But because deep learning algorithms can automatically extract characteristics and learn frameworks from deep patterns, researchers are turning to them more and more. Deep learning algorithms have supplanted traditional categorization methods in the field of computer vision. From a computational standpoint, these methods have garnered a lot of attention lately and have yielded fantastic results. Consequently, there has been a lot of interest in recent years in the use of deep learning algorithms for video-based human activity detection. Below are some related work:

- "Deep Learning for Human Activity Recognition: An Implementation with Minimal Resources." With an emphasis on performance, this research investigates the use of deep learning models for human activity detection.
- A Study on Human Action Recognition Using Deep Learning Techniques by Yang Wang, Andrey Gavrilov, Mohammad Mehedi Hassan, and Abdulmotaleb El Saddik. This study provides an overview of deep learning techniques applied to human activity recognition, discussing different architectures, datasets and challenges.
- Angelo Cardador and Marco A. Gutiérrez "Convolutional Neural Networks for Human Activity Detection Using Mobile Sensors". This paper investigates the use of Convolutional Neural Networks (CNN) for human activity detection based on data collected from mobile sensors.
- "Human Action Detection Using Convolutional Neural Networks" by Jan Schröder, Niklas Tim Kohl, and Patrick van der Smagt. The authors explore applications of CNNs for human activity detection from sensor data, discuss model architectures and performance evaluation..
- Zhaoxia Yin and Jianwei Niu, "Long-Term Short-Term Memory Networks for Human Activity Recognition." This article focuses on the use of long-short-term memory (LSTM) networks for human activity detection, discussing model design and experimental results.

V. PROPOSED FRAMEWORK

For classifying human activities, System Vision-based technology makes use of infrared, depth, or video cameras. Videos can be distinguished using motion features in video-based human activity detection. The vision-based approach makes use of the RGB or depth image. It is not necessary for the user to wear gadgets or affix sensors to their body. This approach is becoming more and more common as a result, making the HAR framework straightforward and user-friendly in a variety of contexts. The most popular kind of deep learning approach is convolutional neural networks (CNN). Applications using computer vision frequently use CNNs. It is made up of processing layers made of circles for the photos.

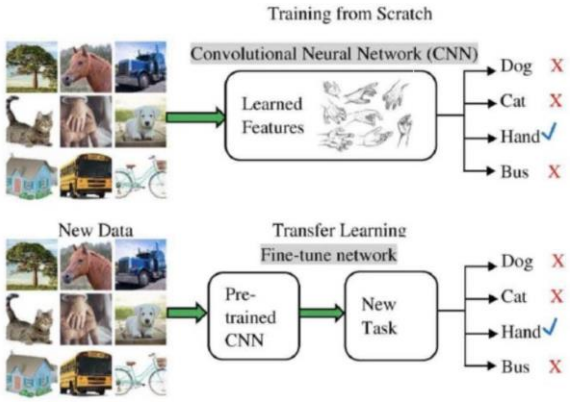

Fig. 1. Fig. 1. Schematic diagram to demonstrate transfer learning

A. MERITS OF PROPOSED FRAMEWORK

- We use CNN for human action detection on the action detection kinetic dataset.
- We use transfer learning to derive deep image features and trained machine learning classifiers.
- The user does not need to wear any equipment or attach sensors to the person.
- Convolutional Neural Networks (CNN), VGG-16 (also called Oxford Net) algorithm is used.

B. MODULES

1) TRANSFER LEARNING: Transfer learning is the process of adding knowledge from earlier, intensive training to the current model. Deep network models can be trained using a substantially lower quantity of data thanks to transfer learning. It was applied in order to decrease training time and raise model accuracy. In this work, we employ transfer learning to mine data from massive datasets like ImageNet. First, each operation's frames are taken out of the videos. To extract deep picture features and train machine learning classifiers, we employ transfer learning. All CNN models in ImageNet employ pre-trained weights as the foundation for transfer learning. A dataset called ImageNet contains 20,000 activity classes. The Weizmann dataset receives the information from the ImageNet pretrained weights since the features found in this work fall inside its domain. The characteristics are taken out of the CNN's penultimate layer. Figure 1 illustrates the fundamental idea of transfer learning. The main approaches in transfer learning are: Keeping the initial neural model that was pre-trained for a sizable dataset and modifying the trained model's weights for the target dataset. Obtaining and representing features using the neural model that has been trained beforehand, then applying a general classifier like Support Vector Machine Logistic Regression.
2) USER: The project can be started by running the main.py file. The user has to input the path to a video file.The primary camera of the system is represented by

Github Link - https://github.com/yasaswini8777/NNDL-Final-Project

the openCV class Video Capture (0), and the secondary camera is represented by Video Capture (1). It is demonstrated that we can load a video file from a disk without a camera using video capture (path to a video file). The variables Vgg16 and Vgg19 are set.The model selection in the code can be altered by the user, who can also fill it in in other ways.

3) HAR SYSTEM: Vision-based techniques can be used for human activity recognition from videos. The RGB or depth images are used in the vision-based technique. The user is not required to wear any gear or affix any sensors to their body. This approach is becoming more and more common as a result, making the HAR framework straightforward and user-friendly in a variety of contexts. For every task, we began removing frames from the videos. To acquire learned machine learning classifiers and deep picture features, transfer learning is specifically utilized.

4) VGG16: One model of a convolutional neural network is VGG16. Deep Convolutional Networks for Large-Scale Image Recognition With approximately 14 million photos categorized into 1000 classes in ImageNet, the model attains 92.7 percent top-5 test accuracy. Among the popular models submitted to the ILSVRC-2014 was this one. By sequentially replacing several 33 kernel-sized filters in the first and second convolutional layers, which were previously huge kernel-sized filters (11 and 5, respectively), it performs better than AlexNet. For weeks, VGG16 had been using NVIDIA Titan Black GPUs for training. We can begin by utilizing the pre-trained VGG-16 model as a feature extractor in order to apply VGG-16 for HAR. Every frame of the video contains features that can be extracted from the convolutional layers of VGG-16. These features can then be used as inputs to a classifier to predict the activity shown in the video. As an alternative, we can modify VGG-16 for HAR by adding a new layer tailored to the task at hand in place of the last fully linked layer. Then, using a labeled dataset of movies, we may train this updated model to identify the features most important for HAR. We may use the trained model to categorize fresh films according to the activity they include.

C. SYSTEM DESIGN

1. UML DIAGRAMS: A common visual language used to model, \textbf{design,} and describe software systems is called UML, or Unified Modeling Language. It is a graphical language that may be used to model \textbf{many} aspects of a system, such as its interactions, behavior, and structure. It is made up of a collection of diagrams and \textbf{notational} elements.

Software developers, designers, architects, and other stakeholders in the software development process can all benefit from the standardized and widely acknowledged method of communicating system design concepts and ideas that UML offers. This helps them detect possible difficulties or issues early in the design process by enabling them to communicate and depict the system in a clear, succinct, and unambiguous manner.

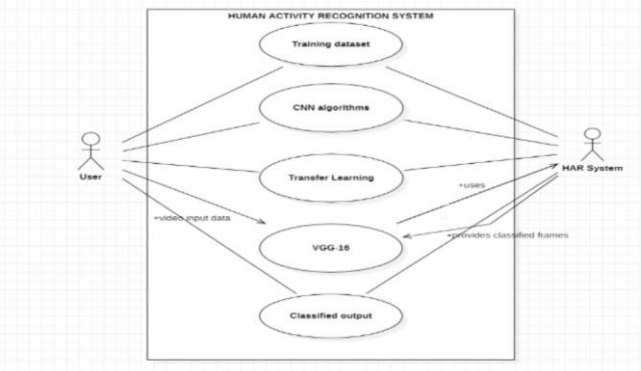

Fig: use case diagram

Fig. 2. Use Case Diagram

A variety of diagram kinds, including class, activity, sequence, use case, and many more, are included in UML. Every kind of diagram serves a distinct function and offers an alternative viewpoint on the system that is being described.

For example, class diagrams are used to show a system's static structure, which consists of its classes, characteristics, and relationships. On the other side, activity diagrams depict how processes or activities move through a system. Sequence diagrams display how items or system components interact with one another over time.

All things considered, UML is an effective tool that aids programmers and designers in the standard and effective visualization, communication, and documentation of complex systems.

2. USE CASE DIAGRAM:

A use case diagram illustrates how players, such as users or systems, interact with a system within a particular scenario or environment. A use case diagram's primary goal is to present a high-level overview of a system's behavior while capturing the functional requirements of the system.

Use case diagrams usually include relationships, actors, and use cases. Use cases indicate certain \textbf{actions} or tasks that the system is capable of performing, whereas actors represent users or other systems that interact with the system being represented. Actor-use case relationships illustrate how actors work with the system to \textbf{achieve} their objectives.

Use case diagrams help stakeholders communicate and understand one another better in software development, project management, and business analysis. They guarantee that all stakeholders have a common understanding of the behavior of the system, assist in identifying the functional needs of the system, and record user experiences with the system. Prior to beginning development, they also assist in locating possible mistakes or weaknesses in the system design.

3. CLASS DIAGRAM

Github Link - https://github.com/yasaswini8777/NNDL-Final-Project

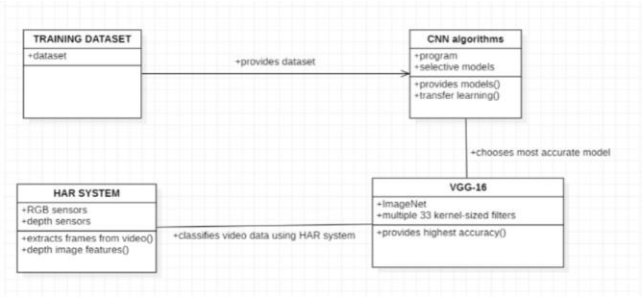

Fig. 3. Fig. Class Diagram

A Unified Modelling Language (UML) diagram known as a class diagram is used to show how a system or piece of software is organised in terms of its classes, characteristics, methods, and relationships.

A collection of classes and their connections make up a class schema. Every class is shown as a rectangular box with the name of the class printed inside. Typically, a class consists of three sections: the class name at the top, the class properties (data members) in the centre, and the class methods (member functions) at the bottom.

Category boxes are connected by lines to show relationships between them. Several relationships can be represented using a class diagram, such as:

- Inheritance: A solid line with a closed arrowhead pointing from a subclass to its super-class, symbolising the relationship between a subclass and its super-class.
- Association: a line connecting the two class boxes represents the relationship between two classes.
- Aggregation: A specific relationship denoted by a diamond on the side of the class holding the instances, which depicts an entity relationship in which one class consists of one or more instances of another class.
- Composition: A more robust type of association in which the composite object's lifespan is connected to the composite object's lifetime, symbolised by the filled diamond shape on the side of the class that houses the instances.

  A class diagram, in general, offers a high-level perspective of a program or system's architecture and is helpful in comprehending the connections between classes and how they cooperate to produce system functioning.

4. ACTIVITY DIAGRAM:

One kind of Unified Modelling Language (UML) diagram that shows how operations or activities flow through a system, process, or workflow is called an activity diagram. It is employed to simulate the behaviour of a system or process and illustrates the series of choices and actions necessary to do a job or accomplish an objective.

Nodes and edges, which stand for activities or functions and transitions or flows between functions, make up an activity graph. An operation diagram's primary node kinds are:

- Initial node - represents the starting point of the activity diagram.

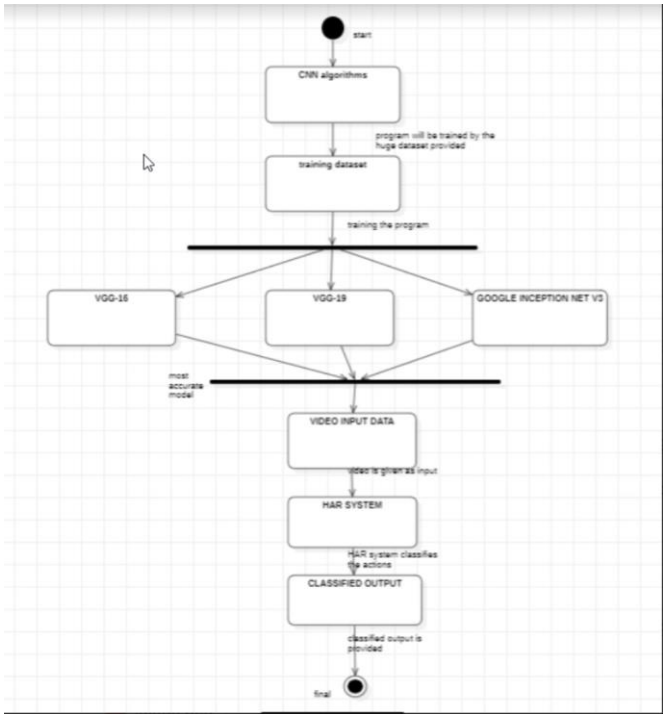

Fig. 4. Fig. Activity Diagram

- Action node - r**means** a specific action or activity **to be** performed.
- Decision node - represents a decision point in a flow where the behavior of a system or process changes based on some condition or criterion.
- Fork node - represents the **separation** of flow **in** two or more parallel paths
- Join node - represents the merging of multiple parallel paths into a single path.
- Final node - represents the endpoint of the activity diagram.

  Edges in an activity diagram can be of different types, such as:
- Control Flow - Represnts the normal sequence of activities in the flow.
- Object Flow - Represents the flow of objects or data between activities.
- Exception Flow - Represents the flow of activities in case of an error or exception.

  An activity diagram, in general, offers a visual representation of a system or process's functions and flow that can aid in understanding, analysing, and improving the system or process.

D. IMPLEMENTATION

A. Dataset :

In particular, we use the Weizmann dataset in our studies to identify activities in order to assess the efficacy of the models. Nine distinct individuals can be seen in a sequence of ninety low- resolution videos carrying out ten different actions:

Github Link - https://github.com/yasaswini8777/NNDL-Final-Project

TABLE 1: DATASET STATISTICS IN TERMS OF NUMBER OF FRAMES PER ACTIVITY

Activity	Number of Frames
Bend	639
Jack	729
Jump	538
Run	346
Side	444
Skip	378
Walk	566
Wave1	653
Wave2	624
Total	4917

Fig. 5. Enter Caption

crouch, jack (or jump, jump jack), jump (or jump two feet forward), hop (or jump two feet forward), run, sidestep (or gallop sideways), jumping, walking, wave l (of one hand) and wave 2 (of two hands). We employed nine activities in our investigation (two- footed jumping not included). Initially, we separate each video into separate frames based on how well they perform.

B. Discussion and Results:

For action detection, we test three distinct convolutional neural networks (CNNs): Google's InceptionNet-v3, VGG-19, and VGG-16. Transfer learning is a technique we utilised to learn from big datasets like ImageNet. By using data from previously trained models, the transfer learning technique trains a neural network in a new area. Using the data from the ImageNet pre- trained weights, we ran a test on the Weizmann dataset. CNNs are used to extract features from the penultimate levels.

Using the VGG-16 CNN model, we applied transfer learning and attained 96.95 percent accuracy. VGG-16 creates a 4096- dimensional vector for each image by extracting features from the fc1 plane, given an input image with dimensions of 224 by 24.

Additionally, we compared the effectiveness of many CNN models—including Google's InceptionNet-v3 and VGG-19—using transfer learning. Google's InceptionNet-v3 and VGG-19 received scores of 95.63 and 96.54 percent, respectively. The test findings indicate that VGG-16 performs better than other CNN models after undergoing transfer learning for all models. The accuracy, precision, recall, and f1 scores of the employed CNN models are displayed in Table 2. The confusion matrix for each of the three CNN models is displayed in Figures 2, 3, and 4.

We contrasted our method's output with that of other methods that did not employ transfer learning using the Weizmann dataset. Higher recognition scores were obtained when the identical content was subjected to transfer learning, according to the experiment's findings. Transfer learning increases accuracy and recognition by 6%. The VGG-6 model's transfer learning outcomes are contrasted with those of alternative methods in Table 3. The usefulness of transfer learning in

Model	Accuracy (in %)	Precision (in %)	Recall (in %)	F1-score (in %)
VGG-16	96.95	97.00	97.00	97.00
VGG-19	96.54	97.00	97.00	96.00
Inception-v3	95.63	96.00	96.00	96.00

Fig. 6. TABLE 2: RESULTS ON ACTIVITY RECOGNITION BASED ON DIFFERENT CNN MODELS IN TERMS OF ACCURACY SCORE, PRECISION, RECALL, AND F1-SCORE.

Model	Accuracy (in %)
VGG-16	96.95
Cai et al. [19]	95.70
Kumar et al. [20]	95.69
Feng et al. [21]	94.10
Han et al. [22]	90.00

Fig. 7. TABLE 3: PERFORMANCE COMPARISON USING WEIZMANN DATASET

conjunction with CNN models to raise detection scores is compared to cutting- edge methods.

Figures 8, 9 and 10 show the confusion matrices of three different convolutional neural networks (CNNs) after transfer learning, Google's VGG-16, VGG-19, and InceptionNetv3 convolutional neural networks (CNNs) were utilised to identify various action frames following trans- directional learning. Despite having very similar visual perception, Figures 2, 3, and 4 demonstrate that VGG-16 is misclassified for skipping when predicting running, VGG-19 is misclassified for jumping and jumping for walking when predicting running, and Google's InceptionNet-v3 is misclassified when predicting running passes. The accuracy of activity detection in CNN models was increased through the use of transfer learning. Additionally, the information supplied by Imagenet's pre-

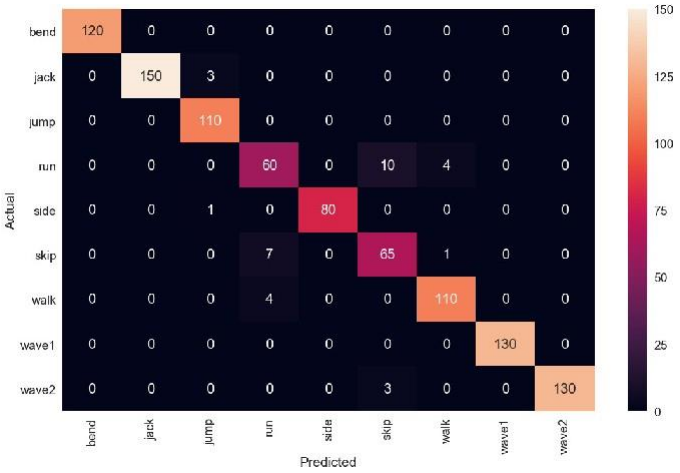

Fig. 8. Confusion matrix for recognizing 9 activities on Weizmann Dataset using VGG-16 Convolutional Neural Network

Github Link - https://github.com/yasaswini8777/NNDL-Final-Project

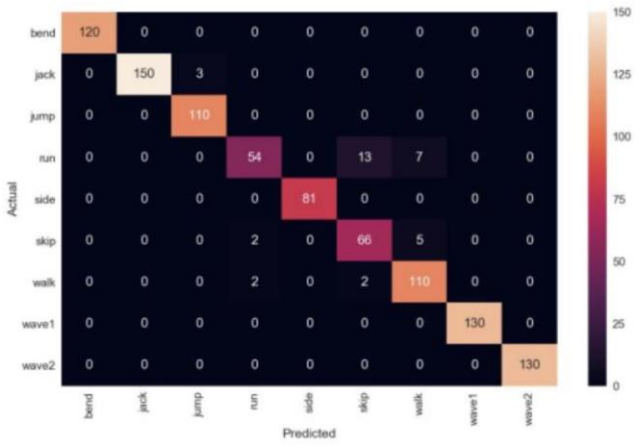

Fig. 9. Confusion Matrix for recognizing 9 activities on Weizmann Dataset using VGG-19 Convolutional Neural Network

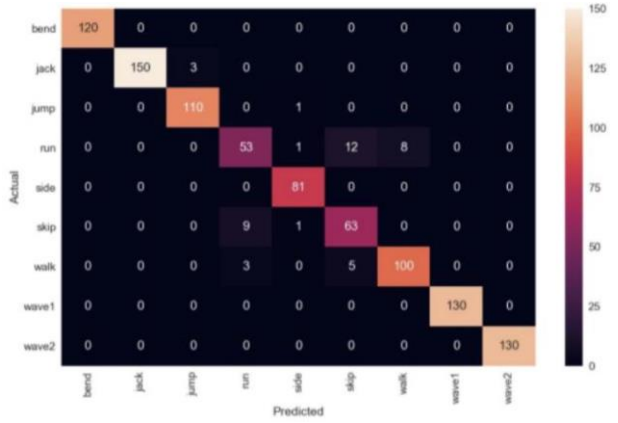

Fig. 10. Confusion Matrix for recognising 9 activities on Weizmann Dataset using Inception- v3 Convolutional Neiral Network

Fig. 11. The Kinetic Dataset

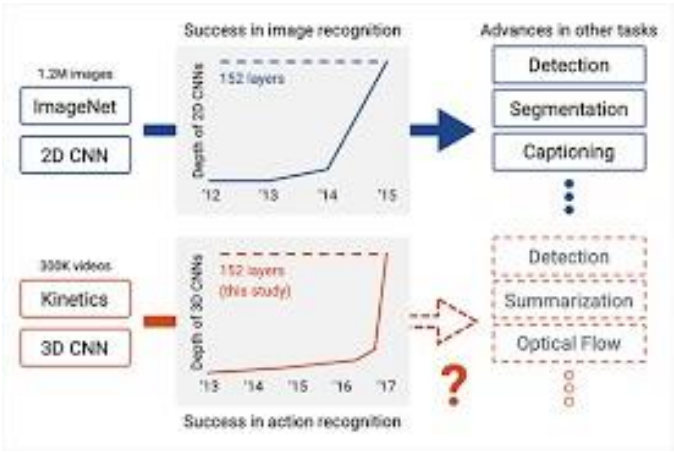

Fig. 12. Enter Caption

trained weights may cause issues for the transfer learning technique employed in our work, as Imagenet contains photos of various classes.

E. DATASET

The Kinetics Dataset:

The pre-trained deep learning model for human action recognition that was used in today's tutorial and was trained on the Kinetic 400 dataset is depicted in the image below. The Kinetic 400 dataset is the one we used to train our human action detection model.

This dataset consists of:

- 400 human activity recognition classes.
- At least 400 video clips per class (downloaded via YouTube).
- A total of 300,000 videos.

The whole list of classes that the model recognises is available here. See the 2017 publication by Kay et al. for additional details regarding the dataset, including its curation.

3D Res Net for Human Activity Recognition

Figure 12 (picture below) demonstrates how advancements in deep neural networks for image categorization using ImageNet have also aided in the recognition of deep learning activities (i.e., videos). Using OpenCV, we will conduct deep learning activity detection in this tutorial. Figure 1, Hara et al. is the image source. The 2018 CVPR study by Hara et al. is the source of the model utilised to identify human activities. In this work, the authors investigate how existing state-of-the-art 2D architectures (such as ResNet, ResNeXt, DenseNet, etc.) can be extended to video classification using 3D kernels.

The authors argue:

- These architectures have been effectively used for image classification.
- the extensive ImageNet dataset made it possible to train such models with such high accuracy.
- Since the Kinetics dataset is also sizable enough, these architectures ought also be able to classify videos by using 3D kernels within the architecture and altering the input

Github Link - https://github.com/yasaswini8777/NNDL-Final-Project

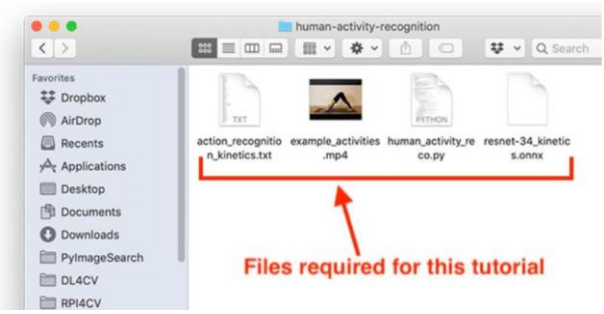

Fig. 13. Download HAR Model For Open Cv

volume shape to incorporate spatiotemporal information. The authors were in fact correct! By modifying both the input volume shape and the kernel shape, the authors obtained:

- 78.4% accuracy on the Kinetics test set
- 94.5% accuracy on the UCF-101 test set
- 70.2% accuracy on the HMDB-51 test set

When spatiotemporal information is added and 2D kernels are swapped out for 3D kernels, these model architectures can be employed for video classification. These findings are comparable to the first-order accuracies reported for state-of-the-art models trained on ImageNet.

Downloading the Human Activity Recognition Model for Open CV

Fig. 13: The necessary files for OpenCV and deep learning to recognise human behaviour. In order to continue with this tutorial, you must download the:

- Human activity model
- Python + Open CV source code
- Example video for classification

You can use the "Downloads" section of this tutorial to download a.zip containing all three. Once downloaded, continue on with the rest of this tutorial. Project structure Let's inspect our project files: Human Activity Recognition with Open CV and Deep Learning tree

- action_recognition_kinetics.txt
- resnet-34_kinetics.onnx
- example_activities.mp4
- human_activity_reco.py
- human_activity_reco_deque.py 0 directories, 5 files Our project consists of three auxiliary files:
- action_recognition _kinetics.txt: The class labels for the Kinetics dataset.
- Res net-34_kinetics.onx: Hara et al.'s pre-trained and serialized human activity recognition convolutional neural network trained on the Kinetics dataset.
- example_ activities.mp4: A compilation of clips for testing human activity recognition. We will review two Python scripts, each of which accepts the above three files as input:
- human_activity_reco.py, blobFromImages (i.e. plural) rather than the blobFromImage (i.e. singular) function — the reason here is that we're building a batch of multiple images to be passed through the human activity recognition network, enabling it to take advantage of spatiotemporal information.
- 1: The batch dimension. Here we have only a single data point that is being passed through the network (a "data point" in this context means the N frames that will be passed through the network to obtain a single classification).
- 3: The number of channels in our input frames.
- 16: The total number of Frames in the blob.
- 112 (first occurrence): The height of the frames
- 12(second occurrence): The width of the frames. We're now prepared to carry out inference for human activity recognition, which entails labeling the frame with the anticipated label and displaying the forecast on the screen: To get a list of the predicted outputs, blob through the network. Next, we retrieve the label corresponding to the blob's highest prediction (Line 66). We can then render the prediction on each frame in the frames list (Lines 69–73) using the label. This will display the output frames until the q key is pushed, at which point we will break and quit.

Alternate Human Activity Implementation Using a De que Data Structure You'll see the following lines in our earlier section on human action recognition. This application suggests that:

- We read a total of SAMPLE_DURATION. Frames from our input video.
- We pass those frames through our human activity recognition model to obtain the output.
- And then we read another SAMPLE_DURATION Frames and repeat the process. Our implementation is not a rolling prediction as a result. Rather, all that needs to be done is take a sample of frames, categorize them, and then go on to the next batch, discarding any frames from the prior batch.

Human Activity Recognition Results

Let's observe the outcomes of our code for recognizing human behavior! The pre-trained human activity recognition model, the Python + Open CV source code, and an example demo video can be downloaded using the "Downloads" portion of this tutorial. Launch a terminal from there and type the following command: 32

Please note that our Human Activity Recognition model requires at least Open CV 4.1.2.

If you are running an older version of Open CV you will receive the following error: If you receive that error you need to upgrade your Open CV install to at least Open CV 4.1.2. Below is an example of our model correctly labeling an input video clip as "yoga"

It's interesting to note how the model alternates between "yoga" and "stretching leg"; both are technically accurate since, when you're in downward dog, you're practicing yoga and also stretching your legs. The following video, which our

Github Link - https://github.com/yasaswini8777/NNDL-Final-Project

Fig. 14. Yoga

Fig. 15. Skateboarding

human activity recognition model appropriately identifies as ”skateboarding,” is an example:

It is easy to understand why the model also predicted ”parkour” because the skater is performing an activity that a park visitor might do—jumping over a railing. Anyone hungry? If so, you might find ”making pizza” to be interesting. However, before to eating, make sure you’re “washing hands” before you sit down to eat: If you choose to indulge in “drinking beer” you better watch how much you’re drinking — the bartender might cut you off: 33

As you can see, considering how easy it was to modify ResNet to take 3D inputs instead of 2D, our human activity recognition model is still working fairly well, despite its imperfections. Although human activity recognition is still a challenging problem, advances in deep learning and convolutional neural networks are making significant progress.

F. RESULTS/ EXPERIMENTATION

You will discover how to use OpenCV and Deep Learning for Human Activity Recognition in this lesson. Depending on the job, our human activity identification model can identify more than 400 activities with an accuracy of 78.4–94.5 percent. A selection of the tasks is displayed below: Numerous activities include arm wrestling, baking, ice skating, driving a tractor, eating hot dogs, flying kites, getting tattoos, caring for horses, embracing, and more! Human activity recognition has several useful uses, such as:

- Classifying and categorising a dataset of videos on disc automatically.
- Assisting a new hire with task training and supervision so they can execute it appropriately (e.g., spreading out the dough, preheat the oven, adding sauce, cheese, toppings, etc.).
- Confirming that a worker in the food service industry has cleaned their hands after using the lavatory or handling food that may cross-contaminate (such as salmonella and chicken).
- Watching customers at bars and restaurants to make sure they aren’t overindulged.to become knowledgeable about using Deep Learning and Open CV for human activity recognition Human Activity Recognition Using Deep Learning and Open CV We’ll talk about the Kinetics dataset.

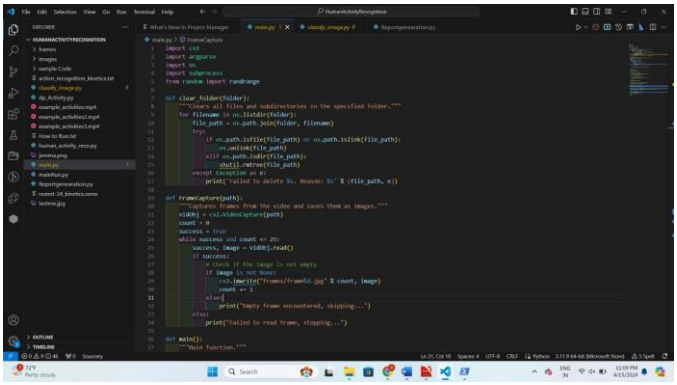

Fig. 16. main.py

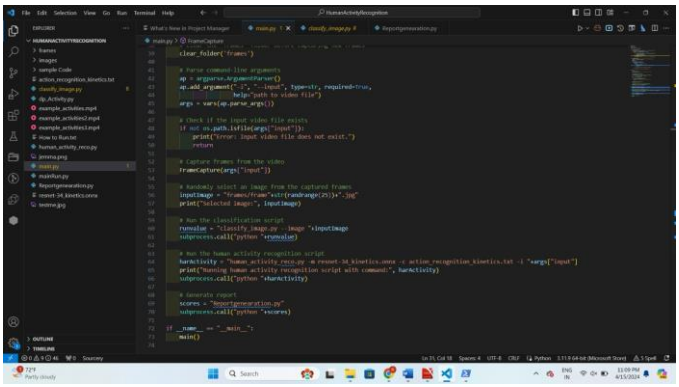

Fig. 17. main.py

Next, we’ll talk about extending Res Net—which normally makes use of 2D kernels—to make use of 3D kernels, allowing us to incorporate a spatiotemporal element that is utilised for activity identification. Next, we will use the Python programming language and the Open CV module to construct two variants of human activity recognition. We’ll examine the outcomes of applying human activity recognition to a few sample videos to conclude the course.

CODE SCREENSHOTS:

OUTPUT:

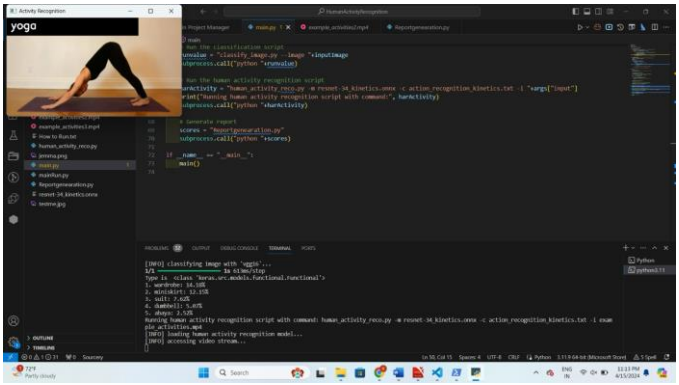

Fig. 18. Output1

Github Link - https://github.com/yasaswini8777/NNDL-Final-Project

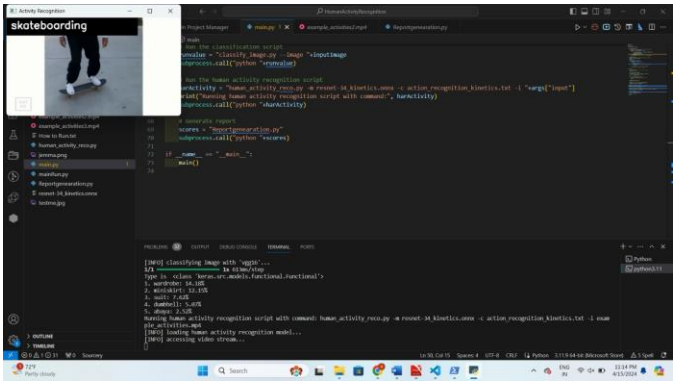

Fig. 19. Output2

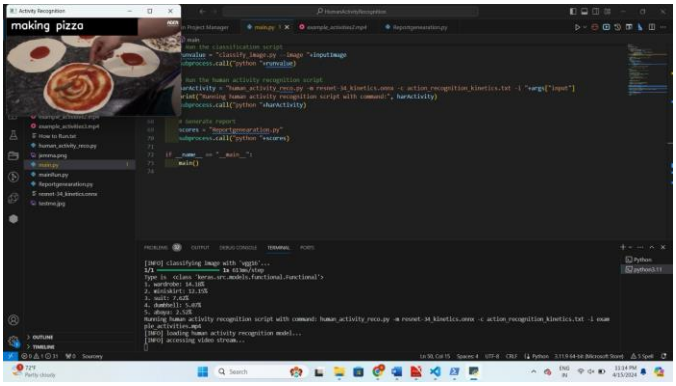

Fig. 20. Output3

G. REFERENCES

[1] Samundra Deep,Xi Zheng Department of Computing Macquarie University "**Sydney, Australia Conference"** Leveraging CNN and Transfer Learning for **Vision-based Human Activity Recognition**". International Acess number:19572619.DOI:10.1109/ITNAC46935 2019.9078016. TelecommunicationNetworks and Applications Date Added to IEEE Xplore:27 April 2020 INSPEC R. Alazrai, M. Hababeh, B.A. Alsaify, M.Z. Ali, M.I.Daoud, An End-to-End. Deep Learning Framework for Recognizing Human-to-Human Interactions. Using Wi-Fi Signals, The developed end to end deep learning model provides 86.3 accuracy for all human-to-human interactions recognition. The proposed model is not developed for group-to-group interactions. This model will work only for Human-to Human Interactions. IEEE Access. 8 (2020) 197695– 197710.

[2] T. Dobhal, V. Shitole, G. Thomas, G. Navada, Human Activity Recognition using Binary Motion Image and Deep Learning, Binary Motion Image Deep learning model gives good accuracy for both 2D and 3D datasets consistent speed of action performed by a human. The model does not give a reasonable detection rate if more than one person is involved in the 3D image. Procedia. Comput. Sci. 58 (2015) 178–185. M.M. Hassan, M.Z. Uddin, A. Mohamed, A.Almogren, A robust human activity recognition system using smartphone sensors and deep learning, KPCA outperform Support Vector Machine (SVM) and Artificial Neural Network (ANN). KPCA outperform Support Vector Machine (SVM) and Artificial Neural Network (ANN).Futur. Gener. Comput. Syst. 81 (2018) 307–313.

[3] R. Janarthanan, S. Doss, S. Baskar, Optimized unsupervised deep learning assisted reconstructed coder in the on-nodule wearable sensor for human activity recognition, improves the feature selection and extraction using an unsupervised deep learning model. The performances degrade in large datasets with different types of human activities.Meas. J. Int. Meas. Confed. 164 (2020) 108050. A. Jeyanthi Suresh, J. Visumathi, Inception ResNet deep transfer Learning model for human action recognition using LSTM, It provides the best accuracy score of 92 per cent and 91 per cent for the different data sets. It takes a tremendous amount of training time .Mater. Today Proc. (2020).

[4] Y. Jia, Y. Guo, G. Wang, R. Song, G. Cui, X. Zhong, Multi-frequency and multi-domain human activity recognition based on SFCW radar using deep learning, Neurocomputing. Developed deep learning model ncreases the recognition accuracy by 1.3% by additionally introducing the range maps. The proposed model is not developed for group- to- group interactions N. Zehra, S.H. Azeem, M. Farhan, Human activity recognition through ensemble learning of multiple convolutional neural networks, It takes less amount of preprocessing time because the proposed model support automatic feature extractions. Model is not suitable for concurrent activity recognition 2021 55th Annu. Conf. Inf. Sci. Syst. CISS 2021. (2021).

[5] S. Ullah, D.H. Kim, Sparse feature learning for human activity recognition, It provides long term dependencies. It provides less accuracy for the real-time data.Proc. - 2021 IEEE Int. Conf. Big Data Smart Comput. BigComp 2021. (2021) 309– 312.

[6] J. Schmidhuber, "Deep Learning In Neural Networks: An Overview", Neural Networks, vol. 61, pp. 85-117, 2015.

[7] D. C. Ciresan, U. Meier U, and L. M. Gambardella, "Deep, Big, Simple Neural Nets For Handwritten Digit Recognition", Neural computation, vol. 22, no. 12, pp. 3207-3220, 2010.

[8] Kay W, Carreira J, Simonyan K, Zhang B, Hillier C, Vijayanarasimhan S, Viola F, Green T, Back T, Natsev P, "The Kinetics Human Action Video Dataset", 2017.

[9] Yu Zhao, Rennong Yang, Guillaume Chevalier, Ximeng Xu, and Zhenxing Zhang, "Deep Residual Bidir-LSTM for Human Activity Recognition Using Wearable Sensors," Mathematical Problems in Engineering, Volume 2018.

[10] D. Das and A. Chakrabarty, "Human Gait-Based Gender Identification System Using Hidden Markov Model And Support Vector Machines," in Conf. Comput. Commun. Autom. ICCCA 2015, pp. 268–272, 2015.

[11] N. Y. Hammerla, S. Halloran, and T. Ploetz, "Deep, Convolutional, And Recurrent Models For Human Activity Recognition Using Wearables," arXiv preprint arXiv:1604.08880, 2016.

Github Link - https://github.com/yasaswini8777/NNDL-Final-Project

[12] S. M¨unzner, P. Schmidt, A. Reiss, M. Hanselmann, R. Stiefelhagen, and R. D¨urichen, “Cnn-Based Sensor Fusion Techniques For Multimodal Human Activity Recognition,” in Proceedings of the 2017ACM International Symposium on Wearable Computers, ser. ISWC ’17. New York, NY, USA: ACM, 2017, pp. 158–165.

[13] M. Panwar et al., “CNN Based Approach For Activity Recognition Using A Wrist-Worn Accelerometer,” in Proc. Annu. Int.Conf. IEEE Eng. Med. Biol. Soc. EMBS, no. July, pp. 2438–2441, 2017.

[14] A. Jain and V. Kanhangad, “Human Activity Classification in Smartphone’s Using Accelerometer and Gyroscope Sensors,” IEEE Sensors Journal, vol. 18, no. 3, pp. 1169-1177, 1 Feb.1, 2018.

[15] A. Ignatov, “Real-Time Human Activity Recognition From Accelerometer Data Using Convolutional Neural Networks,” Appl. Soft Comput. J., vol. 62, pp. 915–922, 2018.

[16] L. Sifre, “Rigid-motion Scattering for Image Classification,” PhD thesis, Department of Informatics, CMP Ecole Polytechnic, France., 2014.

[17] A. Krizhevsky, I. Sutskever, andG. E.Hinton, “Imagenet Classification With Deep Convolutional Neural Networks,” in Proceedings of the 26th Annual Conference on Neural Information Processing Systems (NIPS ’12), pp. 1097–1105,Lake Tahoe, Nev, USA, December 2012.

[18] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, “Energy Efficient Smartphone-Based Activity Recognition Using Fixed-Point Arithmetic,” Journal of Universal Computer Science, vol. 19, no. 9, pp. 1295–1314, 2013.

[19] N Srivastava, E Mansimov, R Salakhudinov,” Unsupervised Learning Of Video Representation Using LSTM”, International conference on machine learning, pp 843–852, 2015.

[20] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt, “Action Classification In Soccer Videos With Long Short Term Memory Recurrent Neural Networks”, in Proceedings of ICANN, 2010.

[21] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt. Sequential Deep Learning For Human Action Recognition,” Human Behavior Understanding, 2011.

[22] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell., “Long-Term Recurrent Convolutional Networks For Visual Recognition And Description”. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 2625–2634, 2015.

[23] A. Karpathy, G. Goderici, S. Shetty, T. Leung, R. Sukthankar, and F. F. Li., “Large-Scale Video Classification With Convolutional Neural Networks,” In Proceedings of CVPR, 2014.

[24] K. Simonyan and A. Zisserman, “Two-Stream Convolutional Networks For Action Recognition In Video.” In arXiv preprint arxiv:1406.2199, 2014.

[25] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen., “Mobilenetv2: Inverted Residuals And Linear bottlenecks.” In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 4510–4520. IEEE, 2018.

LaTeX

Github Link - https://github.com/yasaswini8777/NNDL-Final-Project