
TeacherActivityNet:

A Personalized Assistive System for Enhancing Cognitive Support and Improvement of Work-Life Balance in Academic Settings

By

Shoib Ahmed Shourav - 0122420022

Submitted in partial fulfilment of the requirements
of the degree of Master of Science in Computer Science and Engineering

August 26, 2025



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
UNITED INTERNATIONAL UNIVERSITY

Abstract

This research establishes a new idea that supports cognitive enhancement and work-life balance for academic office-based faculty members. The improvements in computer vision led to the creation of work-based wellness solutions which assist educational settings to manage their staff's mental health requirements between intellectual requirements. Workload management methods and self-evaluation approaches from traditional times demand significant manual efforts while creating efficiency problems thus demonstrating the value of automated video-based support systems as better alternatives. Scientists have achieved notable improvements in Human Activity Recognition (HAR) detection yet studies about HAR based personalized workplace support systems designed for faculty office use are scarce. Current research examines student performance in classrooms rather than providing essential support for faculty cognitive processes and wellbeing. Therefore there exists an important deficiency in cognitive and wellbeing assistance for educators. The paper fills this gap through TeacherActivityNet, our proposed model with the help of newly created Workplace Activity Recognition Dataset which represents a new video database designed to detect nine different faculty office activities such as Arriving, Counselling, Eating, Idle, Leaving, Sleeping, Talking, Using_Phone and Working. We adapted the YOLOv8n architecture to develop TeacherActivityNet for our model implementation that achieved 0.841 mAP50 through fine-tuning with our generated dataset. Comparison against existing systems reveals that TeacherActivityNet brings potential advantages to personal assistive systems by supporting better academic cognitive performance and life-work balance within educational institutions.

Acknowledgements

This research would not have been possible without the invaluable contributions of my two supervisors, Prof. Dr. Swakkhar Shatabda and Mr. Nahid Hossain. I had the privilege of working with Dr. Swakkhar Shatabda for over a year, during which we carefully explored and refined the research topic. With his exceptional expertise in computer vision, he quickly identified and proposed this research direction. His insightful guidance and visionary thinking have been instrumental in shaping this work. Mr. Nahid Hossain has been a pillar of support from the very beginning. His mentorship has played a crucial role in my growth as a researcher. His unwavering support and dedication have been invaluable, and I am deeply grateful for his continuous encouragement throughout this journey.

I would also like to express my sincere gratitude to Prof. Dr. Md. Hasanul Kabir, the head examiner and reviewer of this research work, for his valuable time, constructive feedback, and insightful suggestions that have helped improve the quality of this work. Additionally, I extend my heartfelt appreciation to Prof. Dr. Khondaker Abdullah-Al-Mamun, the current Director of the MSCSE Program (Professor, Department of CSE, Director, MSCSE Program & Director, IRIIC), for his continuous support and leadership, which have greatly contributed to my academic journey. I also wish to acknowledge Prof. Dr. Md. Motaharul Islam, former Director of the MSCSE Program, for his guidance and encouragement during his tenure. Furthermore, I am grateful to the entire MSCSE program office for their unwavering willingness to help. Their readiness to assist whenever needed has made a significant impact on my research experience.

Finally, I am profoundly thankful to my family for their constant emotional support. Their encouragement has been my strength.

Contents

Table of Contents	iv
List of Figures	v
List of Tables	vii
1 Introduction	1
1.1 HAR Applications	1
1.2 Motivation	2
1.3 Objectives	3
1.4 Contribution	3
1.5 Organization of the Report	4
2 Background	5
2.1 Related Works	5
2.2 Gap Analysis	7
3 Dataset	8
3.1 Overview of the Dataset	8
3.2 Ethical Considerations and Privacy Protocols	8
3.2.1 Written Informed Consent	8
3.2.2 Opt-out Policy	9
3.2.3 Privacy Protection In Model Deployment	9
3.3 Collection	9
3.3.1 Challenges of Data Collection	10
3.4 Annotation	11
3.4.1 Challenges of Annotation	11
4 Methodology	16
4.1 Proposed Architecture	16
4.2 Dataset Augmentation and Preparation	16
4.3 TeacherActivityNet	17
4.3.1 Reason Behind the Residual Connection in the Last Layer	18

4.4	Spatio-Temporal Feature	19
4.4.1	Action Class Wise Advantages	20
4.5	Prediction from Videos	20
5	Experimental Setup	22
5.1	Test and Validation Results with Confusion Matrix	22
5.1.1	Hyper Parameter Settings	22
5.1.2	Evaluation Metrics: mAP@50 and mAP@50-95	23
5.2	Spatio-Temporal Feature Tuning	25
5.2.1	Reason Behind no Improvement in Adaptive Analysis	28
5.3	Confidence Curves with Train and Validation Loss	29
5.4	Model Predictions on Validation and Test Images	31
5.5	Faster RCNN Model Performance	36
5.6	Model Performance on Videos	36
5.7	Ablation Studies	37
5.8	YOLO Models vs TeacherActivityNet	38
5.9	Activity Recognition Model Performance	41
5.9.1	Causes of Poor Activity Recognition Model Performance	41
6	Conclusion	43
6.1	Summary	43
6.2	Limitation	44
6.3	Future Work	44
	References	50

List of Figures

2.1	Overview of our related work	7
3.1	An object-invariant and diverse dataset	11
3.2	Ensuring the dataset is diverse and contains multiple nearby objects	12
3.3	Annotated images of training dataset	13
3.4	Ambiguity between the actions Leaving and Arriving	14
3.5	Ambiguity between the actions Sleeping and Idle	15
4.1	Workflow of our proposed method	16
4.2	Before augmentation	17
4.3	After augmentation	17
4.4	Modified backbone of our proposed TeacherActivityNet	18
4.5	Flow chart for prediction on sample test dataset video with time count for each action class	19
4.6	Comparison of Original Image and Predicted Head & Wrist Positions	20
5.1	Confusion matrix for validation and test datasets of TeacherActivityNet . .	26
5.2	Normalized confusion matrix of TeacherActivityNet	27
5.3	Confusion matrix for Faster R-CNN model for test dataset	28
5.4	Comparison of Adaptive Window Size and Improvement Rate	29
5.5	F1, Precision, and Recall Curves for All Activities	30
5.6	Train and Validation Loss with Metrics Evaluation	30
5.7	Comparison between ground truth labels and model predictions on valida- tion dataset for batch-0	31
5.8	Comparison between ground truth labels and model predictions on valida- tion dataset for batch-1	32
5.9	Comparison between ground truth labels and model predictions on valida- tion dataset for batch-2	33
5.10	Comparison between ground truth labels and model predictions on test dataset for batch-0	33
5.11	Merged view of ground truth labels and model predictions side by side for test dataset for batch-0	34

5.12 Merged view of ground truth labels and model predictions side by side for test dataset for batch-1	34
5.13 Comparison between ground truth labels and model predictions on test dataset for batch-1	35
5.14 Comparison between ground truth labels and model predictions on test dataset for batch-2	35
5.15 Comparison between original and cropped image of action class Working . .	42

List of Tables

3.1	Action class definition with duration and total instances in training dataset	14
5.1	Box precision, recall, mAP50 and mAP50-95 results for test dataset using TeacherActivityNet	24
5.2	Box precision, recall, mAP50 and mAP50-95 results for validation dataset using TeacherActivityNet	24
5.3	The mAP50 for test dataset using Faster RCNN ResNet50 FPN	36
5.4	Predicted frame counts for real time video from test dataset using the model	37
5.5	Comparative baseline analysis with our TeacherActivityNet model	38
5.6	Box precision, recall, mAP50 and mAP50-95 results for test dataset using YOLOv5 for each action class	38
5.7	Box precision, recall, mAP50 and mAP50-95 results for test dataset using YOLOv8 for each action class	39
5.8	Box precision, recall, mAP50 and mAP50-95 results for test dataset using YOLOv9 for each action class	39
5.9	Box precision, recall, mAP50 and mAP50-95 results for test dataset using YOLOv10 for each action class	39
5.10	Box precision, recall, mAP50 and mAP50-95 results for test dataset using YOLOv11 for each action class	40
5.11	Box precision, recall, mAP50 and mAP50-95 results for test dataset using YOLOv12 for each action class	40
5.12	Summary of the performance comparison of YOLO-based models with Teacher-ActivityNet	40
5.13	Comparing activity recognition models	41

Chapter 1

Introduction

We have developed our concept from Human Activity Recognition (HAR). Before developing the personalized assistive system within academic offices it becomes crucial to grasp other current uses of HAR technology. Our main research goal is to help educators inspect their office hour procedures for maximizing mental capacity and creating health-focused life-work balance schemes. Through this strategy educational quality should increase because it advances the student-teacher relationship. The TeacherActivityNet model stands necessary to serve professional requirements in academic institutions while extending its value across professional office dynamics. The dataset reveals workplace activity patterns to support workplace wellbeing together with efficiency promotion. Workplace effectiveness receives better objective support by using HAR-based systems which minimize potential bias from manual observation processes.

1.1 HAR Applications

HAR applications are gaining attention in many aspects of our daily life. Significant advancements have been made in HAR for sports, healthcare, agriculture, fitness, surveillance, and more. In recent years, considerable amount of work has been done in the field of Human Activity Recognition in various contexts such as sports [1, 2, 3, 4, 5, 6, 7, 8, 9], pose estimation [10, 11, 12], crime scene detection [13, 14], healthcare [15, 16, 17, 18], etc. While solving the problem of human activity recognition from videos, the researchers primarily focus on a few aspects - analyzing the applicability of Machine Learning (ML) and Deep Learning (DL) methods in different HAR tasks, creating new datasets for domain-specific HAR-related tasks, and proposing novel solutions for diverse HAR problems. Educational organizations have been the center of experimentation to introduce automation using various AI methods to streamline processes [19]. [20] have critically analyzed the use of AI and emerging technologies in the classroom and recommended that the computer vision-based surveillance system can ensure the safety and security of the classroom alongside tracking the students' participation and attendance. Additionally, a smart surveillance system can help teachers in plagiarism detection and student supervision in an online

setting [21]. These systems can also help identify abnormal activities on school premises through CCTV footage analysis [22].

The advancement of computer vision models has opened up new avenues for task automation, including enhancing the work-life balance and cognitive support of faculty members. It can be challenging and time-consuming to analyze teacher actions in educational institutions, such as improving office hours' efficacy, promoting mental health, and boosting student involvement. Moreover, traditional methods of self-management and time tracking are not only time-consuming but also prone to errors and inefficiencies. To address these challenges, automated assistive systems based on video analysis can offer a promising solution for supporting faculty wellbeing and optimizing office activities.

However, intelligent video surveillance systems in educational organizations primarily focus on classroom and student activity tracking. There has been a lack of existing work on analyzing teachers' activities in an office setting. On the other hand, to solve the HAR problems, the majority of the researchers have adopted machine learning and deep learning-based techniques like CNN [23, 24], RNN [25, 26], LSTM [27, 28], SVM [29], Naive Bayes [30], older versions of You Only Look Once (YOLO) [31], and so on. The use of the latest version of YOLO might have good prospects in office surveillance. Convolutional Neural Networks (CNNs) can robustly detect spatial aspects in sensor/image data, are therefore an excellent solution to HAR. However, it needs large amounts of labelled data and computational power. Recurrent Neural Networks (RNNs) and Long Short-Term Memory networks (LSTMs) work well in modeling temporal situations in sequential HAR data; LSTMs reduce the issue of the vanishing gradient proving RNNs; although, this can be cumbersome and often time expensive to train. Support Vector Machines (SVMs) can work with smaller data with robust performance and have lower computation requirements but it strongly depends on hand-crafted features and might fail to work with very large and complex data. Naive Bayes possess a simple and a fast characteristic that may oversimplify feature dependency that would limit accuracy in the HAR tasks. Previous versions of YOLO based models are also very useful in terms of real-time capability in object detection for vision-based HAR but are not very accurate in overlapping actions. Our work aims to address this gap in how YOLO would efficiently recognize the teachers' activities in their offices.

1.2 Motivation

There are several research done on workplace activity analyzing systems [32, 33, 34, 35, 36, 37, 38, 39] using computer vision but these are mostly related to worker safety related research on construction sites of any other workplace. These are mainly on site worker activity tracking for their safety and security. There are some workplace performance analyzing systems available which only tracks employee's time in the specific region or how they are using the total time, effectively or ineffectively [40, 41]. No research exists about personalized assistive systems for academic faculty during their office duties and

researchers lack adequate datasets regarding this topic. The main driving force for our work stems from this recognition. We strongly supports building a new dataset and the TeacherActivityNet model since such work will enhance educational environments and spark other office workplace assistive system implementations. The implementation of activity recognition technology enables better management of time which leads to enhanced performance together with increased well-being as well as productivity levels. Academic offices benefit from enhanced teacher-student interactions in scheduled meetings although broader professional spaces experience better operational practices together with enhanced work and life equilibrium.

1.3 Objectives

We have many objectives of this research and we have listed the important ones here:

- Creating a video dataset of teacher activities in their offices which consists of nine action classes.
- Modifying YOLOv8n to build an efficient detection model TeacherActivityNet to analyze teacher activities.
- Compare image prediction performances with video prediction performances
- Providing an comparative analysis of our proposed methods alongside with other existing solutions.

1.4 Contribution

We have some contribution to this research and we have highlighted the important ones here:

- A new video dataset, known as Workplace Activity Recognition Dataset, is introduced that contains nine types of activities of teachers in offices. Such a dataset, as far as we know, does not exist. We also mention the unsurpassed peculiarities of the dataset made up, which are the variety of activities and the intricacy of annotations.
- In our paper, we have already modified and fine-tuned the YOLOv8n model to train so that our TeacherActivityNet model becomes an efficient model to detect teacher activity during office settings. We contrast it with multiple recent versions of YOLO, mainly with YOLOv11 and YOLOv12, and our model in some cases, outperforms them.
- We have also provided both image-based predictions with video-based predictions to give useful information on strengths of both methods as well as limitations.

- We offer a comparative study of our proposed model against known detection models like Faster R-CNN and other state-of-the-art activity recognition models, establishing the effectiveness of TeacherActivityNet model in academic office-based faculty activity recognition.

1.5 Organization of the Report

The contents of the report are structured as follows: Chapter 2 presents related works, Chapter 3 discusses the dataset creation challenges, annotation, augmentation and final preparation, Chapter 4 covers the methodology, Chapter 5 analyzes the results and shows ablation studies, Chapter 6 provides the conclusion.

Chapter 2

Background

In the background study of the research work, we are going to study the relevant articles. We will also try to find the possible gaps that we can work on and propose solutions to the new challenges we face.

2.1 Related Works

Human Action Recognition (HAR) is a critical research area with applications across security, surveillance, sports, education, and more. Many studies have focused on improving the prediction and recognition of various activities using both machine learning (ML) and deep learning (DL) techniques applied to image and video data.

A wireless method named the Zigbee technique was proposed to track employee activity using computer vision [41]. The dataset used in this process consists of videos collected from CCTV cameras in the workplace. In order to find the efficiency of their system, they compared their measured time with the actual in and out time collected from biometric entry. However, the system is only supposed to function for a specific region. Sikder et al. [42] introduced the KU-HAR dataset containing videos of 90 participants performing 18 different actions. Using a Random Forest classifier, they achieved a precision of 90%. These ML approaches benefit from interpretability and lower computational cost but may be limited in modeling complex temporal patterns.

Image-based DL approaches have also advanced significantly. Rashmi et al. [43] collected 688 image frames and over 54,000 samples from CCTV cameras in computer laboratories, using YOLOv3 for student action detection. This method provides real-time detection but requires extensive labeled data. Kumar [44] used Multi-task Cascaded Convolutional Networks (MTCNN) for facial expression-based student engagement, demonstrating DL's strength in automatic feature extraction.

In the sports domain, Host et al. [45] provided an overview of ML and DL techniques applied mostly to ball-based sports using video datasets. Latha et al. [46] proposed CNN and LSTM models using the UCF-50 video dataset for activity recognition. Xiao et al. [2] introduced deformable convolution and adaptive multi-scale feature methods to analyze

sports videos from datasets like UCF Sports, UNF 50, and YouTube (UCF 11). Goh et al. [47] focused on fault detection in badminton using YOLOv5 trained on 1,900 images, achieving higher accuracy than human judges.

Video-based HAR using ML and DL techniques has also been explored extensively. Singh et al. [48] applied CNN and RNN with Inception V3 on the UCF crime dataset of 1,800 videos to predict crime scenes, highlighting the importance of large datasets and data augmentation. Shreyas et al. [49] proposed a 3D CNN for anomalous human activity detection on the same dataset, outperforming SVM and binary classifiers. Nale et al. [50] used pose estimation and LSTM on the NTU-D 60 dataset, evaluating geometrical relations of skeletal joints to detect suspicious activity. Liu et al. [22] proposed Temporal Segment Transformers and Video Swin Transformer models to recognize abnormal behavior on the CABR50 dataset. Nandhini et al. [51] focused on crime scene detection using a stacked hourglass method and Gaussian classifier, achieving over 90% accuracy on the test set.

Other research on security-related HAR includes Selvi et al. [52] who proposed an Enhanced CNN to predict suspicious actions using multiple datasets such as CCTV footage, CAVIAR, and DCSASS. Rahman et al. [53] developed a system to predict possible anomalies in smart homes. Menaka et al. [54] applied YOLOv3 for real-time suspicious activity detection in ATM surveillance videos. Ankalaki et al. [55] introduced the OP-Tanish activation function in 1D-CNN, outperforming traditional functions like ReLU and SWISH for anomaly detection.

In fitness-related activity recognition, Yang et al. [56] compared Pose-Based Branch (PBB) and RGB-Based Branch (RBB) features using CNN, ResNet152, and 3D-CNN on several datasets, concluding that pose-based methods outperform RGB-based ones. Goh et al. [47] applied YOLOv5 for fault detection in badminton matches, demonstrating high accuracy.

Work-life balance and mental wellbeing have been addressed in fewer studies. Mulyani et al. [57] investigated children's behavior and teachers' work-life balance in school climates. Abdulaziz et al. [58] collected data from 278 teachers and found that work-life balance positively affects organizational commitment, while work overload negatively impacts it.

Machine learning approaches are usually less demanding in terms of amount of data and processing and can be interpretable but might be less efficient in coping with complex spatial-temporalities. Deep learning can find better results on huge and complex data. Videos can be more easily labeled because of the strong correspondence between frames, but cannot provide temporal information, whereas image-based methods can provide temporal information, but such methods require more complex modeling or more data. Appearance-insensitive pose or skeleton based features are also sturdy and have long-term resistance but require reliable pose guessing. We have given an overview of the summary of our related work in the below given Figure 2.1.

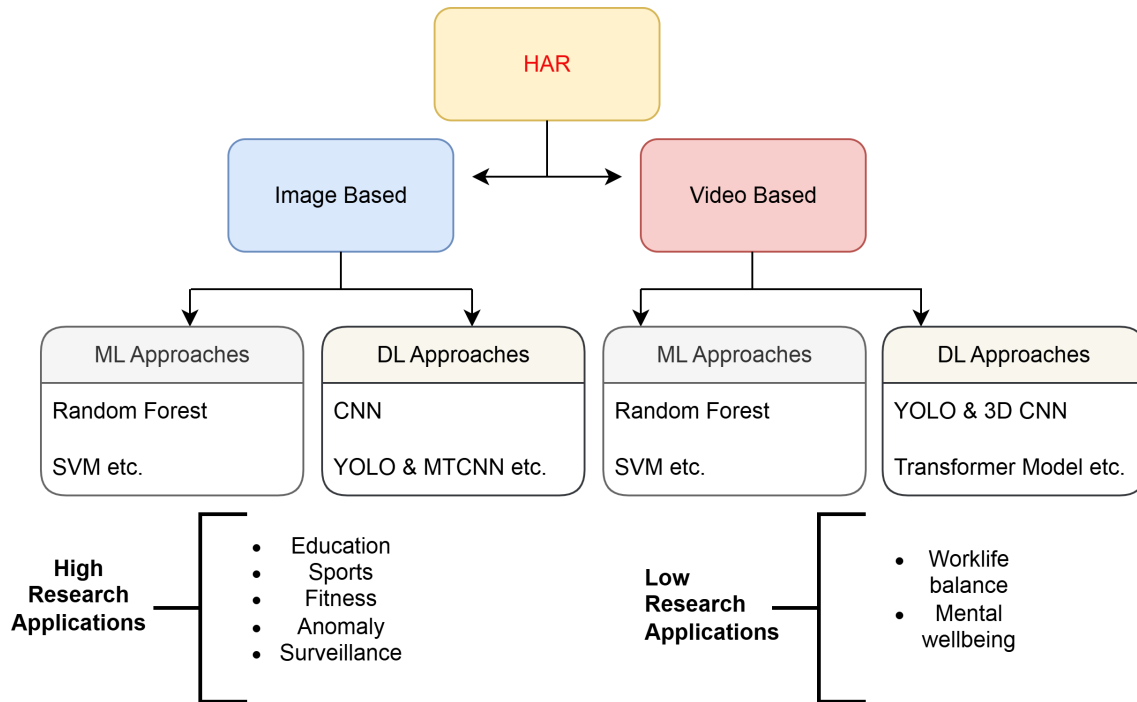


Figure 2.1: Overview of our related work

2.2 Gap Analysis

There is currently no available dataset for analyzing the activities of faculty or teachers in office settings. We aim to create this dataset to help track teachers' activities in office environments, addressing issues such as cognitive overload, time mismanagement, and mental wellbeing challenges.

Chapter 3

Dataset

As the gap analysis revealed that there is no teacher activity analyzing dataset available, we will create one named Workplace Activity Recognition Dataset. We will aim to include diversity in the dataset and ensure it is helpful to many researchers.

3.1 Overview of the Dataset

One of our major contributions is the creation of a dataset that includes a wide range of activities for the task of teacher activities. To accomplish this, we collected videos of 19 participants for 9 action classes. The action classes are Arriving, Counselling, Eating, Idle, Leaving, Sleeping, Talking, Using_Phone and Working. For the diversification of the dataset, the videos were taken from 8 different rooms. The videos were recorded using an iPhone 11 Pro Max, with varying camera distances based on room size and diverse angles capturing multiple participants' actions. The cameraman's average height was 5'6". The dataset includes 19 participants (3 female), mostly aged 20–22, with some around 30. All actions were controlled and guided. There are a total of 147 videos, out of which 110 are used for training, 27 for validation, and 10 for the test dataset. In our dataset, the train, validation, and test split ratio is approximately 75% : 18% : 7%, respectively.

3.2 Ethical Considerations and Privacy Protocols

The participants were asked to sign an informed consent form before making the videos, and since we intend to make the data public for research purposes, there will be no opt-out policy. Below are some key points outlining how privacy will be maintained.

3.2.1 Written Informed Consent

We have collected written informed consent from every participant during the data collection process. They have been informed about how the data will be used in research, how it will be shared, who will have access to it, and for how long.

3.2.2 Opt-out Policy

Since we intend to publish the data for all independent researchers to use, we believe that participants are only allowed to opt-out before we publish the data.

3.2.3 Privacy Protection In Model Deployment

Some may raise concerns about the faculty activity analyzing system and its privacy implications. We will only run our model on real-time CCTV camera footage (as shown in Section 4.5) to determine the total duration of each action class. We do not need to store the CCTV recordings. This approach helps us avoid privacy issues in office rooms.

For example, if a faculty member in their office performs only the nine action classes listed in Table 3.1, they will be notified only of the total time spent on those nine actions. If a faculty member engages in an activity that is not included in the defined action classes, our model will classify such actions as part of the background class. As a result, no one will be able to determine what action was performed at that time.

Therefore, if a faculty member is praying or engaging in any private activity, it will be counted as part of the background class, and the specific action will remain unknown. In this way, we can maintain their privacy while still allowing the model to operate effectively in office settings. Since we are only focused on the nine defined action classes, we do not track or analyze any other activities the faculty member may be performing in their room.

3.3 Collection

The dataset creation process took place primarily in two stages: training and validation, and testing. In the first stage, we selected a sample of 12 participants, each asked to perform one action at a time. Nine videos were recorded for each action class per participant. After collecting the training videos, we selected a sample of 3 participants from the remaining 7 and asked them to perform the same actions. The video sets of these three participants across the nine classes were used as the validation set in our experiments.

In the second stage, we recorded 10 videos of the remaining 4 participants who were not involved in the previous stage of data collection. Since we are interested in measuring how accurate our models are in real-time, unlike the previous stage where each action was performed in separate videos, we asked the participants to perform the actions continuously for a specific period. To rigorously test model performance in real-time settings, most of the videos were recorded in rooms that were not used during the earlier phase of data collection.

Table 3.1 provides descriptions of the action classes, including their duration and the number of instances after annotation.

3.3.1 Challenges of Data Collection

We faced many challenges during the data collection process. We have divided these challenges into two sections, as outlined below.

3.3.1.1 Background Problem

We initially assumed that the actions would be independent of the background in the images, as all actions ideally should be. Based on this assumption, we recorded videos of around 10 participants in a single room at the same teacher’s desk. We then began the annotation and augmentation process to prepare our dataset for training.

After training the model, we proceeded to test its performance on a new dataset, which was recorded in different rooms. However, our trained model failed to accurately predict the action classes due to the consistent background in the training dataset and the differing backgrounds in the test dataset.

We realized that the training dataset also needs to include a variety of backgrounds. To overcome this first challenge, we had to recreate the dataset with greater background diversity.

3.3.1.2 Actions Dependent on Objects

The first problem with actions becoming dependent on objects occurred when a mobile phone was visible on the desk. Although we now have different backgrounds in the training dataset, some actions still caused problems. We have two action classes, `Using_Phone` and `Talking`, where participants use mobile phones. In the other action classes, they do not use the mobile phone. Therefore, in the remaining action classes, we instructed the participants to keep their mobile phones in their handbag or pocket.

However, we need to create diverse dataset so that it is object-invariant. We have kept a mobile phone on the desk to make sure that the dataset is not dependent on these objects for any action class. In Figure 3.1, we can see that the action class is not `Using_Phone`, but `Working`.

The solution was to keep the mobile phone on the desk even if the action being performed did not involve the use of the phone.

The second problem arose when we recorded videos for the action class `Eating`. The only time we included food, a water bottle, or a tiffin box was when they were visible on the desk. In other action classes, these items were absent. To solve this issue, we made sure to include these items while performing other action classes as well.

In Figure 3.2, we can see that the participant is performing the `Idle` action class, but we included a tiffin box, water bottle, a laptop, and the participant also has a mobile phone in hand. We have tried our best to keep the dataset diverse. We also ensured that the `Counseling` action class involved multiple students, with some joining the counseling session and others leaving.



Figure 3.1: An object-invariant and diverse dataset

3.4 Annotation

After collecting the dataset, we generate the image frames from the training and validation videos using Roboflow¹ through manual annotation. Three annotators performed manual labeling, with a verifier ensuring the accuracy of all annotations. From each frame, a bounding box is drawn to annotate the object with an action class. Extra caution is exercised while annotating the images that look the same but fall into different action classes. For example, in Leaving and Arriving action classes, there is a moment in the videos when the participants leaving the seat and taking the seat, look almost identical. Another case is while performing Idle action, the participants closed their eyes which potentially conflicts with Sleeping action. Taking these ambiguous scenarios into consideration, such frames are discarded from the dataset. After annotating, the numbers of instances in the training, validation, and test sets are 6498, 1337, and 808, respectively. In Figure 3.3 we have shown the annotated images for training.

3.4.1 Challenges of Annotation

We had three annotators and one reviewer. We held meetings to discuss the details of the action classes, ensuring we were all on the same page and identifying the action classes in the same manner. After the first annotation was completed, we checked our model's

¹<https://roboflow.com>

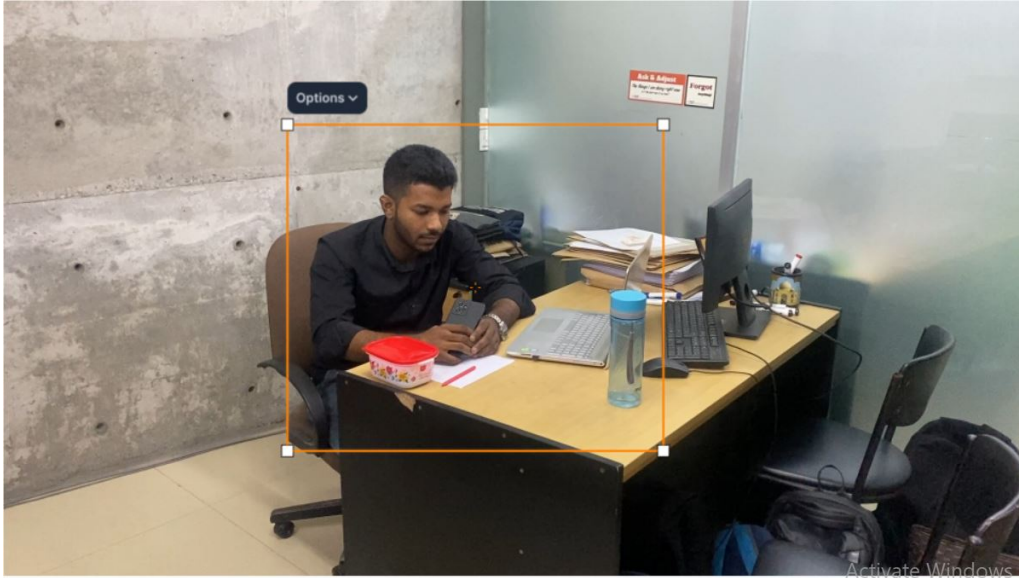


Figure 3.2: Ensuring the dataset is diverse and contains multiple nearby objects

performance across the nine action classes. We concluded that some of the action classes were performing poorly due to poor labeling.

In the action classes Leaving and Arriving, we observed similar positions where the participant was either trying to take a seat or trying to get off the seat to leave the room. As a result, the model became confused by these types of image frames. Initially, we discarded such frames. In Figure 3.4, we can hardly predict whether the participant is leaving or arriving in the room. Later, we decided to keep these frames, as we believe we can address this issue by applying spatio-temporal focus during model training.

There were also other action classes that overlapped. For example, while performing the Idle action, some participants closed their eyes, which made it appear as though they were sleeping. These image frames caused the model to perform poorly in the Idle action class. We discarded frames like these. In Figure 3.5, the participant is looking down while performing the Idle action, but it appears as though their eyes are closed. In such frames, the model becomes confused about the action class.

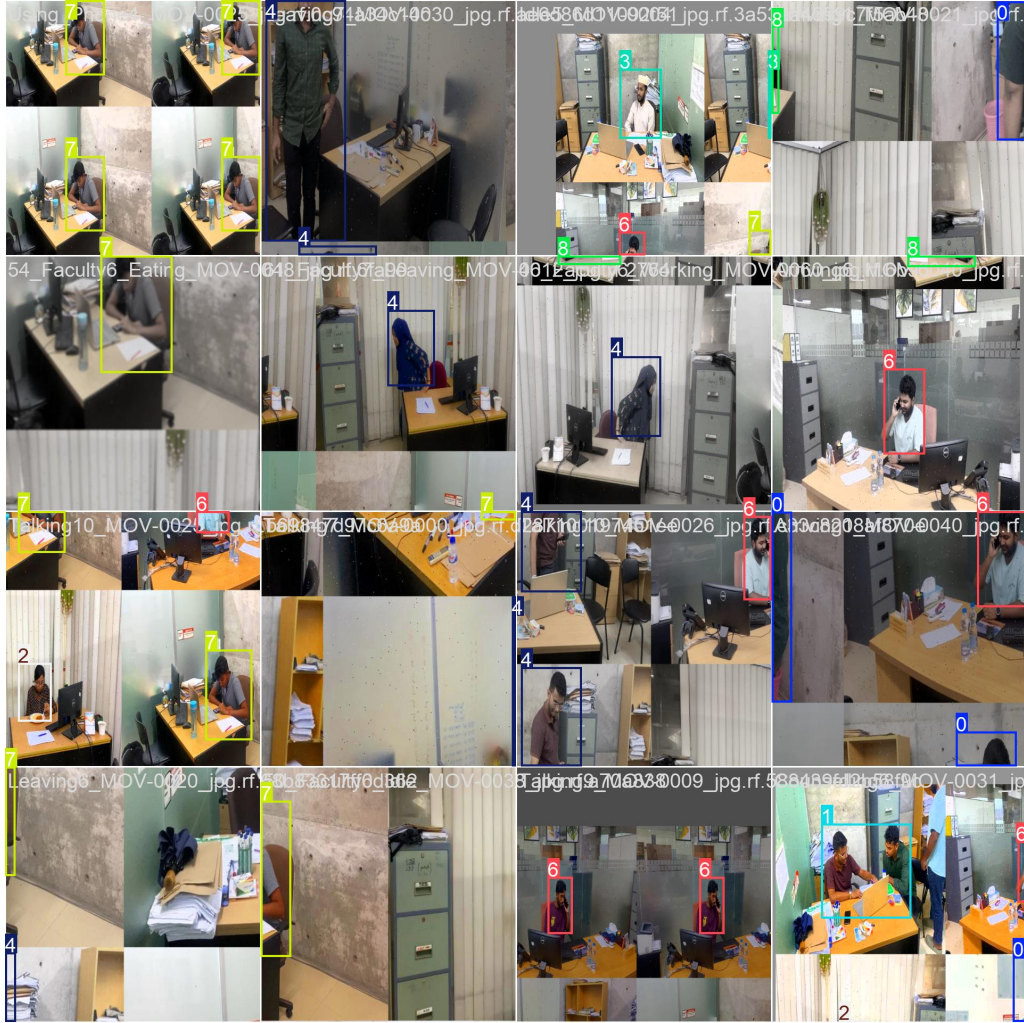


Figure 3.3: Annotated images of training dataset

Table 3.1: Action class definition with duration and total instances in training dataset

Class Name	Class ID	Definition of the Action Class	Duration Per Video	Total Instances
Arriving	0	Person arriving in the room and taking his/her seat.	5 seconds	665
Counselling	1	Teacher giving counselling time to multiple students.	30/60 seconds	728
Eating	2	Person eating or drinking.	10 seconds	739
Idle	3	Idle for some time, doing none of the other mentioned actions.	30/60 seconds	755
Leaving	4	Person leaving his/her seat and going out of the room.	5 seconds	654
Sleeping	5	Person sleeping in two positions: laying his/her head on the chair or putting their head down on the table.	10 seconds	739
Talking	6	Talking via mobile phone (not considering talking to persons).	30/60 seconds	727
Using_Phone	7	Using his/her phone while sitting in the chair or standing.	30/60 seconds	748
Working	8	Focused on the computer/laptop while using mouse or keyboard.	30/60 seconds	743

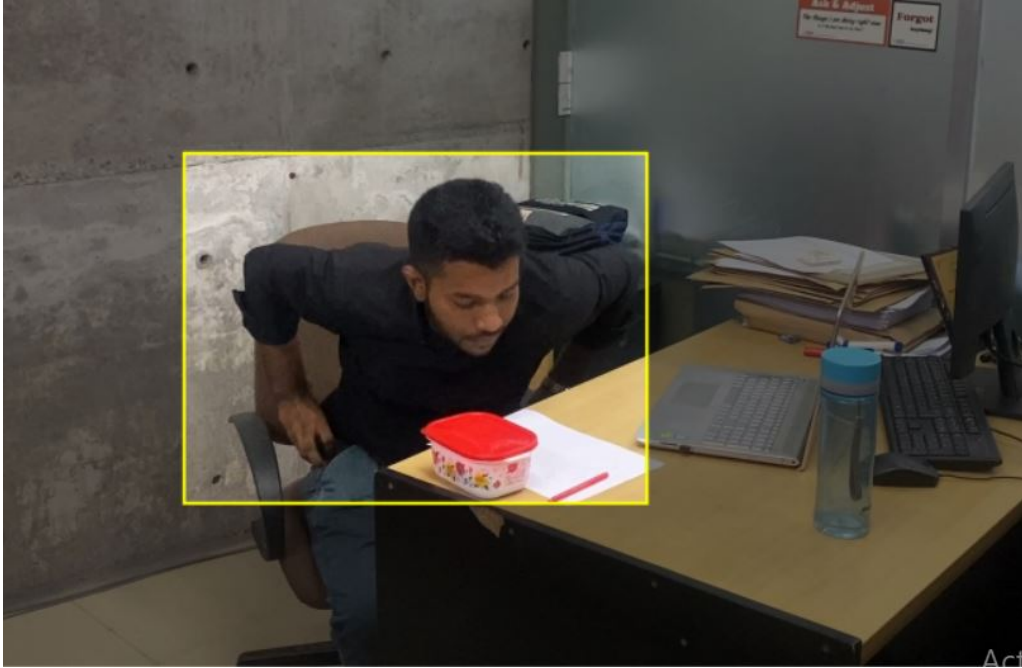


Figure 3.4: Ambiguity between the actions Leaving and Arriving



Figure 3.5: Ambiguity between the actions Sleeping and Idle

Chapter 4

Methodology

4.1 Proposed Architecture

Our work consists of three major phases - creating the dataset which already has been explained in Chapter 3, modifying the YOLOv8 model architecture to obtain better performing detection model, and proposing a method for real-time prediction. Figure 4.1 presents the step-by-step descriptions of our work. Initially, we curate our dataset followed by preprocessing it. After that, we make modifications to the YOLOv8n model architecture to introduce the TeacherActivityNet. We train and fine-tune the TeacherActivityNet model to predict the action classes from the videos of the faculty members. Besides, we measure action class-wise time for each faculty member through face recognition.

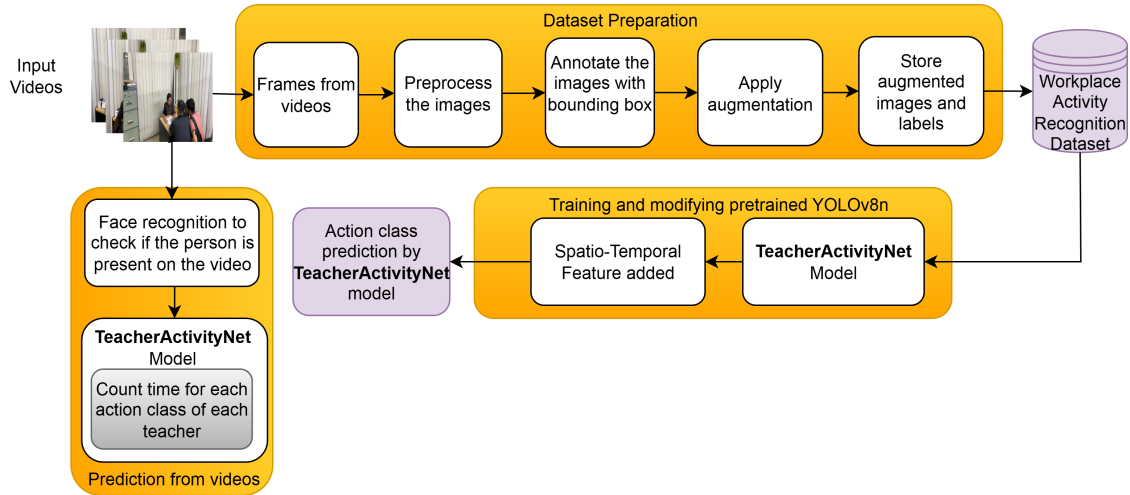


Figure 4.1: Workflow of our proposed method

4.2 Dataset Augmentation and Preparation

The detailed dataset preparation process will be presented later in Figure 4.1. To be compatible with YOLOv8, the images are reshaped to 640×640 pixels. Two different



Figure 4.2: Before augmentation



Figure 4.3: After augmentation

augmentations are also applied to increase the generalizability of the prediction models. Initially, the images are flipped horizontally. And to give the models the ability to predict from CCTV footage, a noise of 0.3% is added. Both augmentation techniques are employed in accordance with the recommendations made by [48]. In Figure 4.2 and Figure 4.3, a sample before augmentation and another one after augmentation are presented. Only the training images are augmented and increased from 6498 to 19494 instances.

After annotation, each image has two files saved in two directories - one is an image and another is the label file that contains bounding box coordinates. We use the YOLOv8-oriented bounding box dataset. In this dataset, the label file has 9 values instead of 5 which is common in rectangular bounding box coordinates. The label of our images has quadrilateral coordinates. The first value is the class ID and the remaining values are coordinates of four corners like (x_1, y_1) , (x_2, y_2) , (x_3, y_3) , and (x_4, y_4) .

4.3 TeacherActivityNet

YOLOv8 is considered state-of-the-art for object detection in real-time. This version of YOLOv8 has more speed and accuracy than the previous versions. A few key components of YOLOv8 are anchor-free detection, multi-scale predictions, and decoupled head architecture. We adopt the YOLOv8n as the base model for our detection task. In YOLOv8n there are three main parts in the architecture - Backbone, Neck, and Head.

In our proposed model, we modify the backbone of the YOLOv8n architecture. In the backbone, there are five feature pyramids (P) and four stages with each having a c2f (Cross-Stage Partial Network with 2 Convolutions and Fusion) module and a convolution module. We modify the last layer of the convolution module with a residual connection. The input of the convolution module of the last layer adds to the output of the same layer. This process is done for every convolution module. The modified convolution module is shown in Figure 4.4. In Figure 4.4, the internal mechanism of the modified

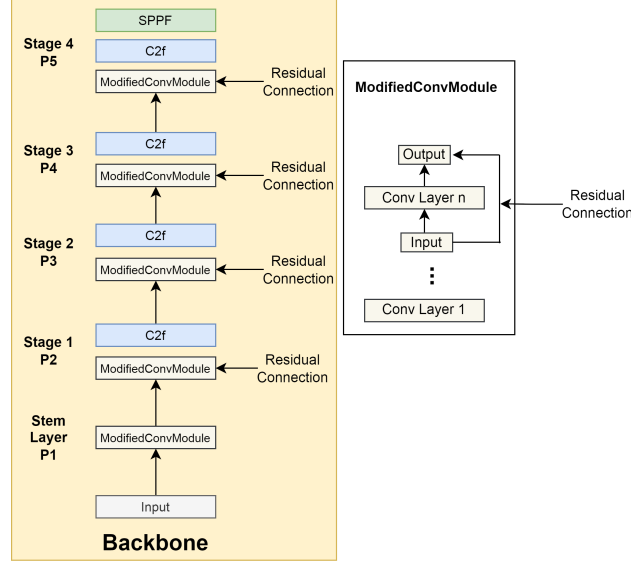


Figure 4.4: Modified backbone of our proposed TeacherActivityNet

convolution module with residual connection is presented. It adds a residual connection when the input and output channels of the convolution layer are the same with a stride of (1,1). This can help mitigate the vanishing gradient problem and potentially improve the learning of deeper features. Residual connections offer a direct pathway for gradients to propagate backward, helping to alleviate the vanishing gradient problem. Mathematically, considering the gradient of the loss L with respect to the input x , we have:

$$\frac{\partial L}{\partial x} = \frac{\partial L}{\partial y} \cdot \left(\frac{\partial F(x)}{\partial x} + 1 \right) \quad (4.1)$$

The addition of the $+1$ term ensures that the gradients can flow back directly, even when $\frac{\partial F(x)}{\partial x}$ becomes very small. The added residual connections might introduce a small computational overhead, but it is likely to be minimal given that they are only added under specific conditions. For benchmark analysis, in addition to creating detection models with our proposed TeacherActivityNet, we use pre-trained and fine-tuned YOLOv8 and FasterRCNN Resnet50 FPN model. All the models are trained on annotated images and labels with bounding box coordinates.

4.3.1 Reason Behind the Residual Connection in the Last Layer

The best approach to use in determining activity by the use of a residual connection is to employ it to the final convolutional layer. In backpropagation, this layer is last to reach the loss function, so the addition of a residual connection permits preserve that gradients can be easily propagated back to avoid the problem of vanishing gradients. There is also the last column which captures the high and abstract features critical in the detection of complex activities. Including a residual connection at this point, allows the network to improve these features instead of learning new representations, thus, positively affects

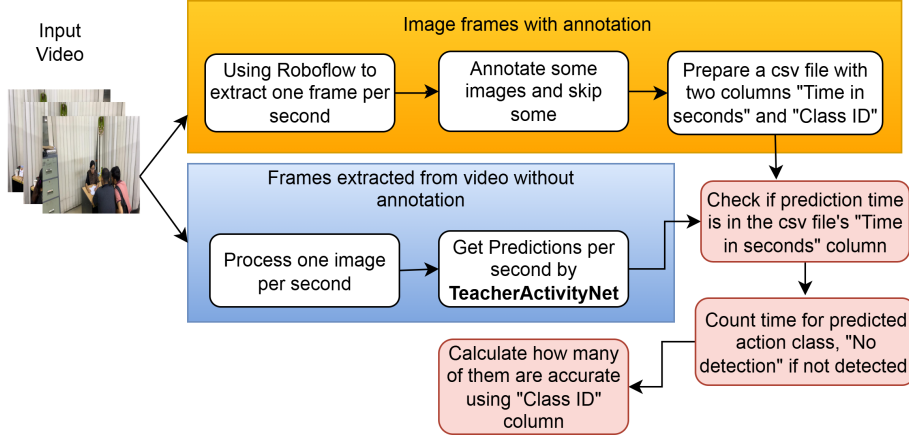


Figure 4.5: Flow chart for prediction on sample test dataset video with time count for each action class

training stability and convergence. Using the residual connections on the earlier layer or on a multi-layer makes it even more complex and more computationally expensive, as well as poses the threat of disrupting the pre-trained features that have been already learnt by the lower layers. Multiple residuals or long skip-connections also can challenge the hierarchical feature extraction in YOLO, which can degrade detection accuracy. Consequently, centralizing the residual connection on the final layer offers the best trade-off between gradient descent, feature distillation and the least interference with the pre-trained network structure.

4.4 Spatio-Temporal Feature

Initially, the performance of the model for action classes such as Arriving, Leaving, and Eating was very poor. We tried to add Spatio-Temporal features to the model during the training phase to improve the mAP50 for these action classes. This Spatio-Temporal feature focuses on both space and time, allowing us to focus on multiple frames to identify a pattern for any action class.

For the Eating action class, we used the pose-based YOLOv8 model to detect the wrist and head positions with movement while focusing on multiple image frames. The YOLO model does not predict the position of the mouth or hand, but it can detect the wrist and head. By using Spatio-Temporal features, we focused on more than one frame and checked that the wrist and head distance changed over time. This helped us determine whether the user was actually consuming food or not. Although it didn't improve much in predicting the Eating action class, the Spatio-Temporal feature, where we focused on multiple frames to detect the action class, helped increase the mAP50 for the Arriving and Leaving action classes.

In Figure 4.6, we can see that YOLOv8 can predict wrist and head positions in image frames using pose-based YOLOv8 models. At the same time, we tried to detect the food

item in the same image frame but failed in this instance. To predict the Eating action class, we tried many variations of the wrist-to-head distance and different numbers of frames to detect any movement or any increase in the distance between the head and wrist, indicating food being moved from a place to near the head.

In the training dataset, there are some videos where participants are actually drinking juice or water. In these videos, there is some movement of the wrist's central position, but the head remained in the same position. Even after focusing on multiple frames, the performance of the model on the Eating action does not improve much. In the same figure, using the YOLOv8 base model, we tried to detect the food item. In Figure 4.6a, the model cannot detect the food item. So, in many frames, the movement of the wrist and head is detected, but the base YOLO model did not detect the food. This is one of the main reasons for the poor performance of the Eating action class.

4.4.1 Action Class Wise Advantages

For the Eating action class, although the mAP50 increased slightly, for the action classes Arriving and Leaving, we gained a significant advantage. There were some image frames where the model was really confused between these two action classes, as shown in Figure 3.2.

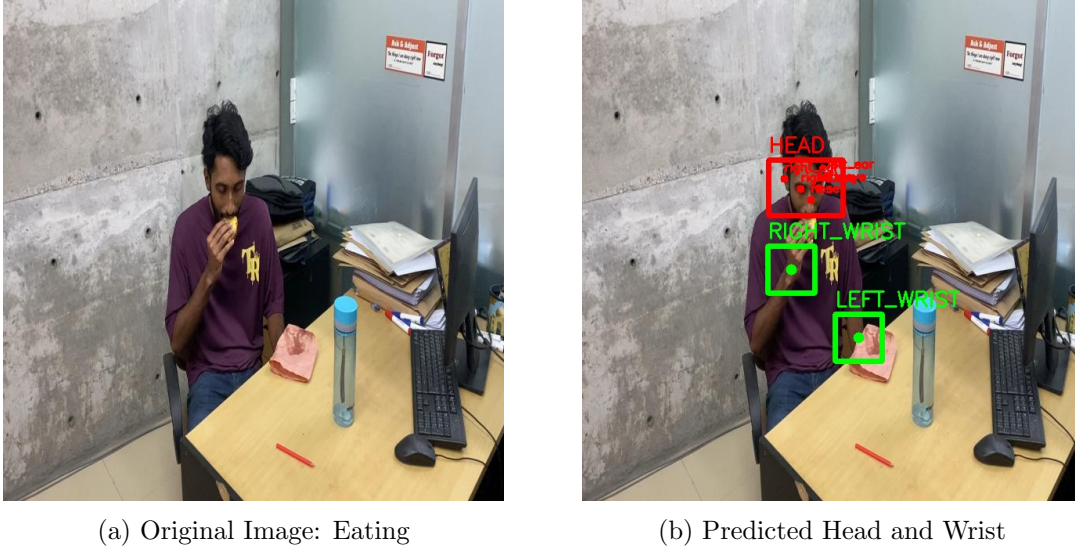


Figure 4.6: Comparison of Original Image and Predicted Head & Wrist Positions

4.5 Prediction from Videos

A big challenge for any computer vision model is measuring performance in the real world. In addition to finding the validation and test performance of our trained TeacherActivityNet model, we also measure its detection performance from videos. First we, detect the face of the participant in the video. So, we can count the time for each participant.

The face recognition part involves a deep neural network architecture in dlib, which is a 29-layer convolutional ResNet. It takes $150 \times 150 \times 3$ pixel images of faces and produces 128-dimensional feature vectors, providing a learned embedding space such that images of the same person are clustered together, and far apart, when representing different identities. In Figure 4.5, the steps to The face recognition part involves a deep neural network architecture in dlib, which is a 29 layer convolutional ResNet. It takes $150 \times 150 \times 3$ pixel images of faces and produces 128-dimensional feature vectors, providing a learned embedding space such that images of the same person are clustered together, and far apart, when representing different identities. We first take the input video, then extract one frame per second from the video. After extracting the frames, we annotate some images and skip others. The reason for skipping some images is to avoid those where a participant is not performing any of the actions mentioned in Table 3.1. In such cases, our trained model will not be able to predict any action classes for those frames. This typically happens when a participant is switching between action classes. For example, if a participant is performing Talking and then switches to Eating, they might first drop the phone on the desk, put it in their bag, or place it somewhere, and then begin to focus on eating. During these transition frames, they are not performing any of the specified action classes. Therefore, we skip these frames.

Once we have annotated all the image frames we want to annotate, we prepare a CSV file with two columns: "Time in seconds" and "Class ID." This serves as our ground truth for prediction from the proposed TeacherActivityNet model.

After annotation and having the ready CSV file, we can input the original/real test video into the trained model. TeacherActivityNet will first process one image per second to match our annotation process. It will provide a prediction for every second, which we will then match with the "Time in seconds" column in the CSV file. We will also compare the "Class ID" column with the predicted class. Note that TeacherActivityNet can predict any of the nine action classes, but it may also predict background, which we denote as no prediction made by the model. Then, we can count how many seconds the model predicted accurately. In this way, we can show how many frames we annotated and how many were correctly predicted.

Chapter 5

Experimental Setup

In this section, we discuss our experimental results. We use a machine with Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz processor with NVIDIA TITAN xp 12GB GPU to train our TeacherActivityNet models, as well other YOLO based models along with faster RCNN based models.

5.1 Test and Validation Results with Confusion Matrix

As we use TeacherActivityNet as a prediction model for our study, it has two different sets, one is used for training and the other for validation. This method takes images labeled with a bounding box. Our proposed model achieved 0.929 as the mean Average Precision(mAP50) on the validation dataset and 0.841 on the test dataset.

5.1.1 Hyper Parameter Settings

YOLOv8 model has been trained with hand-selected hyperparameters and augmentation techniques matching their specific needs at office activity tracking tasks. The training did not allow a certain number of epochs but used early stopping with a level of patience of 5, so that when the performance of the model did not improve further, it was suitable to stop automatically. The input image files were resized to 640x640 pixels and a batch size of 16 was selected that allows the separating training as well as optimization of GPU memory usage.

The large amount of data augmentation was used to increase the robustness of the model and generalization. Geometric augmentations encompassed mosaic augmentation, mixup, constrained rotation, translation, shearing and slight perspectival alteration in order to reflect upon the natural range of the office human activities and camera angles. The method of Photometric adjustments was done through the regulated changes made in all the aspects of hue, saturation and brightness and they were used to achieve varied lighting conditions. Copy-paste augmentation was only recourse to diversify objects but disable vertical-flipping operation to maintain real human orientations in the office scenario.

The process of hyperparameter tuning was iterative with default settings and was concentrated on the parameters of augmentation, configurations of models training, and on customizations according to the domain. This optimization comprised nine specific classes of actions, which had a difference in terms of temporal duration and space characteristics. Spontaneous movement based activities like arriving and leaving which are highly transitional in nature as compared to other classes like sleeping and working needed stronger geometric augmentations in order to extract dynamics of movement whereas classes which have a lot of posture-based such as sleeping and working meant that there was the need to be highly conservative in transformations so that the spatial critical details were not lost. Object-interaction movements required triviality of color and shape deformities, so as to preserve significant hand object interactions.

In general, the chosen hyper-parameters achieved a balance between the temporal features, pose sensitivity, the preservation of involvement of objects, and contexts. This method allowed efficient and successful learning to support various activities in the office with an effective early stopping approach avoiding over-fitting emergency.

5.1.2 Evaluation Metrics: mAP@50 and mAP@50-95

The evaluated results are reported using mAP@50 and mAP@50-95 metrics.

The quantitative measure of the YOLO model performance was the mean Average Precision (mAP), which is one of the most commonly used metrics in object detection tasks. It describes the trade-off between recall and precision by calculating the average precision (AP) over a range of confidence thresholds.

The Average Precision for each action class c , denoted as AP_c , is defined as the area under the precision-recall curve:

$$AP_c = \int_0^1 P_c(r) dr$$

where $P_c(r)$ represents the precision at recall level r for class c .

5.1.2.1 mAP@50

The mAP@50 metric calculates the mean of the Average Precision values across all classes at a fixed Intersection over Union (IoU) threshold of 0.5. It is computed as:

$$mAP@50 = \frac{1}{N} \sum_{c=1}^N AP_c^{IoU=0.5}$$

where N is the total number of classes.

5.1.2.2 mAP@50-95

The mAP@50-95 metric provides a more comprehensive evaluation by averaging the Average Precision over multiple IoU thresholds ranging from 0.5 to 0.95 in steps of 0.05:

$$mAP@50-95 = \frac{1}{N} \sum_{c=1}^N \left(\frac{1}{T} \sum_{t=1}^T AP_c^{IoU=i_t} \right)$$

where $T = 10$ represents the number of IoU thresholds $i_t \in \{0.5, 0.55, 0.6, \dots, 0.95\}$.

This evaluation strategy ensures that the model’s detection performance is assessed both at the common IoU threshold of 0.5 and across a range of stricter thresholds, providing a comprehensive understanding of localization accuracy and robustness across different action classes.

Table 5.1: Box precision, recall, mAP50 and mAP50-95 results for test dataset using TeacherActivityNet

Class	Images	Instances	P	R	mAP50	mAP50-95
all	808	808	0.738	0.827	0.841	0.491
Arriving	25	25	0.454	0.800	0.819	0.558
Counselling	183	183	0.952	0.965	0.990	0.593
Eating	59	59	0.610	0.371	0.413	0.270
Idle	64	64	0.663	0.688	0.700	0.406
Leaving	30	30	0.478	0.933	0.871	0.534
Sleeping	106	106	0.908	0.953	0.980	0.572
Talking	96	96	0.881	0.849	0.873	0.454
Using_Phone	119	119	0.789	0.941	0.940	0.404
Working	126	126	0.908	0.943	0.981	0.630

Table 5.2: Box precision, recall, mAP50 and mAP50-95 results for validation dataset using TeacherActivityNet

Class	Images	Instances	P	R	mAP50	mAP50-95
all	1337	1337	0.902	0.894	0.929	0.610
Arriving	117	117	0.947	0.752	0.925	0.654
Counselling	181	181	0.960	0.927	0.954	0.338
Eating	159	159	0.977	1.000	0.995	0.663
Idle	130	130	0.645	0.992	0.868	0.620
Leaving	109	109	0.734	0.908	0.895	0.576
Sleeping	172	172	1.000	0.601	0.764	0.521
Talking	137	137	0.863	0.942	0.974	0.623
Using_Phone	154	154	0.993	0.921	0.992	0.745
Working	178	178	0.997	1.000	0.995	0.746

In Table 5.1, the achieved results using TeacherActivityNet for different action classes are shown separately for the test dataset. Box precision, recall, mAP50 and mAP50-95 value for each class is shown in this Table5.1. The data instances are considered as images with bounding box to generate these test results. From this Table 5.1, we can observe that the Eating action class has the lowest performance in mAP50 whereas the prediction of Counselling and Working is showing higher prediction performance. Though the dataset used contains fewer instances of Leaving and Arriving action classes, the prediction worked well for these action classes. The overall mAP50 for the dataset using TeacherActivityNet is around 0.841 on the test dataset considering all the action classes. In Table 5.2 the performances for each action class is shown for validation dataset using

TeacherActivityNet model. In this table the Eating action has very good mAP50, but for the rest dataset the performance is very poor.

In Figure 5.1a, the confusion matrix using the TeacherActivityNet pre-trained model for the validation dataset is provided. Predictions for most of the action classes are very good on the validation dataset. The prediction of the action class Idle is average. The main reason behind average accuracy is due to some image frames of Idle action class where participants were looking down. It looks like their eyes are closed. So, the model predicts Sleeping. As we can see 59 image frames predicted as Sleeping were actually Idle.

In Figure 5.1b, prediction accuracy drops in the test dataset. Counselling class has very high accuracy. The Eating class has the lowest accuracy. The main reason behind this is actually the Using_Phone action class. In some of the image frames, we can see the participant holding the food like they are holding the mobile phone. That is why in 14 instances the model is predicting Using_Phone instead of Eating. To get a better view of the confusion matrix we have given the normalized version of the confusion matrix here in Figure 5.2. As our test dataset is imbalanced, it will help us better understand the prediction of the model. In Figure 5.3 we can see the predictions for each class by the Faster RCNN based pretrained model. The best performing action classes are Counselling and Talking. However, the action classes Arriving and Eating perform poorly with a low mAP50 of 0.3685 and 0.0382 respectively.

By looking at all the confusion matrix we can suggest that our TeacherActivityNet perform really well compared to FasterRCNN based pretrained models. Later, we will compare most of the YOLO models to find out if the proposed TeacherActivityNet model is really doing better than the rest of the models.

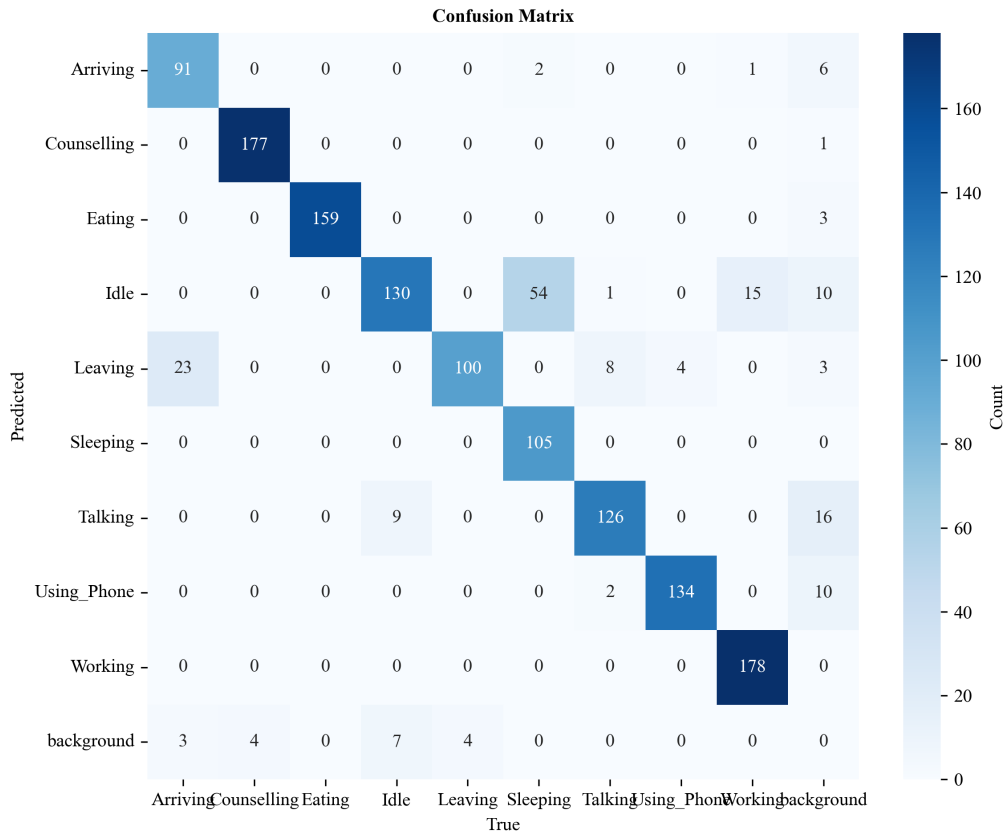
5.2 Spatio-Temporal Feature Tuning

While testing the dataset we have used Spatio-Temporal feature with focusing on the temporal tracking. We have found that due to dataset imbalance even with a action class specific windows set, mAP50 for any of the action classes were not improving.

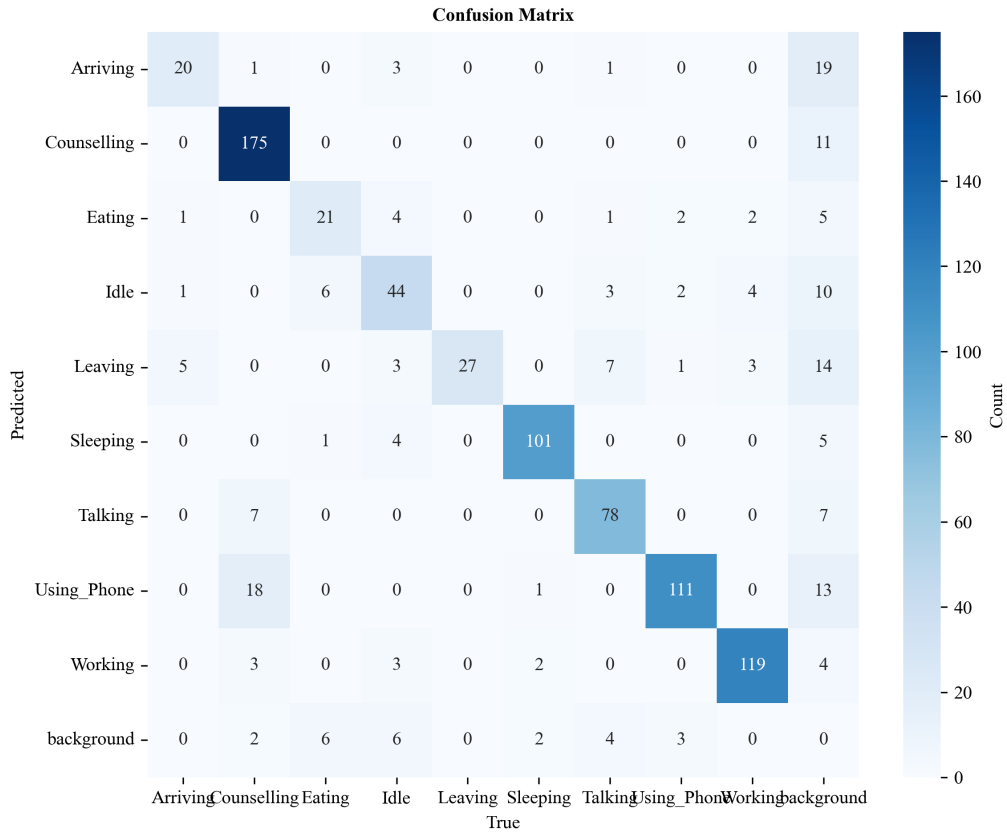
We propose an adaptive temporal smoothing consistency of actions recognition using our proposed TeacherActivityNet. Contrary to fixed windows, the system uses class-specific window size according to action duration: 3 frames of an action with a small duration (Arriving, Leaving), 510 frames of actions with moderate shift duration (Eating, Talking, Idle, Using Phone, and Working) and 1520 frames of stable actions at the same duration (Counselling, Sleeping).

The four implemented and compared smoothing strategies are (i) adaptive majority voting, (ii) confidence-weighted voting, (iii) weighted averaging with recency bias, and (iv) an ensemble of the foregoing methods. The hybrid ensemble was found to be best adapted to responding to short actions and stable to long actions.

In contrast to motion-vector based algorithms, object tracking and class-specific time-buffers using IoU are used on our approach to capture the prediction consistency over

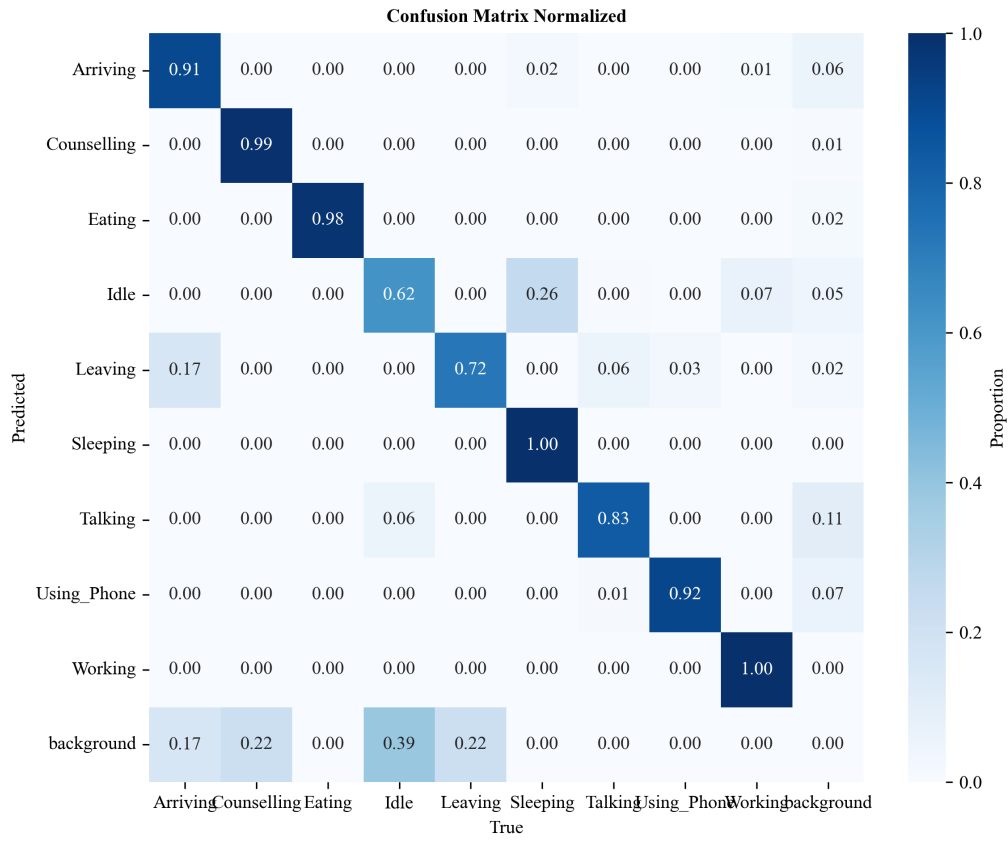


(a) Confusion matrix for validation dataset

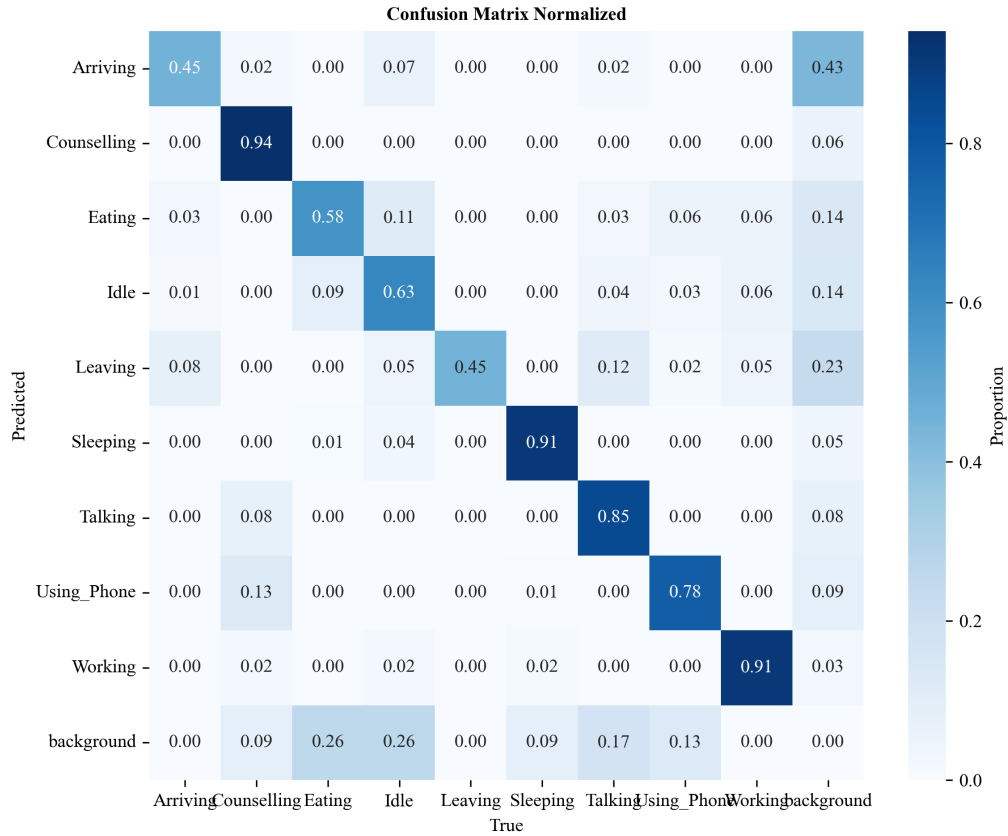


(b) Confusion matrix for test dataset

Figure 5.1: Confusion matrix for validation and test datasets of TeacherActivityNet



(a) Normalized confusion matrix for validation dataset



(b) Normalized confusion matrix for test dataset

Figure 5.2: Normalized confusion matrix of TeacherActivityNet

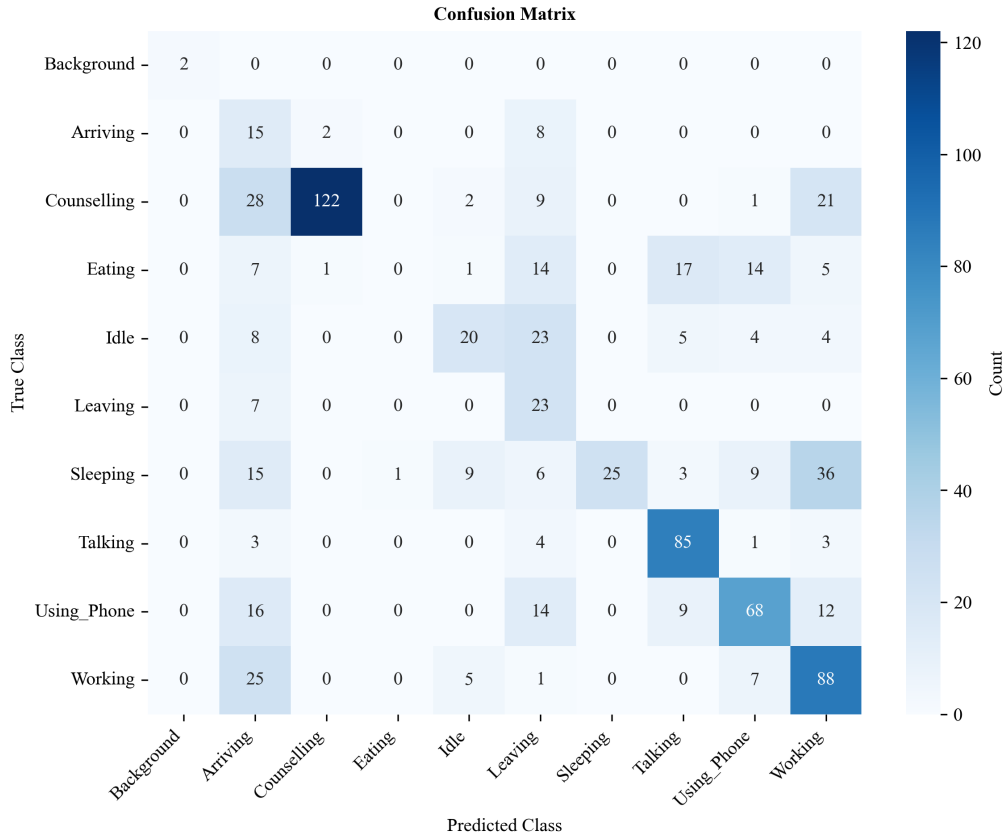


Figure 5.3: Confusion matrix for Faster R-CNN model for test dataset

time without optical flow. The results demonstrate that adaptive smoothing can enhance mAP@50, in particular, the confusion between visually similar classes (e.g., Idle vs. Sleeping), at the same time preserving real-time performance.

5.2.1 Reason Behind no Improvement in Adaptive Analysis

The main reason behind the poor improvement in the adaptive window size analysis is that, in our test dataset, the class-wise image instances are not balanced. This is because the actions are different and cannot be performed by the participants for the same period of time. For example, the time for the Arriving and Leaving action classes is shorter than for the other action classes. This causes an imbalance in the test dataset. Additionally, multiple participants performed the actions in different ways, taking varying amounts of time. Therefore, we cannot set a single window size for all the classes. Even with different, class-specific window sizes, our model still fails to improve prediction rates.

In Figure 5.4a we can see the window size is set according to action class. And in Figure 5.4b we can see no action class is getting an improvement.

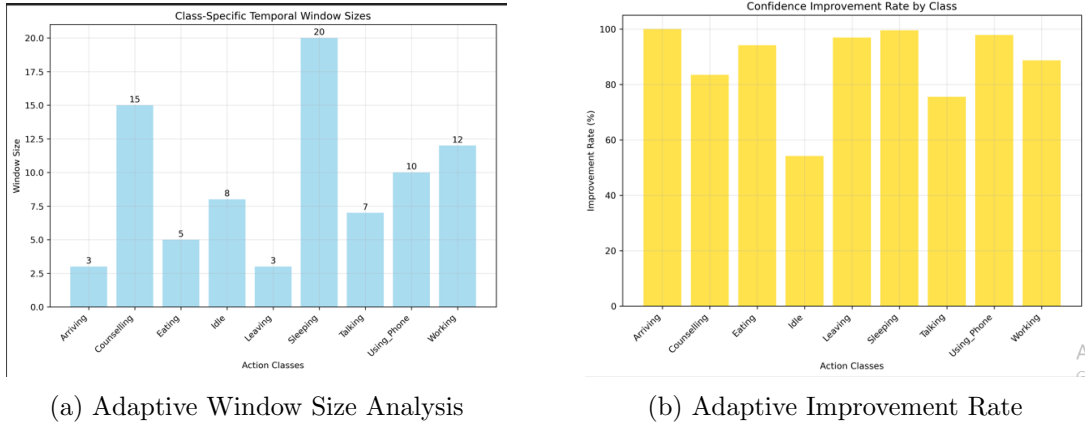


Figure 5.4: Comparison of Adaptive Window Size and Improvement Rate

5.3 Confidence Curves with Train and Validation Loss

We are going to analyze the performance of the confidence curves of our model Teacher-ActivityNet. And we will also see the train and validation dataset loss evaluation here.

Over the test dataset the confidence curve for F1 score, precision, precision-recall, and recall is shown in Figure 5.5. From Figure 5.5a, our results revealed the confidence threshold with the maximum F1 score among all classes is 0.279 with the F1 score value of 0.76. Figure 5.5b reflects the average precision value for all classes which is 1.00 at a confidence threshold of 0.914. Again, Figure 5.5c shows the precision-recall curve reflecting the mean Average Precision(mAP0.5) value in 0.841. Finally, Figure 5.5d shows a recall curve with an average recall value of 0.95. However, it is observed that for all other values except precision, the average performance of sleeping is much lower than the other action classes.

In Figure 5.6 train and validation results are shown for bounding box loss(box_loss), class prediction loss(cls_loss), distribution focal loss(df_l_loss), precision - recall for the training dataset and mAP for validation dataset. For the loss functions, it reflects a decrement of loss. It is also observed that the loss function for the training dataset is showing decreasing behavior, on the other hand, for the validation set, the result fluctuates.

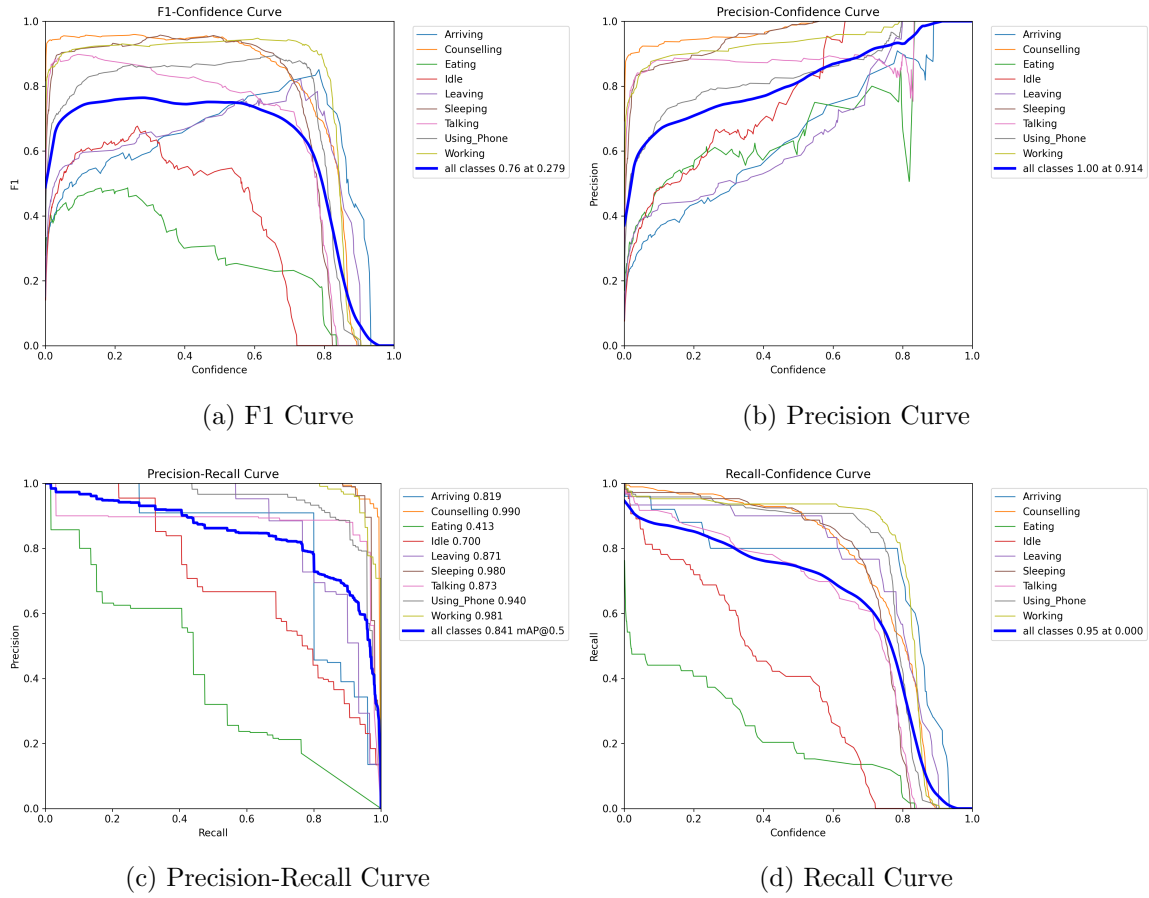


Figure 5.5: F1, Precision, and Recall Curves for All Activities

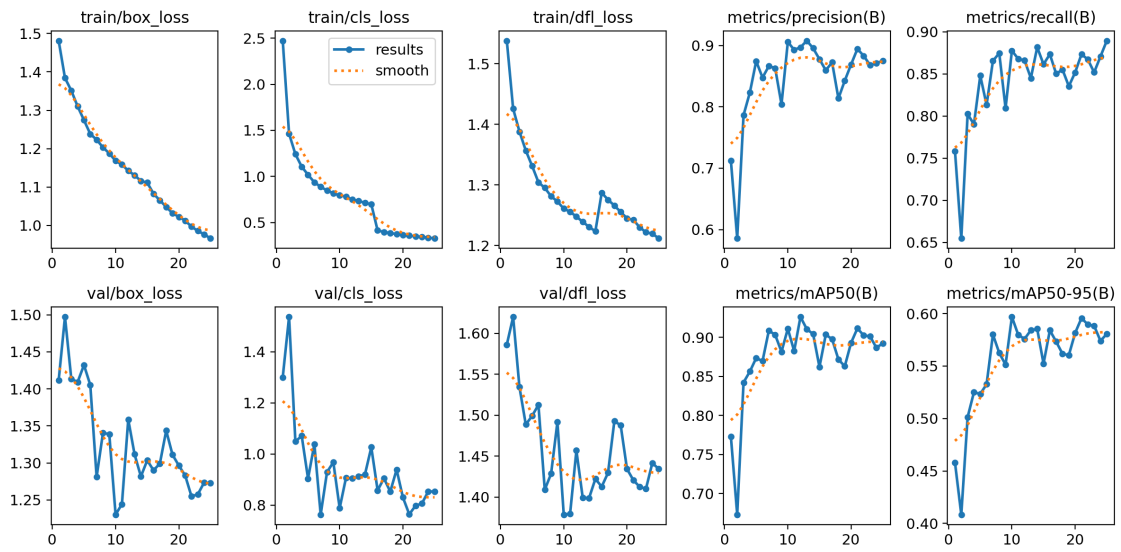


Figure 5.6: Train and Validation Loss with Metrics Evaluation

5.4 Model Predictions on Validation and Test Images

In this section we are going to compare the labeled images with prediction made by our proposed TeacherActivityNet model. In all the images we are going to see here, the batch size is 16. So, we are going to compare those 16 labeled images with their predictions by our model.

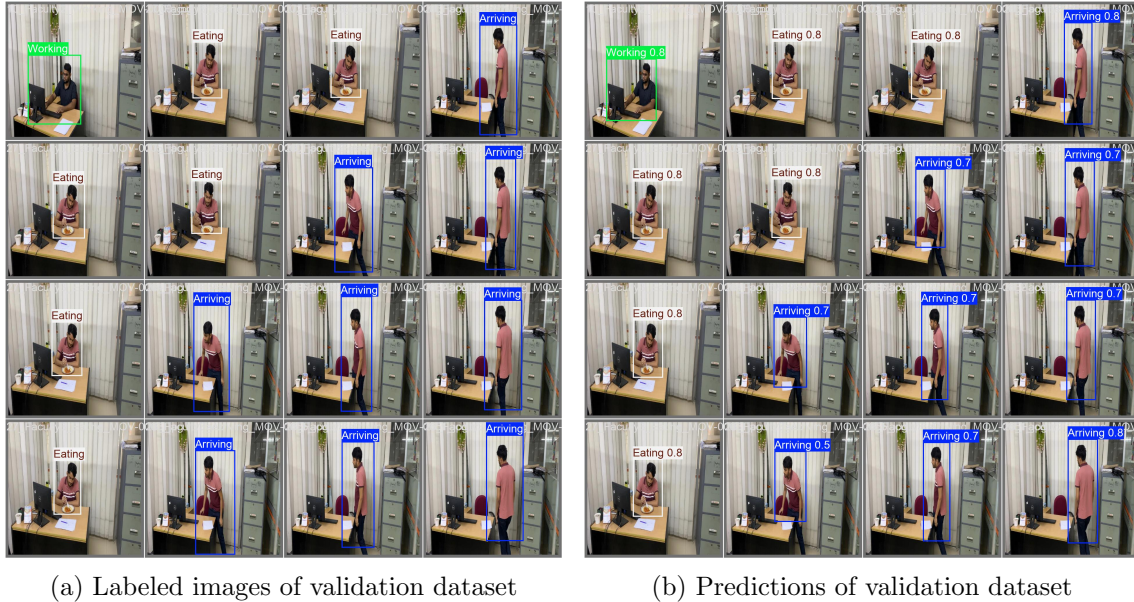


Figure 5.7: Comparison between ground truth labels and model predictions on validation dataset for batch-0

In Figure 5.7a, validation labels are shown. We can see the action class names along with every image. And there will some combined images due to mosaic built-in augmentation in YOLO. In Figure 5.7b, the predicted images of validation by the TeacherActivityNet models are shown. The bounding box and the class score are also provided for each activity. The class score reflects how reliable the prediction result TeacherActivityNet is for each class.

In Figure 5.8 we can see similar model performance. model predicts each action class perfectly. In Figure 5.9 all the Arriving actions predicted accurately by our model TeacherActivityNet in validation dataset. And the confidence score for each prediction is 0.9.

In Figure 5.10a we can see the labels of the test images and in Figure 5.10b we can see the predictions made by our proposed model TeacherActivityNet. All predictions here are correct, except for one image. We can find this image in the second row first column. We have extracted the same two pictures side by side in Figure 5.11. The image frame has action class Using_Phone, but the model predicts two action classes. One is correct, which is Using_Phone but another is an incorrect prediction for the action class Arriving. Although it looks like a correct prediction, we can see the back of someone who is entering the room maybe. But in reality there was a transparent glass from which we can see the next room, where a person was just sitting in a chair. Our model predicted that and it

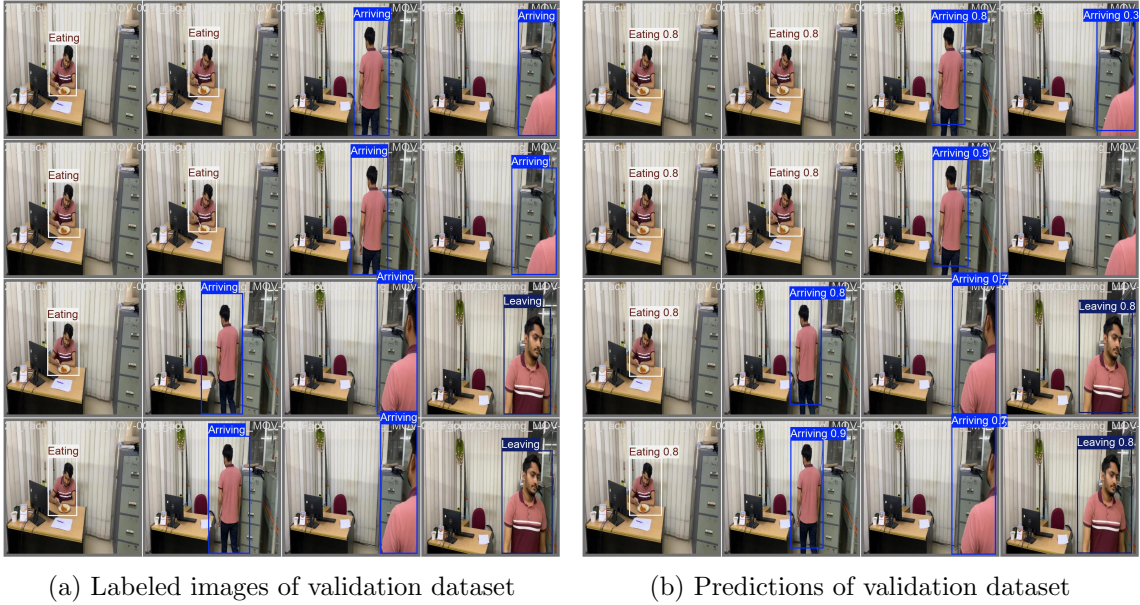


Figure 5.8: Comparison between ground truth labels and model predictions on validation dataset for batch-1

becomes confused, but at least even if the picture is blur our model can predict action class, that is the only positive sign of our model TeacherActivityNet.

In Figure 5.13a we can see the second set of labeled images and in Figure 5.13b we can see predictions of the model. Again, the model is accurate in every image frame, but the problem remains the same here. The third person participates in the predictions on multiple image frames from the last column and the last two rows. The last row of the two images was merged and is shown in Figure 5.12. The action classes are perfectly predicted, but the model is counting his part here. That person is not even in the same room.

In Figure 5.14a we can see the test dataset another batch label is given and in Figure 5.14b the prediction has been made. In these prediction the problem remains same because of the person sitting in the next room. The Counselling getting predicted multiple time due to that person and also some time the model is predicting Arriving.

In summary, we can say that our proposed TeacherActivityNet model is performing very well in predicting each instance. For some of the action classes, the prediction score is really high, like 0.9. So, the model is very confident about those image instances. Even though YOLO-based models are mainly object detection models, not action recognition models, our model is predicting the actions perfectly in most of the image instances. There are still some concerns regarding the action class Eating, but we are confident that, like the other action classes, we can improve the prediction for those instances in the future as well.

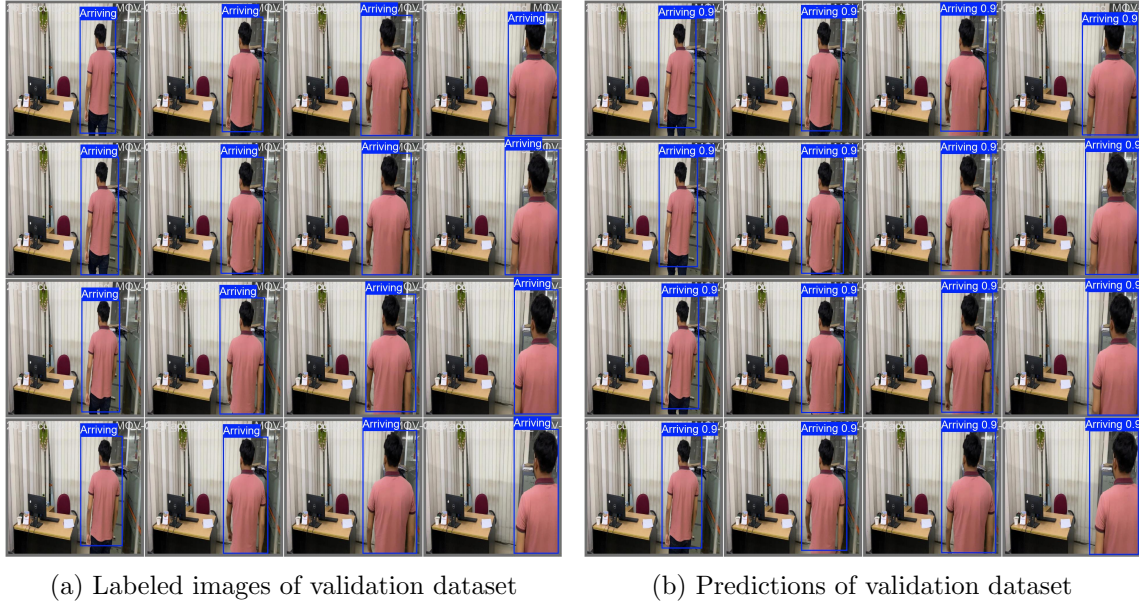


Figure 5.9: Comparison between ground truth labels and model predictions on validation dataset for batch-2

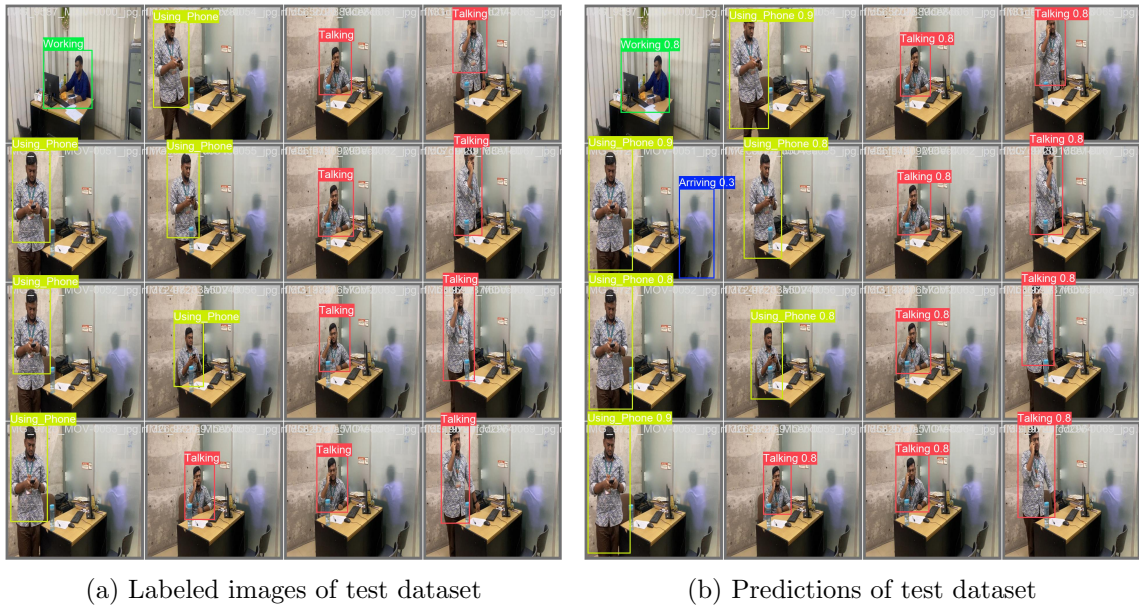


Figure 5.10: Comparison between ground truth labels and model predictions on test dataset for batch-0

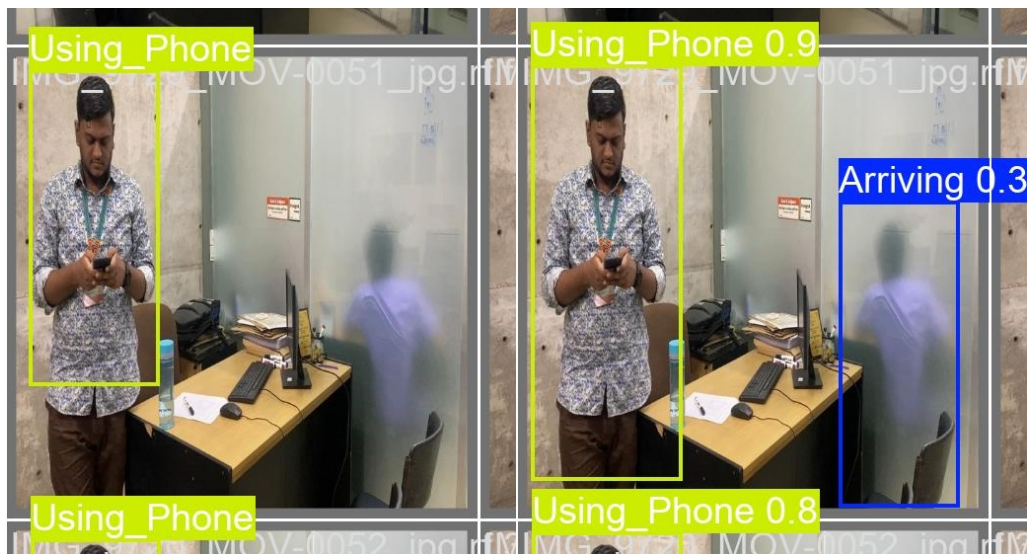


Figure 5.11: Merged view of ground truth labels and model predictions side by side for test dataset for batch-0

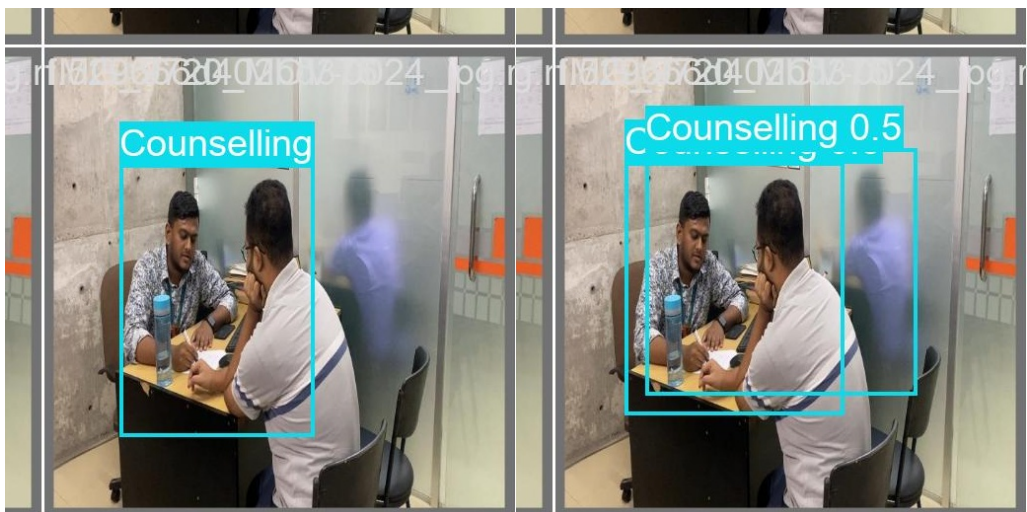


Figure 5.12: Merged view of ground truth labels and model predictions side by side for test dataset for batch-1

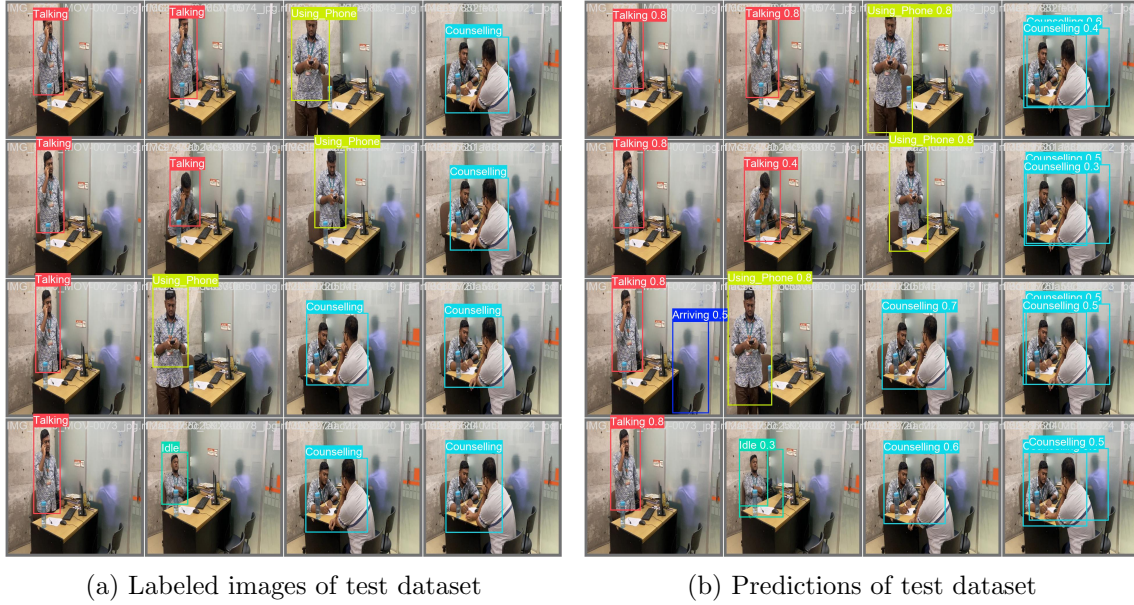


Figure 5.13: Comparison between ground truth labels and model predictions on test dataset for batch-1

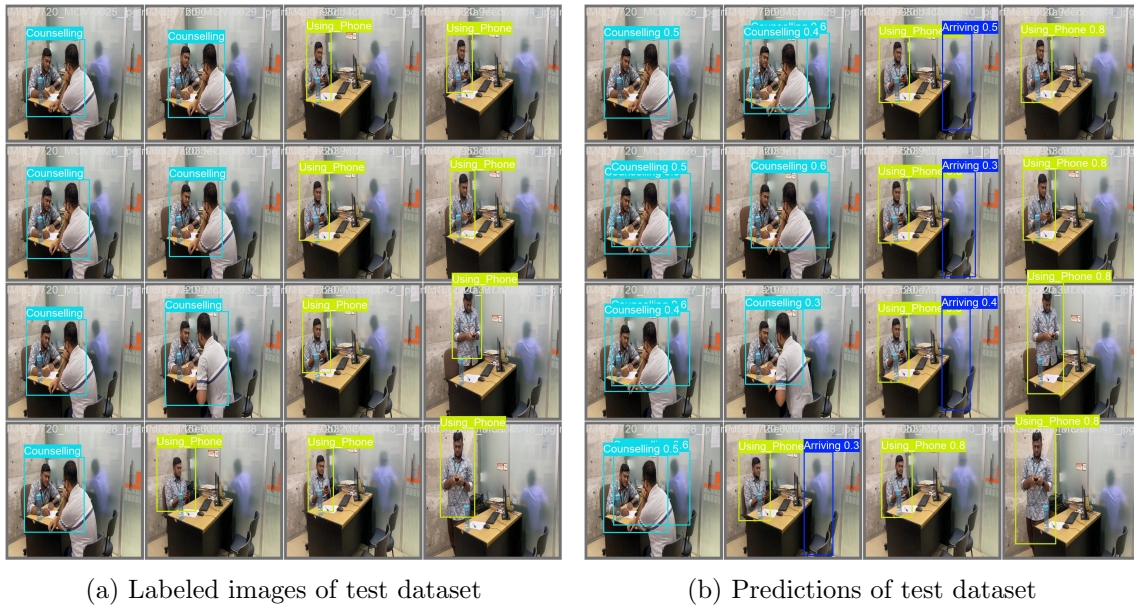


Figure 5.14: Comparison between ground truth labels and model predictions on test dataset for batch-2

5.5 Faster RCNN Model Performance

We will find out the Faster RCNN based model performance on test dataset. We can check out the model prediction over each action class.

Table 5.3: The mAP50 for test dataset using Faster RCNN ResNet50 FPN

Action Class	Total	Correct	mAP50
Arriving	25	15	0.3685
Counselling	183	122	0.9644
Eating	59	0	0.0382
Idle	64	20	0.4518
Leaving	30	23	0.4572
Sleeping	104	25	0.5303
Talking	96	85	0.8073
Using_Phone	119	68	0.6465
Working	126	88	0.6256

In order to evaluate some other method results using our dataset, we use the Faster RCNN ResNet50 FPN pre-trained model. In Table 5.3, the results achieved for the Faster RCNN ResNet50 FPN model are shown in terms of correct predicted instances and thus mAP50. From the table, we can find that the prediction of eating class is around 0.0382 for this model. However, the prediction rate for the Idle class is significantly low for this model in our dataset. Counselling class can be considered as having the highest performance in terms of mAP50 as 0.9644. The overall achievement rate for Faster RCNN ResNet50 is 0.543 which is lower than the overall rate of TeacherActivityNet.

5.6 Model Performance on Videos

All the results discussed above are generated using image instances with a specific bounding box. In order to consider the real-time scenario, we take a video with no boundaries to evaluate the performance of the model for all 9 action classes. Therefore, we took a video of 127 seconds from the test dataset to perform the task. In the first phase of the task, we perform face recognition to correctly predict the person. We have taken a picture of the participant who is present in the video to perform the recognition task. Using Python's built-in face_recognition library, which employs the dlib ResNet-29 model, we detected the person. We can easily detect the first and last appearances of the person in a video. After the face recognition process is performed successfully, we use our proposed model TeacherActivityNet to predict the action class for each task. To compare the predicted results and generate the accurately predicted tasks, the video was divided into image frames and annotated using Roboflow. After that, the predicted and annotated image frames are compared to find the correctly predicted instances. In Table 5.4, the annotated

frame numbers and the predicted frame numbers are given for each action class. The total predicted time for each action class is also provided to make a decision on the overall task prediction for each person. The No Detection class indicates the number of times the model predicted that the person was not involved in any of the 9 available action classes. No Detection can be correct or incorrect. This action class in real-time videos is very much possible due to the frequent switching of actions by the participant. The time when the participant is switching our proposed model TeacherActivityNet will not be able to predict any of the action classes.

Table 5.4: Predicted frame counts for real time video from test dataset using the model

Action Class	Time in Seconds	Annotated Frames	Accurately Predicted Frames
Arriving	3	3	1
Counselling	38	29	29
Eating	0	7	0
Idle	26	4	4
Leaving	8	3	3
Sleeping	12	12	11
Talking	11	8	5
Using Phone	14	10	10
Working	2	12	2
No Detection	15	N/A	N/A

5.7 Ablation Studies

In Table 5.5, a comparison of different convolutional models in YOLOv8 with TeacherActivityNet is shown. We will try to find out if any convolutional module performs better than the TeacherActivityNet.

Depthwise Separable Convolutional Module(DSC of YOLOv8): This method uses the DWConv convolutional module and has mAP50 lower than the other two modules.

Simple Grouped Shuffle Convolutional Module(Simple GS of YOLOv8): This is a combination of Standard Convolutional Module and Depthwise Separable Convolution module for group sampling. We can clearly see that it has the best inference speed of 1.82 ms/image compared to the other two methods.

Standard Convolutional Model(SC Module of YOLOv8): This typically refers to a deep learning model that uses standard convolutional layers for feature extraction. From Table 5.5, it is observed that this model has outperformed the other two applied methods(DSC and Simple GS). It has mAP50 on the test dataset of 0.720.

TeacherActivityNet: From Table 5.5, we can say that our proposed model TeacherActivityNet demonstrates superior performance than other convolutional modules with mAP50 of 0.841 in the test dataset. However, the inference speed of the Simple GS convolutional module is marginally better than that of TeacherActivityNet. Although the TeacherActivityNet inference speed is slower, compared to other convolutional modules,

we can trade a little speed for more accuracy.

Table 5.5: Comparative baseline analysis with our TeacherActivityNet model

Metric	DSC (YOLOv8n)	SimpleGS (YOLOv8n)	SC (YOLOv8n)	TeacherActivityNet
mAP50	0.473	0.503	0.720	0.841
Inference Speed	2.16 ms/image	1.82 ms/image	9.34 ms/image	1.9 ms/image

5.8 YOLO Models vs TeacherActivityNet

In this section, we are going to compare some of the YOLO-based models, such as YOLOv5, YOLOv8, YOLOv9, YOLOv10, YOLOv11, and YOLOv12, with TeacherActivityNet. We will compare their results based on mAP50 on the validation and test datasets. In Table 5.12, we have compared the performance of most of the YOLO-based models with TeacherActivityNet on the test and validation datasets. In the test dataset, TeacherActivityNet has outperformed all other models with an mAP50 of 0.841. The second-best model is YOLOv11 with an mAP50 of 0.743.

First, we show the performance of each action class in the test dataset across different YOLO models.

In Table 5.6, we can see the performance of the action classes using YOLOv5. In Table 5.7, the performance is slightly better than YOLOv5. We have used the YOLOv8 model here. In Table 5.8, we obtained a better result using YOLOv9. Although the Eating action class mAP50 is very poor, the overall performance is good. In Table 5.9, we got almost the same result as YOLOv9 using the YOLOv10 model. In Table 5.10, the performance is very good. For the action classes Arriving and Leaving, there are significant gains. In Table 5.11, using YOLOv12, the performance slightly decreased for many action classes.

Table 5.6: Box precision, recall, mAP50 and mAP50-95 results for test dataset using YOLOv5 for each action class

Action	Images	Instances	Box (Precision)	Recall	mAP50	mAP50-95
all	808	808	0.680	0.706	0.697	0.410
Arriving	25	25	0.384	0.880	0.628	0.397
Counselling	183	183	0.959	0.951	0.984	0.622
Eating	59	59	0.355	0.299	0.331	0.210
Idle	64	64	0.654	0.503	0.537	0.334
Leaving	30	30	0.423	0.633	0.478	0.304
Sleeping	106	106	0.695	0.962	0.810	0.479
Talking	96	96	0.862	0.585	0.772	0.384
Using Phone	119	119	0.800	0.773	0.814	0.346
Working	126	126	0.990	0.767	0.920	0.615

In summary, we can say that our proposed TeacherActivityNet model has outperformed the rest of the YOLO models. It also has a better inference speed as well.

Table 5.7: Box precision, recall, mAP50 and mAP50-95 results for test dataset using YOLOv8 for each action class

Action	Images	Instances	Box (Precision)	Recall	mAP50	mAP50-95
all	808	808	0.733	0.722	0.720	0.435
Arriving	25	25	0.592	0.720	0.620	0.397
Counselling	183	183	1.000	0.988	0.995	0.645
Eating	59	59	0.399	0.475	0.425	0.289
Idle	64	64	0.512	0.542	0.484	0.275
Leaving	30	30	0.629	0.833	0.690	0.408
Sleeping	106	106	0.915	0.809	0.904	0.549
Talking	96	96	0.808	0.657	0.670	0.341
Using_Phone	119	119	0.769	0.555	0.719	0.365
Working	126	126	0.975	0.919	0.973	0.645

Table 5.8: Box precision, recall, mAP50 and mAP50-95 results for test dataset using YOLOv9 for each action class

Class	Images	Instances	P	R	mAP50	mAP50-95
all	808	808	0.691	0.711	0.733	0.424
Arriving	25	25	0.672	0.840	0.813	0.529
Counselling	183	183	0.986	1.000	0.995	0.601
Eating	59	59	0.253	0.339	0.237	0.124
Idle	64	64	0.685	0.562	0.525	0.265
Leaving	30	30	0.277	0.867	0.676	0.451
Sleeping	106	106	0.876	0.642	0.857	0.482
Talking	96	96	0.772	0.563	0.675	0.350
Using_Phone	119	119	0.713	0.856	0.877	0.370
Working	126	126	0.989	0.733	0.940	0.643

Table 5.9: Box precision, recall, mAP50 and mAP50-95 results for test dataset using YOLOv10 for each action class

Action	Images	Instances	Box (Precision)	Recall	mAP50	mAP50-95
all	808	808	0.728	0.686	0.736	0.437
Arriving	25	25	0.453	0.840	0.729	0.484
Counselling	183	183	0.989	0.955	0.992	0.624
Eating	59	59	0.736	0.189	0.415	0.298
Idle	64	64	0.522	0.500	0.517	0.304
Leaving	30	30	0.422	0.833	0.622	0.385
Sleeping	106	106	0.845	0.774	0.886	0.519
Talking	96	96	0.804	0.438	0.655	0.338
Using_Phone	119	119	0.825	0.849	0.841	0.355
Working	126	126	0.952	0.795	0.971	0.623

Table 5.10: Box precision, recall, mAP50 and mAP50-95 results for test dataset using YOLOv11 for each action class

Action	Images	Instances	Box (Precision)	Recall	mAP50	mAP50-95
all	808	808	0.688	0.729	0.743	0.443
Arriving	25	25	0.609	0.880	0.833	0.564
Counselling	183	183	0.984	0.990	0.994	0.645
Eating	59	59	0.320	0.390	0.340	0.231
Idle	64	64	0.510	0.537	0.488	0.257
Leaving	30	30	0.308	0.833	0.759	0.489
Sleeping	106	106	0.817	0.674	0.816	0.467
Talking	96	96	0.898	0.490	0.689	0.319
Using_Phone	119	119	0.763	0.832	0.781	0.348
Working	126	126	0.983	0.933	0.990	0.664

Table 5.11: Box precision, recall, mAP50 and mAP50-95 results for test dataset using YOLOv12 for each action class

Action	Images	Instances	Box (Precision)	Recall	mAP50	mAP50-95
all	808	808	0.677	0.691	0.715	0.422
Arriving	25	25	0.596	0.709	0.705	0.455
Counselling	183	183	0.950	0.973	0.989	0.620
Eating	59	59	0.396	0.525	0.368	0.248
Idle	64	64	0.464	0.234	0.375	0.212
Leaving	30	30	0.466	0.867	0.715	0.486
Sleeping	106	106	0.680	0.896	0.841	0.483
Talking	96	96	0.766	0.648	0.747	0.357
Using_Phone	119	119	0.808	0.565	0.754	0.362
Working	126	126	0.971	0.804	0.940	0.576

Table 5.12: Summary of the performance comparison of YOLO-based models with TeacherActivityNet

Model	Test mAP50	Validation mAP50	Inference Speed (ms/image)
YOLOv5	0.697	0.923	3.8
YOLOv8	0.720	0.941	2.1
YOLOv9	0.733	0.932	5.7
YOLOv10	0.736	0.927	3.1
YOLOv11	0.743	0.922	2.0
YOLOv12	0.715	0.932	3.0
TeacherActivityNet	0.841	0.929	1.9

5.9 Activity Recognition Model Performance

Our main task is to recognize the activities of the teachers. To achieve this, we have explored several activity recognition models to gain insight into spatial and temporal-based approaches. We have analyzed the performance of three models: two of them, MC3-18 and R2plus1D, are based on the ResNet-18 architecture, while the third, the Temporal Segment Transformer (TST), is a custom 3D CNN model built on ResNet3D-18. The TST is enhanced with temporal pooling and a fully connected classifier, designed for efficient spatio-temporal feature learning and action recognition in video data.

Table 5.13: Comparing activity recognition models

Metric	MC3-18	R2plus1D	Temporal Segment Transformer
Validation Accuracy	48.15%	40.74%	70.42%
Test Accuracy	37.11%	32.85%	58.34%

In Table 5.13, we can see that the performance of the activity recognition models is not satisfactory, with the performance of the test dataset being notably poor. These models were trained directly on videos. Based on the results, we can conclude that the TeacherActivityNet and FasterRCNN-based pre-trained models outperform the activity recognition models.

5.9.1 Causes of Poor Activity Recognition Model Performance

The main reason behind the poor performance of the activity recognition models is that they focus on the entire image frame, rather than the specific region that actually represents the action. In YOLO-based models, we have image annotation with bounding box labels available. These YOLO-based models focus only on the specific regions of the image, not on the unnecessary parts of the image frame. Some may suggest that we should have cropped the image frames to show only the person. However, if we had done that, another problem would have emerged. Some of our defined actions depend on the importance of objects in the image frames, not just the person. For example, if someone is performing the Working action class and we crop the person from the image frame, we would no longer see the computer or laptop, which are necessary for the Working action class. According to the definition of the class, those objects are essential. Therefore, we cannot crop the videos or extracted frames of the videos. In Figure 5.15, we have shown the comparison between the original image and the cropped image. In Figure 5.15a, we can see that the monitor is visible, which is responsible for the Working action class. In Figure 5.15b, we can no longer see the monitor. These activity recognition models that focus on the entire frame lead to poor performance overall. We can conclude that TeacherActivityNet, or any other YOLO model, would definitely perform better in our task of faculty personalised

assistive system.



(a) Original image of Working class



(b) Cropped image of Working class

Figure 5.15: Comparison between original and cropped image of action class Working

Chapter 6

Conclusion

6.1 Summary

TeacherActivityNet model trained with a new dataset that enhances cognitive functions and life-work harmony through activity recognition in educational office settings by using recorded video data. TeacherActivityNet will represent a time-based but objective index of teacher activities in the office because it quantifies the time spent in nine semantically meaningful action categories (Arriving, Leaving, Counselling, Eating, Idle, Sleeping, Talking, Using.Phone and Working). This time split is able to provide teachers with timely and practical knowledge concerning what their day truly looks like. Greater self-knowledge (achievable with the specific measures of total counselling time, phone use, arrival, and leaving patterns) can help the teachers to manage the inevitably existing discrepancies between instruction, administrative, and non-work behaviors and make specific changes in everyday practice.

With regard to cognitive and well-being, there are three ways in which the TeacherActivityNet can be constructive. First, it decreases the mental demands of self-monitoring and planning due to changing a complex daily behavior into basic time metrics, making mental resources available to teachers to perform other high-priority tasks. Second, objective feedback contributes to better self-regulation: if the teacher observes that students spend too much time on their phones or too little communicating with others, she can adjust routines and restore the balance, which is linked not only to lower stress levels but also to higher levels of focusing. Third, longitudinal monitoring allows the identification of harmful patterns (e.g., frequent idle or sleep periods during the working time, habitual late arrivals or high overtime rates) early on, thus, allowing appropriate organizational or personal action to be undertaken to safeguard mental health and job satisfaction.

We want to stress the idea that these benefits are likely, but not certain. Strong empirical validation is required: we suggest testing TeacherActivityNet in longitudinal designs that should involve links between objective activity data, provided by the system, and psychological and occupational measures, which have received validation (e.g., well-being scales, stress or burnout inventories, and cognitive performance tests). The resultant

mixed-methods evaluation will show how changes in the measured behavior are associated with changes in cognitive indicators and well-being, and the effect sizes resulting in various institutional settings.

The proposed model TeacherActivityNet delivers a major enhancement in average precision with respect to the base model performance. Proper safeguards have been developed to protect both the personal information and data security of everyone who participates in the study. A benchmark analysis was done between TeacherActivityNet and various YOLO-based systems along with newly introduced YOLOv11 and YOLOv12 models and Faster R-CNN-based systems in addition to activity recognition frameworks. TeacherActivityNet led all competition models in assessment which suggests it can function as basis for creating customized educational assistive systems.

6.2 Limitation

One of the major limitations of our dataset is that the videos were captured using smartphones to simulate CCTV recordings. We have not considered action classes where participants can perform multiple actions together, such as eating while giving counseling to students. There could be many more mixed-actions like this. In addition, we have not considered multiple teachers in a single room. However, we can handle this situation with our current model by using the YOLO model's person detection to crop regions for each teacher.

6.3 Future Work

Utilizing actual CCTV footage in future work could potentially improve the precision of the model during fine-tuning. We can work on mixed-action classes and intend to work with multiple teachers. We will compare the performance of TeacherActivityNet with other similar datasets, if available, to justify its overall performance. For action classes with lower average precision, we can work on pose estimation as well. The foundation of our work allows us to create custom assistive systems for diverse office workspaces which seek to boost mental health and maximize work hours while boosting office output. We will try to figure out why the modifications made to the backbone of YOLOv8 do not work well with the rest of the YOLO models. Specifically, with YOLOv12, we will try some additional modifications to improve its performance.

References

- [1] Kristina Host and Marina Ivašić-Kos. An overview of human action recognition in sports based on computer vision. *Heliyon*, 8(6):e09633, 2022.
- [2] L Xiao, Y Cao, Y Gai, E Khezri, J Liu, and M Yang. Recognizing sports activities from video frames using deformable convolution and adaptive multiscale features. *j. cloud comput.* 12 (1), 167 (2023), 2023.
- [3] Hongwei Yin, Richard O Sinnott, and Glenn T Jayaputera. A survey of video-based human action recognition in team sports. *Artificial Intelligence Review*, 57(11):293, 2024.
- [4] Fei Wu, Qingzhong Wang, Jiang Bian, Ning Ding, Feixiang Lu, Jun Cheng, Dejing Dou, and Haoyi Xiong. A survey on video action recognition in sports: Datasets, methods and applications. *IEEE Transactions on Multimedia*, 25:7943–7966, 2022.
- [5] Ersan Zhou and Heqing Zhang. Human action recognition toward massive-scale sport sceneries based on deep multi-model feature fusion. *Signal Processing: Image Communication*, 84:115802, 2020.
- [6] Neziha Jaouedi, Nouredine Boujnah, and Med Salim Bouhlel. A new hybrid deep learning model for human action recognition. *Journal of King Saud University-Computer and Information Sciences*, 32(4):447–453, 2020.
- [7] Mukhiddin Toshpulatov, Wookey Lee, Suan Lee, Hoyoung Yoon, and U Kang. Ddc3n: Doppler-driven convolutional 3d network for human action recognition. *IEEE Access*, 2024.
- [8] Preksha Pareek and Ankit Thakkar. A survey on video-based human action recognition: recent updates, datasets, challenges, and applications. *Artificial Intelligence Review*, 54(3):2259–2322, 2021.
- [9] Jiatian Liu. Convolutional neural network-based human movement recognition algorithm in sports analysis. *Frontiers in psychology*, 12:663359, 2021.
- [10] Md. Atikuzzaman, Tarafder Razibur Rahman, Eashita Wazed, Md. Parvez Hossain, and Md. Zahidul Islam. Human activity recognition system from different poses with

- cn. In *2020 2nd International Conference on Sustainable Technologies for Industry 4.0 (STI)*, pages 1–5, 2020.
- [11] Yahui Zhang, Shaodi You, Sezer Karaoglu, and Theo Gevers. 3d human pose estimation and action recognition using fisheye cameras: A survey and benchmark. *Pattern Recognition*, page 111334, 2025.
 - [12] Vittorio Mazzia, Simone Angarano, Francesco Salvetti, Federico Angelini, and Marcello Chiaberge. Action transformer: A self-attention model for short-time pose-based human action recognition. *Pattern Recognition*, 124:108487, 2022.
 - [13] Nandhini T J and K Thinakaran. Deep neural network-based crime scene detection with frames. In *2023 Eighth International Conference on Science Technology Engineering and Mathematics (ICONSTEM)*, pages 1–8, 2023.
 - [14] Jaeyeong Ryu, Ashok Kumar Patil, Bharatesh Chakravarthi, Adithya Balasubramanyam, Soungsill Park, and Youngho Chai. Angular features-based human action recognition system for a real application with subtle unit actions. *IEEE Access*, 10:9645–9657, 2022.
 - [15] Neha Gupta, Suneet K. Gupta, Rajesh K. Pathak, Vanita Jain, Parisa Rashidi, and Jasjit S. Suri. Human activity recognition in artificial intelligence framework: a narrative review. *Artificial Intelligence Review*, 55(6):4755–4808, January 2022.
 - [16] Zengyuan Lai, Jiarui Yang, Songpengcheng Xia, Qi Wu, Zhen Sun, Wenxian Yu, and Ling Pei. Smart: Scene-motion-aware human action recognition framework for mental disorder group. *IEEE Internet of Things Journal*, 2024.
 - [17] Fahima Hajjej, Madiha Javeed, Amel Ksibi, Mohammed Alarfaj, Khaled Alnowaiser, Ahmad Jalal, Nawal Alsufyani, Mohammad Shorfuzzaman, and Jeongmin Park. Deep human motion detection and multi-features analysis for smart healthcare learning tools. *Ieee Access*, 10:116527–116539, 2022.
 - [18] Menghao Hu, Mingxuan Luo, Menghua Huang, Wenhua Meng, Baochen Xiong, Xiaoshan Yang, and Jitao Sang. Towards a multimodal human activity dataset for healthcare. *Multimedia Systems*, 29(1):1–13, 2023.
 - [19] Felicitas Macgilchrist Ben Williamson and John Potter. Re-examining ai, automation and datafication in education. *Learning, Media and Technology*, 48(1):1–5, 2023.
 - [20] Eleni Dimitriadou and Andreas Lanitis. A critical evaluation, challenges, and future perspectives of using artificial intelligence and emerging technologies in smart classrooms. *Smart Learning Environments*, 10(1), February 2023.
 - [21] Mukesh Kumar Saini and Neeraj Goel. How smart are smart classrooms? a review of smart classroom technologies. 52(6), 2019.

- [22] Hai Chuan Liu, Joon Huang Chuah, Anis Salwa Mohd Khairuddin, Xian Min Zhao, and Xiao Dan Wang. Campus abnormal behavior recognition with temporal segment transformers. *IEEE Access*, 11:38471–38484, 2023.
- [23] Walaa N Ismail, Hessah A Alsalamah, Mohammad Mehedi Hassan, and Ebtesam Mohamed. Auto-har: An adaptive human activity recognition framework using an automated cnn architecture design. *Heliyon*, 9(2), 2023.
- [24] Neha Gaud, Maya Rathore, and Ugrasen Suman. Mhcns-har: Multi-headed cnn-lstm based human activity recognition leveraging a novel wearable edge device for elderly health care. *IEEE Sensors Journal*, 2024.
- [25] Simin Zhu, Ronny Gerhard Guendel, Alexander Yarovoy, and Francesco Fioranelli. Continuous human activity recognition with distributed radar sensor networks and cnn-rnn architectures. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–15, 2022.
- [26] Neha Bansal, Atul Bansal, and Manish Gupta. Analyzing the variability of rnn hyperparameters and architectures for har with wearable sensor data. In *2024 3rd International conference on Power Electronics and IoT Applications in Renewable Energy and its Control (PARC)*, pages 232–237. IEEE, 2024.
- [27] Nikhil Kumar Singh and Koduru Sriranga Suprabhath. Har using bi-directional lstm with rnn. In *2021 International Conference on Emerging Techniques in Computational Intelligence (ICETCI)*, pages 153–158. IEEE, 2021.
- [28] Yue Hu, Xiao-Qing Zhang, Li Xu, Feng Xian He, Zhao Tian, Wei She, and Wei Liu. Harmonic loss function for sensor-based human activity recognition based on lstm recurrent neural networks. *IEEE Access*, 8:135617–135627, 2020.
- [29] Ahmed Younes Shdefat, Nour Mostafa, Zakwan Al-Arnaout, Yehia Kotb, and Samer Alabed. Optimizing har systems: comparative analysis of enhanced svm and k-nn classifiers. *International Journal of Computational Intelligence Systems*, 17(1):150, 2024.
- [30] Kholoud Maswadi, Norjihan Abdul Ghani, Suraya Hamid, and Muhammads Babar Rasheed. Human activity classification using decision tree and naïve bayes classifiers. *Multimedia Tools and Applications*, 80(14):21709–21726, 2021.
- [31] Aniruddha Prakash Kshirsagar and H Azath. Yolov3-based human detection and heuristically modified-lstm for abnormal human activities detection in atm machine. *Journal of Visual Communication and Image Representation*, 95:103901, 2023.
- [32] Amr A Mohy, Hesham A Bassioni, Elbadr O Elgendi, and Tarek M Hassan. Innovations in safety management for construction sites: the role of deep learning and computer vision techniques. *Construction Innovation*, 2024.

- [33] Weili Fang, Botao Zhong, Neng Zhao, Peter Ed Love, Hanbin Luo, Jiayue Xue, and Shuangjie Xu. A deep learning-based approach for mitigating falls from height with computer vision: Convolutional neural network. *Advanced Engineering Informatics*, 39:170–177, 2019.
- [34] Weili Fang, Lieyun Ding, Peter ED Love, Hanbin Luo, Heng Li, Feniosky Peña-Mora, Botao Zhong, and Cheng Zhou. Computer vision applications in construction safety assurance. *Automation in Construction*, 110:103013, 2020.
- [35] Shuai Tang and Mani Golparvar-Fard. Machine learning-based risk analysis for construction worker safety from ubiquitous site photos and videos. *Journal of computing in civil engineering*, 35(6):04021020, 2021.
- [36] Nipun D Nath, Amir H Behzadan, and Stephanie G Paal. Deep learning for site safety: Real-time detection of personal protective equipment. *Automation in construction*, 112:103085, 2020.
- [37] Xiaoyu Liu, Feng Xu, Zhipeng Zhang, and Kaiyu Sun. Fall-portent detection for construction sites based on computer vision and machine learning. *Engineering, Construction and Architectural Management*, 32(3):1499–1521, 2025.
- [38] Haitao Wu, Botao Zhong, Heng Li, Peter Love, Xing Pan, and Neng Zhao. Combining computer vision with semantic reasoning for on-site safety management in construction. *Journal of Building Engineering*, 42:103036, 2021.
- [39] Ibrahim Yousif, Jad Samaha, JuHyeong Ryu, and Ramy Harik. Safety 4.0: Harnessing computer vision for advanced industrial protection. *Manufacturing Letters*, 41:1342–1356, 2024.
- [40] M Ajay Srivathsan and R Rajesh. Work track: An automated employee activity tracking and monitoring system.
- [41] P Yuganthini, A Vigneswari, S Jancy, MD AntoPraveena, et al. Activity tracking of employees in industries using computer vision. In *2021 5th International Conference on Trends in Electronics and Informatics (ICOEI)*, pages 1321–1329. IEEE, 2021.
- [42] Niloy Sikder and Abdullah-Al Nahid. Ku-har: An open dataset for heterogeneous human activity recognition. *Pattern Recognition Letters*, 146:46–54, 2021.
- [43] M Rashmi, TS Ashwin, and Ram Mohana Reddy Guddeti. Surveillance video analysis for student action recognition and localization inside computer laboratories of a smart campus. *Multimedia Tools and Applications*, 80(2):2907–2929, 2021.
- [44] Chakradhar Pabba and Praveen Kumar. A vision-based multi-cues approach for individual students’ and overall class engagement monitoring in smart classroom environments. *Multimedia Tools and Applications*, 83(17):52621–52652, 2024.

- [45] Kristina Host and Marina Ivašić-Kos. An overview of human action recognition in sports based on computer vision. *Heliyon*, 8(6), 2022.
- [46] K Latha, Mohamed Musammil, et al. Human action recognition using deep learning methods (cnn-lstm) without sensors. In *2022 1st International Conference on Computational Science and Technology (ICCST)*, pages 447–451. IEEE, 2022.
- [47] Guo Liang Goh, Guo Dong Goh, Jing Wen Pan, Phillis Soek Po Teng, and Pui Wah Kong. Automated service height fault detection using computer vision and machine learning for badminton matches. *Sensors*, 23(24):9759, 2023.
- [48] Virender Singh, Swati Singh, and Pooja Gupta. Real-time anomaly recognition through cctv using neural networks. *Procedia Computer Science*, 173:254–263, 2020.
- [49] DG Shreyas, S Raksha, and BG Prasad. Implementation of an anomalous human activity recognition system. *SN Computer Science*, 1(3):168, 2020.
- [50] Rohit Nale, Mahesh Sawarbandhe, Naveen Chegogoju, and Vishal Satpute. Suspicious human activity detection using pose estimation and lstm. In *2021 International symposium of Asian control Association on Intelligent Robotics and Industrial Automation (IRIA)*, pages 197–202. IEEE, 2021.
- [51] TJ Nandhini and K Thinakaran. Deep neural network-based crime scene detection with frames. In *2023 Eighth International Conference on Science Technology Engineering and Mathematics (ICONSTEM)*, pages 1–8. IEEE, 2023.
- [52] Esakky Selvi, Malaiyalathan Adimoolam, Govindharaju Karthi, Kandasamy Thinakaran, Nagaiah Mohanan Balamurugan, Raju Kannadasan, Chitapong Wechtaisong, and Arfat Ahmad Khan. Suspicious actions detection system using enhanced cnn and surveillance video. *Electronics*, 11(24):4210, 2022.
- [53] Md Motiur Rahman, Deepti Gupta, Smriti Bhatt, Shiva Shokouhmand, and Miad Faezipour. A comprehensive review of machine learning approaches for anomaly detection in smart homes: Experimental analysis and future directions. *Future Internet*, 16(4):139, 2024.
- [54] K Menaka, R Vinoth Raj, Ch VSS Aravind, Vallepu Vamsi, Shaik Fardeen, and G Yugendar. Yolo algorithm-based suspicious activity detection in atm surveillance. In *2024 International Conference on Advances in Modern Age Technologies for Health and Engineering Science (AMATHE)*, pages 1–5. IEEE, 2024.
- [55] Shilpa Ankalaki and MN Thippeswamy. A novel optimized parametric hyperbolic tangent swish activation function for 1d-cnn: application of sensor-based human activity recognition and anomaly detection. *Multimedia Tools and Applications*, 83(22):61789–61819, 2024.

-
- [56] Yuxing Yang, Federico Angelini, and Syed Mohsen Naqvi. Pose-driven human activity anomaly detection in a cctv-like environment. *IET Image Processing*, 17(3):674–686, 2023.
- [57] Sri Mulyani, Anas A Salameh, Aan Komariah, Anton Timoshin, Nik Alif Amri Nik Hashim, R Siti Pupu Fauziah, Mulyaningsih Mulyaningsih, Israr Ahmad, and Sajid Mohy Ul din. Emotional regulation as a remedy for teacher burnout in special schools: Evaluating school climate, teacher’s work-life balance and children behavior. *Frontiers in Psychology*, 12:655850, 2021.
- [58] Alanood Abdulaziz, Makhmoor Bashir, and Abdulaziz Abdulmohsen Alfalih. The impact of work-life balance and work overload on teacher’s organizational commitment: do job engagement and perceived organizational support matter. *Education and Information Technologies*, 27(7):9641–9663, 2022.