Understanding human actions: Cognitive assessment and action segmentation using human object interaction

Dissertation Submitted in Fulfillment of the Requirements for the Degree of
PhD in Computer Science

by

Saif Iftekar Sayed
1001275259

Supervised by: Prof. Vassilis Athitsos

Department of Computer Science and Engineering
UNIVERSITY OF TEXAS AT ARLINGTON

December 2022

Dedicated to computer vision research community and efforts towards mental health...

Abstract

This dissertation contributes in development of systems for automatic understanding of human behavior having applications in medicine and surveillance. To enable cognitive assessment in adolescent kids, we developed a vision based, unobtrusive and automated cognitive assessment system called Activate Test of Embodied Cognition (ATEC). Our system can measure hyperactivity and response inhibition which can inform physicians to provide life-changing treatments for kids at an early age. More specifically we created an end-to-end activity recognition system for one of the ATEC task called Cross Your Body which can track multiple activities in an untrimmed video and provide a score that can be directly transferred to expert human's score with high inter-rater reliability. These scores can be utilized to measure executive functioning of kids which is one of the key factor to distinguish onset of ADHD in adolescent kids. We also introduced a new dataset towards development of robust activity recognition system.

We also studied the influence of human object interaction(HOI) in action segmentation task for long duration instructional videos under timestamp supervision. We created a first of it's kind timestamp supervised action segmentation system that utilizes HOI as another source of information and utilized transformer for improved temporal modelling. To enable research using HOI for multi-view action segmentation, we also created a new dataset called (3+1) Rec, which has 1799 long-duration, high quality videos comprising of 3 third person view and 1 egocentric view for each dish the subject is making in a kitchen environment.

Acknowledgments

I'd like to express my sincere gratitude to my professor Dr. Vassilis Athitsos for his constant support and encouragement to perform honest and quality research and make me a strong independent researcher. I would also like to thank my committe members Dr. Farhad Kamangar, Dr. William Beksi and Dr. Chris Conly for their valuable feedback and support for my thesis. I am also thankful for my fellow lab-mates Reza Ghoddoosian, Marnim Galib and Alex Dillhoff for their valuable brainstorming sessions and help towards my thesis. I wish a very good luck and success to all of my team-mates. In honor of the contribution aformentioned people had in my thesis, I have used the plural first person pronoun (we/us) in the thesis.

I am also thankful to Dr. Morris D. Bell and Filia Makedon for giving me an opportunity to work in a meaningful project that will help the society and challenge me to have cross-functional collaboration, build a dataset and a system that will be useful for the research community. I am also very thankful for all the volunteers who helped me with their time to create datasets that will help future research.

Table of Contents

1 Introduction

Automatic understanding of human behavior has several applications in medicine and surveillance. Analysing human actions can enable cognitive assessment of children by measuring their hyperactivity and response inhibition which can give physicians better understanding of their cognitive state. Automatic and non-invasive assessment for cognitive disorders will increase the affordability and reach for these detection methods and can prove life-changing in child's development. Human activity can also be analysed in common settings such as cooking in kitchen and understanding the information of human object interaction can give priors on the underlying activity they are performing.

In the first section, we focus on cognitive assessment. We introduce specifically a new dataset towards development of automated system for the Activate Test of Embodied Congition (ATEC), a measurement that evaluates cognitive skills through physical activity. Evaluating cognitive skills through physical activity requires subjects performing wide variety of tasks with varying levels of complexity. To make the system affordable and reachable to larger population, we created an automated system that can score these human activities as accurately as an expert. To this end, we developed and activity recognition system for one of the most challenging task in ATEC, called *Cross-Your-Body* which can evaluate attention, response inhibition, rhythm and co-ordination, task switching, working memory. We created and annotated the dataset that enabled us for training of vision based activity segmentation models. First, we developed a very accurate system that requires trimmed video as input where every video has only one action and predicts the human activity by tracking the human pose features. Second, we improved the system to create an end-to-end method that can track multiple activities in an untrimmed video which enabled the generation of scores that can directly transfer to the expert human's score with high inter-rater reliability.

In the second section, we study action segmentation in instructional videos under timestamp supervision. In the action segmentation domain, the goal is to temporally divide the input video into set of sequential actions. In fully supervised setting the training

labels are given for every frame while in weakly supervised settings, the labels are at video level and are sequence of actions. While the weakly supervised labels reduces the annotation time for labeling videos, it lacks test performance as comparable to a fully supervised setting by a big gap. To alleviate this problem, in addition to the sequence of actions, timestamp supervision also adds a single frame number for each action which adds significant constraints on when each activity may happen.

We study timestamp supervision under several scenarios. First, we created a new approach that utilizes human object interaction (HOI) as a source of information other than the exisiting flow and rgb information. The system creates new pseudo-groundtruth by expanding the the timestamp annotations using the information from an off-the-shelf pre-trained HOI detector, that requires no additional HOI-related annotations. We also improved the temporal modelling system from temporal convolution based to transformer one which further improved the performance. Second, to enable the research on HOI and multi-view action segmentation, we created a first of it's kind dataset called (3+1)Rec, which has 1799 long-length, high quality videos comprising of 3 third person view and 1 egocentric for each dish the subject is making in a kitchen environment.

In summary, the main contributions in this thesis are as follows:

- We introduce a activity recognition system called Cross-Your-Body, designed by psychologists that can provide accessible and affordable tools for predicting onset of cognitive disorders such as ADHD by tracking spatio-temporal features.

- Our system can generate scores which can be directly translated to measure executive functioning which is one of the key factor to distinguish onset of ADHD in adolescent kids.

- We present a novel approach for timestamp based action segmentation that utilizes human object interaction as another source of information for efficient activity prediction and also adopted transformers for improved temporal modeling.

- We present a large and public real-life dataset of 30 subjects which will be useful in multi-view action segmentation. This dataset consists of high-quality, long-range videos that has 3 third person and 1 egocentric view.

Finally, the list of published papers that constitute this thesis is provided below:

- Saif Iftekar Sayed, and Vassilis Athitsos. Cross Your Body: a Cognitive Assessment System for Children.*In Proceedings of International Symposium on Visual Computing*, pages 97-109, 2021

- Saif Iftekar Sayed, Konstantinos Tsiakas, Morris Bell, Fillia Makedon and Vassilis Athitsos. Cognitive assessment in children through motion capture and computer vision: the cross-your-body task. *In Proceedings of the international Workshop on Sensor-based Activity Recognition and Interaction*, pages 1-6, 2019

The list of published papers that constitute collaborative efforts during my research in the field of action segmentation and robotics is provided below:

- Reza Ghoddoosian, Saif Iftekar Sayed, and Vassilis Athitsos. Hierarchical modeling for task recognition and action segmentation in weakly- labeled instructional videos. *In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2022

- Reza Ghoddoosian, Saif Iftekar Sayed, and Vassilis Athitsos. Action duration prediction for segment-level alignment of weakly-labeled videos. *In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2053–2062, 2021

- Michail Theofanidis, Saif Iftekar Sayed, Joe Cloud, James Brady and Fillia Makedon. Kinematic estimation with neural networks for robotic manipulators. *In Proceedings of the International Conference on Artificial Neural Networks*, pages 795-802, 2018

Part 1:
Cognitive Assessment

2 Cognitive Assessment in Children through Motion Capture and Computer Vision: The Cross-your-Body task

2.1 Abstract

This paper focuses on creating video-based human activity recognition methods towards an automated cognitive assessment system for children. We present the Activate Test for Embodied Cognition (ATEC), which assesses executive functioning in children through physical/cognitive tasks. Detecting activities for children is challenging due to high amount of random motion and variability. This paper focuses on creating a ubiquitous and non-intrusive activity recognition system for upper-body movements. Our proposed methods are evaluated on real-world data from children performing the Cross-your-Body task. The dataset includes 15 children performing 8 types of activities, resulting to 1900 annotated video samples.

2.2 Introduction

Self-regulation, which generally refers to a complex of acquired, intentional skills involved in controlling, directing, and planning one's cognition, emotions and behaviors [1], is an important mechanism associated with variety of outcomes, including school readiness and performance [2]. Executive function refers to the mental processes that enable humans to plan, organize, problem-solve as well as manage their impulses, including cognitive flexibility, working memory, and inhibitory control [3]. Children who face deficits

in executive functions are highly likely to present attention disorders [4]. ADHD or attention deficit hyperactivity disorder is a psychiatric neurodevelopmental disorder found in children and young adolescents and it can start as early as age 6 [5, 6]. Cognitive impairments in executive functions can not only cause bad performance in school settings, but can also show repercussions in family, employment and community settings which can result to several socioeconomic problems, resulting to low self-esteem and self-acceptance [7]. In order to quantify executive function in children, traditional assessments include either paper or computer-based activities, e.g., the NIH toolbox. However, recent studies suggest assessments which include physical activities, for example the Head-Toes-Knees-Shoulders (HTKS) task, which has been extensively tested on 208 children and elicits psychometric measures through physical performance [8].

Our research includes the development of ATEC; the ACTIVATE Test for Embodied Cognition, which includes a set of physical tasks with cognitive demands to assess executive function in motion. A core ATEC task is *Cross-your-Body*, which follows and extends the basic HTKS rules, and is designed to assess working memory and attention, bilateral coordination, rhythm and self-regulation. The HTKS rules include four behavioral activities: "touch your {*head, toes, knees, shoulders*}". The subject is initially instructed to touch the announced body part. Then, the task introduces task switching and requires the child to touch the body part in an "opposite" fashion (e.g. touch knees when told to touch shoulders).

Cross-your-Body requires the subject to touch the correct body part with the hand from the opposite side. Crossing the midline is an integral skill related to bilateral coordination that children learn from infancy. Poor midline crossing can affect reading (tracking with the eye from left to right) and writing (using their dominant hand across the writing page) skills. Moreover, Cross-your-Body is designed to assess rhythm; the child is asked to repeat each movement three times, alternating sides in a timely manner. Task performance is determined both in terms of accuracy (touch the correct part) and rhythm (perform movements in a rhythmic manner). Manual scoring requires a human rater to watch the videos and score the child based on the task rules (accuracy, rhythm) and can be time-expensive and often ambiguous.

The main purpose of our research is to build an automated scoring system for Cross-your-body, which detects and analyzes the performed activities to assess accuracy and rhythm. Current systems like Cognilearn [9] utilize state-of-the-art computer vision algorithms by capturing color frames from the Kinect V2 camera and provide an interface for motion capture and analysis. Deep Learning architectures were proposed as the backbone model [10] and tested on synthetic data with adults performing the task. For this paper,

our dataset includes collected data during the ATEC assessments with children between 5-10 years old in classroom environments.

The main contribution of our paper is a video-based activity recognition system for the Cross-your-Body task, which recognizes the *active hand* that performs the movement, estimates specific spatial hand positions for efficient feature extraction, while including low-confidence prediction class. Our experiments on real-world data indicate the efficiency of our method for reliable and user-independent activity prediction effective on scaling number of users. The structure of the paper is as follows: Firstly, we present related work on similar applications, highlighting the motivation of our work. Then, we present the system architecture and our experimental approach using machine learning techniques. We discuss our experimental protocol and results, describing the data collection and annotation process. Finally, we conclude with some final remarks and our future work.

2.3 Related Work

Emerging technologies have influenced many medical related processes such as diagnosis, rehabilitation and treatment. Computer and data science have opened up another realm of capturing and analyzing data in an automated fashion. These implementations not only demand higher prediction accuracy but also focus on user engagement. Active video game play using consoles like Microsoft Kinect can help rehabilitation of children suffering from Cerebral Palsy [11]. Systems can also monitor the attention state of the child using eye-trackers towards user-friendly and personalized interfaces [12]. Moreover, virtual reality games have been developed for assessment and rehabilitation of children with attention deficit [13].

Inattention and/or hyperactivity or impulsivity symptoms can cause alterations in a person's human movements and reactions [14]. This is the main reason for exploring several sensor-based human activity recognition systems. Such sensors can be employed on the human body or can be placed in the surrounding environment. Hypothesis testing by studying the readings given by wearables showed significant differences for ADHD patients compared to non-ADHD controls [15, 16]. Recent advancements in deep learning have led to the use of convolution neural networks (CNN) to extract embedded acceleration patterns and provide objective measures to help diagnose ADHD [17], but such approaches can be obtrusive since the subject has to wear different types of wearable sensors.

Camera-based settings can provide an unobtrusive environment for data collection

and computer vision and deep learning methods can be used to extract important spatio-temporal features and recognize patterns of interest. In a previous work, a camera-based system was proposed for the HTKS task [9] and evaluated on adults, which used deep learning techniques to extract body pose information for human activity recognition following a frame-based approach. In this work, we follow a segment-based approach, since the nature of activities involved in Cross-your-Body (CYB) is more complex compared to HTKS, i.e., crossing the midline and performing the task in rhythm. Moreover, our proposed methods are evaluated on real-world data from children performing the task.

2.4 CYB System

The primary goal of the system is to reliably recognize the type of performed activity given a video segment. The system initially detects the subject and then tracks its hands over time to recognize the performed activity, as well as when the activity was performed. The overall system is illustrated in Figure [2.1]. The system receives body-motion data from Kinect and then it produces a set of spatio-temporal features used to predict the activity performed by the child. The system include two modules: the *Acquire* and the *Track* module. The Acquire module takes care of capturing and analyzing each frame to create an accurate skeleton vector for the entire video, which after preprocessing it is passed to the Track module for gesture recognition.

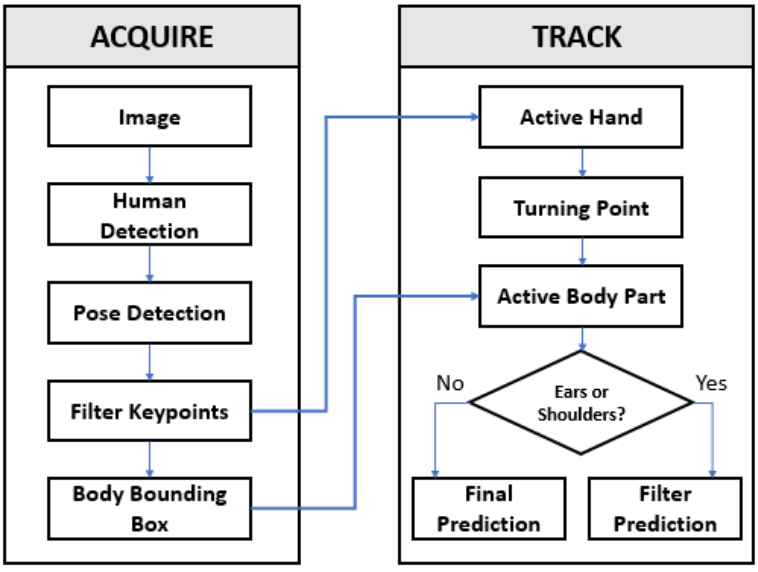


Figure 2.1: System Architecture

2.4.1 Acquire Module

The Acquire module initially fetches the RGB frames from the Kinect and detects the subject of interest, in order to detect and filter its 2d pose, divide the body into regions based on height in order to produce a filtered skeleton joint vector for activity recognition.

2.4.1.1 Human Detection

The first step of the process is to reliably detect the subject of interest. Due to the naturalistic environment, there are often multiple people in the background during the ATEC assessment was essential to first isolate the subject of interest in order to reduce computation complexity. YOLO v3 [18] was used to detect the humans in the scene because of his fast and accurate inference and then based on an empirically decided spatial threshold, the bounding box which fell in that criteria was chosen as the subject of interest which was then was passed to the pose estimation.

2.4.1.2 2D Body Pose Estimation

Microsoft Kinect V2 has an RGB capture resolution of 1920x1080 pixels with a Time-of-Flight depth data as an 512x424 resolution image [19]. The field of view for depth is 70 degrees horizontally and 60 degrees vertically [20]. In this paper, a Kinect V2 is used for acquisition since it tracks more joints and has a higher motion tracking accuracy, with greater stability. Kinect's SDK provides it's own stock SDK that can be used to get the 3D body pose of the skeleton, but the problem is that Kinect's skeletal tracking doesn't perform well under occlusions [21]. In our work we are still using kinect, since it gives us the color and the depth channels of the environment. Currently we are considering only the color modality of the acquisition for our analysis as it is much more consistent and less noisy than kinect's skeletal tracker.

For locating the joints in the RGB images, we leverage the recent advancements in deep learning where data has been trained on millions of images encompassing scenarios like self-occlusions and networks like OpenPose [22] can be very useful to provide accurate estimation of body pose. We have employed the skeleton map result based on the 2016 COCO keypoints dataset challenge and the skeleton structure provided by openpose is as shown in the figure 2.2. Each joint is represented by a 2D vector in the cartesian co-ordinate space. The extracted tensor for a video can be expressed as follows:

$$P_i = [B_1, B_2, ..., B_{18}], i = [1, 2, ..., n] \tag{2.1}$$

Where P_i is a set of 18 2D keypoints location representing respective body joints for a given frame i in a video sample.

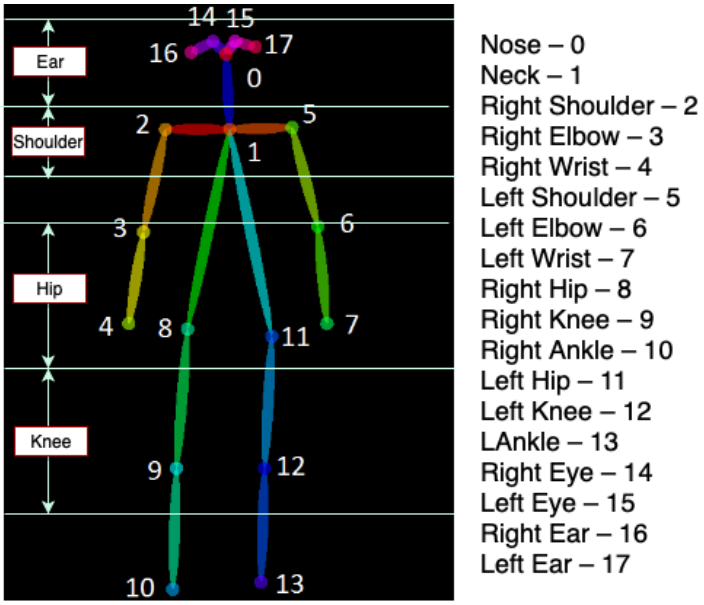


Figure 2.2: Openpose Skeleton Map

2.4.1.3 Filter Keypoints

These points in a video sample are further filtered, where intermediate body points are interpolated in case of misclassifications.

2.4.1.4 Body Bounding Box

After filtering the keypoints, the first frame of the video sample is used to divide the body into 4 areas. This area is based on the required class labels of ears, shoulders, hip and knees. The height of the person is computed using the distance between ear and ankle with a padding of 50 pixels. Using a fixed width consistent with all the subjects the body is spatially divided into 4 parts using a fixed percentage of height per body part. The divided region helped the classifier understand which part the subject was trying to touch and the use of these regions will be explained in the active body part prediction section.

2.4.2 *Track*

Tracking modules involves classifying the activity by using the 2d keypoints. This involves finding which hand was active in other words which hand was performing the gesture/activity and which body part it was touching/interacting with. This involves finding the relevant frames in the video which gives the maximum information for correct classification as well as extracting those spatio-temporal features. The tracking module first finds which hand was active, then tracking the spatial positions of the palm decides where the touching of body happened based on the velocity and curvature of the palm trajectory and eventually classifies the active body part. These steps will be elaborated in the following sections.

Task ID	Task Nature	Video Instructed - "Cross your body touch your.."	Actual Movement Intended
1	Cross Body - Trial 1	E,S,H,K	E,S,H,K
2	Cross Body - Trial 2	E,S,H,K	E,S,H,K
3	Cross Body Ears - Knees	E,K,K,E,K,E,E,K	K,E,E,K,E,K,K,E
4	Cross Body Hips - Shoulders	S,H,H,S,H,S,S,H	H,S,S,H,S,H,H,S
5	Cross Body Hips - Combined	E,H,K,S,K,H,E,K,H,S,E,S	K,S,E,H,H,E,H,K,E,S,H,K,H

Table 2.1: Cross-your-Body versions and rules. Trials 1, 2 do not have cognitive demands; the rest of them introduce task switching

2.4.2.1 *Detect Active Hand*

Before tracking the hand it was important to compute the palm position and not the wrist position and since the body pose estimator of OpenPose didn't give the palm position, an approximate estimation of palm was done by extending the vector passing from the elbow through wrist by a magnitude of 1.25 times the magnitude of the vector between elbow and palm, where the elbow vector is added by the scalar elementwise (Eq. 2.2).

$$\vec{B}_{Palm} = \vec{B}_{elbow} + \|\vec{B}_{wrist} - \vec{B}_{elbow}\| * 1.25 \tag{2.2}$$

The experimental protocol dictated that a valid touch of body part is supposed to be done by the opposite hand-body pair (midline crossing). So to check whether a palm is on the other side of the body a reliable anchor point was needed to decide the horizontal center of the body. Based on the data visualizations and the experimental protocol, subjects were instructed to stand at a fixed location in the scene, hence their feet position is fixed in the whole video and can act as anchor points. The vector passing from midpoint of the 2 ankle and parallel to the y-axis was considered as a border dividing the body into

left and right side.

$$\vec{B}_{midankle} = \frac{\vec{B}_{left_{ankle}} + \vec{B}_{right_{ankle}}}{2} \tag{2.3}$$

For a video sample, let C_{left} and C_{right} be the set of frame indices in a video sample where the system predicted that the hands are in opposite sides of the body. These 2 sets are then passed to a filter where:

$$C_{left}[n] = \begin{cases} n, & \text{if } \vec{B}_{midankle_x} - \vec{B}_{leftpalm_x} > 0 \\ 0, & \text{otherwise} \end{cases} \tag{2.4}$$

$$C_{right}[n] = \begin{cases} n, & \text{if } \vec{B}_{midankle_x} - \vec{B}_{rightpalm_x} < 0 \\ 0, & \text{otherwise} \end{cases} \tag{2.5}$$

After getting the frame indices where the left and right palms cross the midline, these indices were further analyzed to get the active hand which is computed by analyzing the velocities of the palms. The protocol for a correct body movement indicates that subjects moves their hands from a rest position, cross the midline, touch the body part of the opposite side and then bring the hand back to rest by crossing the midline again. More formally, this means that the subject's hand will cross the midline twice (once while approaching and once while leaving) and the hand's velocity vector in the x direction will have opposite direction as it cross the midline. Note that there might be cases when the subject's hand may be in the cross state and still they did touch the body part. If both hands were in cross state, then the hand which crosses the midline later was assumed as the active hand. If none of the hands were in a state of cross, then the classifier is not confident of the prediction and not undergo further steps.

2.4.2.2 Get Active Body Part

Once the system detects the active hand, the next step of the algorithm is to identify which body part is touched. This is the crux of the system as based on the data analysis for the kids, there is a very high intra-class variability on the style of how a subject performs an activity in terms of distance from the palm and the intended body part touch and velocities the palm approaches and leaves a body part after touching. The trajectory of an active hand's palm can be considered as a curve defined in a parametric form by equations $x = x(t)$ and $y = y(t)$, where t is time and x and y are the co-ordinates of the palm. So, a

curvature at any point on the trajectory can be given as:

$$K = \frac{|x'y'' - y'x''|}{[(x')^2 + (y')^2]^2} \tag{2.6}$$

Here x' and x'' are the first and second derivative of the x co-ordinates and similarly for y-co-ordinates. Before getting the points of curvature, the trajectory is smoothed by using a 1 dimensional smoothing filter. Using the spatial positions of the trajectory, further they were filtered based on the velocity of the hand. An empirical threshold of 2 was chosen to filter the positions. Once the spatial positions of the hands are known where the palm trajectory showed highest curvature and the palm was moving slowly, then the mean of these spatial positions is taken and using the bounding boxes of the relevant body parts as shown in figure 2.2, the final prediction is done. If the prediction is ear or shoulder it goes through further processing of ear and shoulder classification module which computes more spatio-temporal features and produces a final prediction (ear/shoulder) by passing the features to a decision tree algorithm.

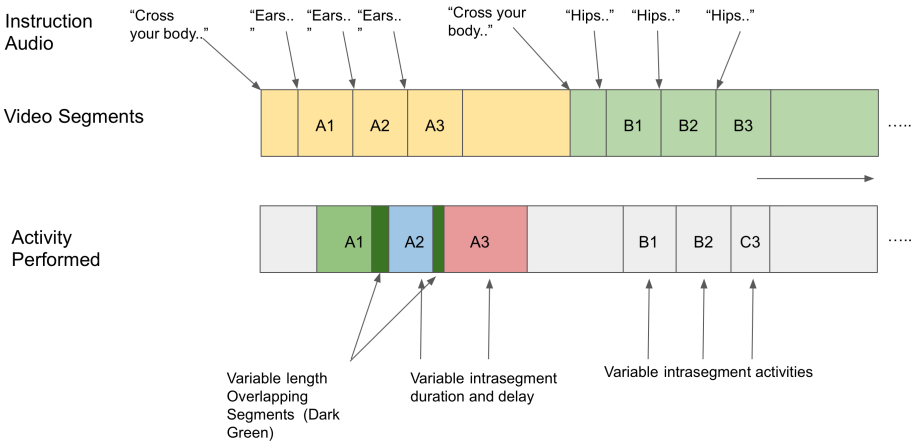


Figure 2.3: Temporal Analysis of activities and rules

The spatio-temporal features into consideration are:

- Hand Shoulder and Hand Ear Distance: To compare that the hand was much closer to ear or shoulder, euclidean distances between the active hand's palm and the opposite side's ear and shoulder were computed taking for the spatial positions where the curvature of the trajectory was maximum and velocity was low. Note, there can be a multiple points where this criteria of curvature and velocity may be true over

time in a video so a mean of these euclidean distances was taken to compare if it was close to ear or shoulders.

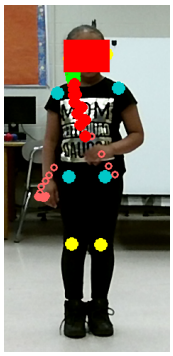
- Shoulder-Palm-Ear Angle: This is a very important feature that can be used to differentiate the touching of ear or shoulder. For example the angle made by the left shoulder, left elbow and left palm will be much closer to the angle made by the left shoulder, left elbow and right ear compared to left shoulder, left elbow and right shoulder when the actual activity performed was left hand touching right ear. Using the formula 2.7, we can compute an angle between 3 joints and the above logic will yield into 3 angles namely Θ_{palm}, $\Theta_{shoulder}$ and Θ_{ear} resulting into addition of information for better prediction between ears and shoulder.

$$\begin{cases} \overrightarrow{AB} = A - B \\ \overrightarrow{BC} = B - C \\ \Theta = \dfrac{\sum_{i=1}^{n} \cos^{-1}\left(\dfrac{\overrightarrow{AB_i} \cdot \overrightarrow{BC_i}}{|\overrightarrow{AB_i}| \cdot |\overrightarrow{BC_i}|}\right)}{n} \end{cases} \tag{2.7}$$

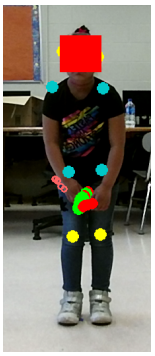
2.4.3 System Protocol

In the context of our research study, children were participated to perform the ATEC activities, including the Cross-your-Body task. For our experiments, we created our dataset including data from 15 participants performing five versions of the task in 2 sessions with a gap of 2 months. In order to ensure a high-fidelity assessment system, all instructions are pre-recorded and same for all children. A large screen is used to display a theme-based music video, where the on-screen host, Aliza, instructs the child to perform the task following her "Cross-your-Body" song. Two Kinects (front and side) are used to capture the movements. The distance between the subject and both of the Kinects is 2m. Table 2.1 illustrates the task versions. Before each task, the subject is shown a task instruction, as well as a demonstration video clip , explaining the task rules. For example, for Trials 1 and 2, the child is told to perform three touches, touching the announced body part, using the hand from the opposite side and alternating sides. For the rest of the tasks, the child is instructed to follow the "opposite" rules; task 3 switches ears and knees, task 4 switches shoulders and hips, while task 5 includes both rules. Every subject undergoes through the same process and there is no prior instructions given other than the video instructions.

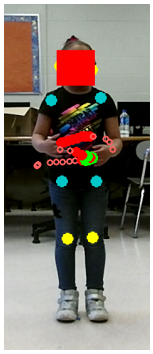
A temporal analysis of the activities performed vs instructed can be seen in fig 2.3 which indicates the progression of a task and is divided in 2 parts: video segments and activity performed. Referring to the row of video segments, the main task is made of



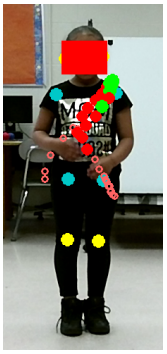
(a) Left Hand Right Ear



(b) Left Hand Right Knee



(c) Right Hand Left Hip



(d) Left Hand Left Shoulder

Figure 2.4: Sample predictions for a subject. Body joint positions are highlighted in yellow (ear), cyan (shoulders and hips) and yellow (knee). Active hand positions for a hand are in thick red, while unfilled red circles indicate an inactive hand. Points with high curvature and low speed are in green.

several sub-tasks and is highlighted in yellow and green respectively. A sub-task begins when the instruction video starts saying ”Cross your body..” while a task segment begins when the actual body part to touch was said. The instruction time gap between every task segment is 1 second and there are 3 segments in every sub-task. Each task segment is an activity of touching a body part and there were overlapping of activities, in other words if A1 and A2 are 2 task segments of touching ear, the subject might be partially touching the ear or moving hand away from that ear while lifting the other hand and approaching the other ear intended for A2. Also since there involves processing of working memory a subject would perform the activity with varying delay after the instruction and since there also involves switching of rules, the activity performed may or may not be correctly done as instructed.

2.5 Experiments and Results

Based on the problem definition 8 activity classes were chosen as LHRE, RHLE, LHRS, RHLS, LHRH, RHLH, LHRK, RHLK, where LH stands for left hand and RH stands for right hand. Also there was a ninth class as nooo indicating system low-confidence. The recorded videos were segmented based on the timestamps of the presented stimuli and a frame level activity annotation was performed resulting into 1900 video samples, where the average frame length of the video was 28. Each annotated segment refers to one (out of three) movements. First 5 subjects were used to set the thresholds required by the algorithm (e.g., bounding box) and the next 10 subjects were used for testing.

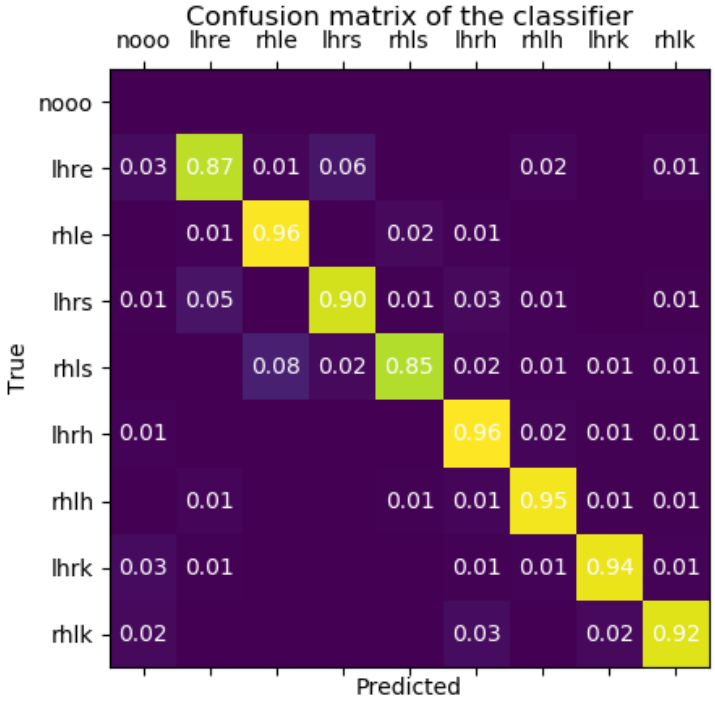


Figure 2.5: Confusion matrix for the test split

As illustrated in figure 2.4, these are some sample predictions of the system and it can be clear that system could focus more on the spatial locations in the trajectory of the hand where it could extract maximum information. The accuracy of the system was measured for these 10 test subjects by comparing with the ground truths and the system could achieve an overall 89.95%. The confusion matrix for the predictions can be seen as in figure 2.5. Based on the confusion matrix, the prediction of ear or shoulder needs

further fine-tuning as the system still gets confused since the distance between ear and shoulder is small and the palm position prediction is not able to capture the fine motion of palm. One way to improve this is to use the depth modality or skin detection for better segmentation of hand and in-turn help to compute the distance and angles between palm, ear and shoulder much more accurately.

2.6 Conclusion and Future Work

A video-based activity recognition system for cognitive assessment in children was presented. Data were collected from the Cross-your-Body task during the ATEC administration with children between ages 5-10. Overall 1900 video samples were segmented and annotated and the system gave an overall accuracy of 89.95%. The automated system was also tested with manual scoring and gave accurate results as comparatively. The system successfully applied temporal modelling dependencies to capture the aforementioned activities. Moving forward, the system will be extended to perform automated scoring given the task rules. Our ongoing work on temporal localization of the activity will provide us with insights on how to automatically score both for accuracy (which part is touches) and rhythm (when the touch occurs). Intelligent interfaces will be used to provide the experts with intuitive data visualization to enhance their decision making.

3 Cross Your Body: A Cognitive Assessment System for Children

3.1 Abstract

While many action recognition techniques have great success on public benchmarks, such performance is not necessarily replicated in real-world scenarios, where the data comes from specific application requirements. The specific real-world application that we are focusing on in this paper is cognitive assessment in children using cognitively demanding physical tasks. We created a system called Cross-Your-Body and recorded data, which is unique in several aspects, including the fact that the tasks have been designed by psychologists, the subjects are children, and the videos capture real-world usage, as they record children performing tasks during real-world assessment by psychologists. Other distinguishing features of our system is that it's scores can directly be translated to measure executive functioning which is one of the key factor to distinguish onset of ADHD in adolescent kids. Due to imprecise execution of actions performed by children, and the presence of fine-grained motion patterns, we systematically investigate and evaluate relevant methods on the recorded data. It is our goal that this system will be useful in advancing research in cognitive assessment of kids.

3.2 Introduction

Mental illness can cause several undesirable effects on a person's emotional, mental or behavioral states[23]. It is estimated that around 450 million people are currently affected by mental health issues, including schizophrenia, depression, attention-deficit hyperactivity disorder(ADHD) and autism spectrum disorder (ASD)[23]. More specifically, ADHD, which is a psychiatric neurodevelopmental disorder found in children and young adolescents, can have its traces evident as early as age 6 [5]. Such traces may include deficits in

executive functions[4] inhibiting them to perform mental processes like planning, organizing, problem-solving as well as managing their impulses, including working memory, cognitive flexibility and inhibitory control [3]. These developmental shortcomings causes detrimental effects not only in their school performances but also at a higher level, trigger many negative effects in family, employment and community settings which can result into several socio-economic problems, causing low self-esteem and self-acceptance [7]

While current methods use fMRI or sMRI scans[24], facial expressions[25] or clinical notes[26],these methods provide good prognosis of the subject's cognitive condition at the brain activity/blood flow level, but are expensive and not portable. Embodied cognition tackles this problem with an understanding that our sensorimotor experiences with our social and physical environment helps in developing and shaping our higher cognitive processes[27]. Inspired by this approach, research [8] adopted the Head Toe Knee Shoulder(HTKS) task to assess these psychometric measures of self-regulation through physical performance for 208 subjects using obtrusive wearable sensors.

Similarly a recent work [28] created a system called ATEC whose scores showed significant correlation with concurrent measurements of executive functions and significant discriminant validity between At-risk children and Normal Range children on multiple pre-existing tests like the CBCL Competency, CBCL ADHD Combined score, BRIEF-2 Global Executive Composite, BRIEF-2 Cognitive Regulation Index and SNAP-IV ADHD Combined Score. They measure psychometric scores such as Response Inhibition, Self-Regulation, Rhythm and Coordination which constitutes the Executive Function(EF) Score and Working Memory Index Score. These scores provide valuable information for differentiating adolescent kids susceptible to ADHD as compared to normal[29]. They used human annotators to evaluate the activities performed by kids, while current systems like [30], [9] use computer vision technique to detect these activities, but they do not produce scores that can be translated to produce these psychometric scores. The main contribution of the paper is to create a system that can produce an automated score of rhythm and accuracy, which is the fundamental component of creating the psychometric measures by utilizing the recording and scoring protocol followed ATEC system for the cross your body task and compare it with the human scores. The data has been recorded in real-time in an indoor environment, and shows children performing fine-grained activities. In this system, the children follow instructions to touch, using each hand, a specific body part (ear, shoulder, knee, or hip) on the other side of the body.

3.3 Related Work

Neuroimages like fMRI or sMRI have been traditionally used as clinical data for applications such as identification of ADHD [31, 24] which use CNN to identify local spatial patterns of modulations of blood flow in a section of brain. While there are compelling research methods that can separate kids with ADHD from control subjects, these techniques require costly acquisition of brain scans and face the issue of portability. Instead of learning such information at a micro-level, one can study the effects of the disorder at macro-level, as human movements and how they can be affected by hyperactivity and/or inattention[14]. This inspired several wearable sensor based approaches[15, 16] and a significant difference was evident between non-ADHD controls and ADHD patients, but such methods require obtrusive sensors.

With the advent of sophisticated activity recognition systems, such human movements can be tracked and was first tried on adults using HTKS task[9]. The prior work most related to ours is the method presented in [30], which was also applied to videos of children performing the Cross-Your-Body task.

Our work has significant differences, and advantages, compared to [30] and can be used for fully-automated scoring of the children's performance. The method of [30] cannot be used on its own for fully automated scoring, due to two limitations which our method overcomes: the first limitation was that it required human annotations to convert a child's performance into several segmented videos, each segment corresponding to a specific instance of the hand moving to touch a body part. Second, the output of the system in [30] simply classified each video segment, without providing frame-level labels indicating when exactly the subject touches the instructed body part. Frame-level labels are necessary for scoring the rhythm and timing of each child's performance, which is an important aspect of the human experts' evaluation protocol.

3.4 Data Acquisition and Protocol Definition

The goal of the system is to facilitate the development of an automated scoring environment, whose output correlates as much as possible with scores produced by human experts for the same videos. Simulated datasets have been previously created for similar tasks using adults[9], but they lack the unique motion dynamics that the children participants display in our recording.

3.4.1 Data Recording

The subjects were recorded at multiple indoor locations and strict quality control was maintained (such as keeping the distance from the subjects to the camera within a specified range), to ensure consistent acquisition quality and every kid was given the same instruction. The data was recorded using Kinect V2. A screen is used to display a music video where the host instructs the kids to perform the task following the song of "Cross-your-body". There are 5 tasks overall and each task has varying sub-segment. In a single sub-segment, the subject is told to perform 3 touches based on the announced body part using the hand from the opposite side and alternate sides, for example "Cross your body touch your hips, hips, hips. For first 2 tasks the subject is required to touch the same body part as instructed, but for the other 3 tasks the challenge becomes cognitively demanding as they are told to follow opposite rules, for example in task 3 subject requires to touch ears when instructed to touch knees and vice versa. Similarly task4 switches shoulders and hips and task5 includes rules of both task 3 and 4. The scoring system is followed according to the research done by Bell [28]

3.4.2 Scoring Scheme

The objective of our data processing module is to apply activity segmentation algorithms to evaluate the performance of the participants. The scoring protocol created by the psychologist experts specifies 2 scores: accuracy and rhythm. The accuracy score depends on the amount of times that the subject touches the desired body part correctly, and the rhythm score depends on the amount of times that the subject touches the desired body part within one second after receiving the instruction. These two scores signify different psychometric measures necessary for measuring self-regulation, response inhibition, working memory, co-ordination and attention [28]. Thus, our system is designed to produce both those scores.

For the accuracy score part, the goal is to detect if the relevant activity is happening or not in a video. A potential approach is to manually segment and annotate these videos, to ensure only one activity happening per video, and then to recognize the activity in each trimmed video. It has limited real-world use, as it requires significant manual effort to produce the trimmed video. Furthermore, as this approach produces a video-level class label, as opposed to frame-by-frame labeling, it could not be used for computing rhythm score, which requires identifying the time when the hand touches the respective body part.

To address these limitations, in this paper we follow a problem formulation where the system output should predict both the body part that is touched, and the time during which

it is touched by the hand. The input to the system is a video segment, which typically has more than one action. For the training examples, the ground truth includes frame-level labels. The input video segment is provided automatically by the video capturing system, with no need for manual annotations, based on the time that each instruction is provided to the child. These instructions are pre-recorded, and the time that they are issued is known to the system.

Based on the above considerations, we formulate our problem as an frame-level supervised action segmentation problem, similar to formulations applied on the MPIICooking2 dataset[32] to understand fine-grained activities. We note that our problem could also possibly be tackled as an activity localization problem, but in our work so far we have not pursued that approach.

3.4.3 Annotation Scheme:

The annotation scheme we adopted in our system is illustrated in Fig. 3.1. Here we annotate explicitly those frames where the hand approaches the body part, then touches, and finally leaves the body part. As seen in the figure, a gesture is considered valid only if the hand has crossed the midline of the body and fulfilled the 3 sequential steps. The steps and the corresponding frames are highlighted in the figure. We first identify the time segment when the hand is about to touch the body part. We chose an approximate distance of a few inches around the designated body part. In the subsequent time segment the subject touches the body part, and in the third time segment the hand leaves the body part and gets to a distance of few inches from it. A video segment of video is assigned the appropriate label if it fulfills all these steps, else it is designated as background(BG). A strict protocol was followed as there are cases where a subject keeps the hand crossed and near the body part while using the other hand to touch the other side of the body part.

3.4.4 Dataset Statistics

Overall the dataset consists of 894 total videos recorded for 19 subjects, and has on an average 2.7 activities per video. The average length of a video sample is around 3.3 seconds, while the maximum and minimum length of the videos are around 3.6 and 3.1 seconds respectively. There are around 2500 activities in these videos having durations ranging from 0.03 seconds to 1.36 seconds. The dataset consists of 8 classes without background. The class-wise distribution of the dataset is as shown in the table 3.1. There are 8 classes indicating the combination of hand and body part. The 4 lettered class label is comprised of the first and second letter indicating the side: left(l) or right(l), the second

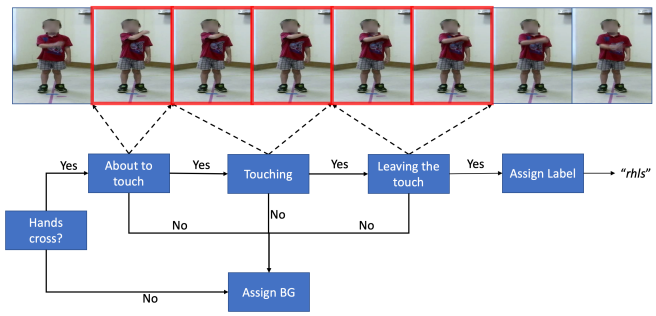


Figure 3.1: Annotation Scheme for the Cross Your Body System. Highlighted Red frames were given annotation as right hand left shoulder(rhls)

letter stands for hand(h) and the fourth letter stand for the body part: ear(e), shoulder(s), hip(h), knee(k).

Class label	Index	Min(sec)	Max(sec)	Mean(sec)	Sample Count
lhre	0	0.03	0.93	0.21	319
rhle	1	0.03	0.90	0.23	224
lhrs	2	0.03	1.17	0.20	388
rhls	3	0.03	0.87	0.20	285
lhrh	4	0.03	1.37	0.25	338
rhlh	5	0.03	1.00	0.24	257
lhrk	6	0.03	0.83	0.19	337
rhlk	7	0.03	0.87	0.21	284

Table 3.1: Class-wise duration distribution of the dataset.

3.4.5 Data Properties

The recorded data has several attractive properties that distinguish it from the existing datasets. *High Quality:* All the videos were recorded in Full HD resolution and were recorded in indoor conditions that ensured good illumination quality.

Richness and Diversity: The only practice that the subjects received was showing them once a video illustrating how to touch each body part. There were no instructions given to the children during the recording phase to improve their gestures or motion patterns. This resulted in a very realistic dataset that has many unique and novel motion patterns, as can be seen in figure 3.2. The dataset has high intra-class variation of and also has occlusions during performance of these activities. Also the diversity in the speed at which a subject touches a body part varies drastically, resulting into cases where the touch of a body part spans just a few frames.

Figure 3.2: Examples of action instances in Cross Your Body Data. The left part shows instances belonging to categories within the set of touching-shoulder, from top to bottom "right hand left shoulder", "left hand right shoulder", "right hand left shoulder", illustrating the level of variations that the touching-shoulder action can have. On the right the first 2 samples from top to bottom illustrate examples of touching-shoulder and touching-ear instances in which one hand occludes another, while the third sample illustrates occlusion by head while touching the shoulder.

Fine-grained action differences: Since there are classes like touch ear v/s touch shoulder, this dataset is unique from the point of view of inter-class variations, as there are samples belonging to different classes where the body pose looks very similar. Furthermore, since the resolution of the hands is relatively small, hand detection and tracking is a challenge, thus posing a unique use-case for activity detection and recognition algorithms.

3.5 System Definition

The goal of the system is to recognize and score the fine-grained actions in this dataset. Given the scoring requirements, the system is essentially an action segmentation system. Note that this dataset can also be used to evaluate activity localization algorithms, but we have so far not pursued that approach. Action segmentation involves frame-level predictions of an untrimmed video that may contain one or more activities.

We systematically evaluated 3 action segmentation methods based on Temporal Convolution Networks(TCN), namely MSTCN++[33], ASRF[34] and DTGRM[35] on the data and also included one pose based activity recognition system ST-GCN[36] to understand the significance of pose. Training protocols follow the original paper unless stated otherwise.

For action segmentation, the setup involves a training set of N videos, where each video is composed of frame-wise feature representations $x_{1:T} = (x_1, ..., x_T)$, where T is the length of the video. Using these features the system outputs the predicted action class

likelihoods $y_{1:T} = (y_1, ..., y_T)$, where $y_t \in R^C$ and C is the number of classes. During test time, given only a video, the goal is to predict $y_{1:T}$.

All of the experiments were performed using a user-independent 6-fold cross validation system where it was ensured that, for each split, there is no training video of any person appearing in the test set. The remaining section explains the feature extraction and the analysis on the performance of these methods.

3.5.1 Feature Extraction

The actions in the video are highly dependent not only on the motion patterns but also on the appearance information. Due to the fine-grained nature of the activities like touching ear v/s touching shoulder, missing on the latter information will lead to incorrect prediction. We used the 2 modalities of data, mainly optical flow and RGB frames to produce intermediate frame-representation using I3D network[37] pretrained on Kinetics dataset. We have chosen a temporal window of 16 frames to compute the I3D features. The I3D features extracted from each modality is concatenated together to produce a feature representation $\mathbf{x}_i \in \mathbb{R}^{2048 \times T_i}$, where T_i is the length of the video i

3.5.2 Action Segmentation

We chose Temporal Convolution Networks(TCN)-based modelling systems, because TCNs have a large receptive field and work on multiple temporal scales, and thus they are capable of capturing long-range dependencies between the video frames. The reason we chose this multi-scale option is because the instruction given usually follows a theme. For example, if the instruction is "Cross your body touch your Ears, Ears, Ears", the subject is expected to touch ears three times (each time with the opposite hand than the previous time). Consequently, frame-level predictions become easier if the network understands that the actions happening in the video are related to ears, and that hands alternate.

MSTCN++ is an improvement over MSTCN where the system generates frame level pedictions using a dual dilated layer that combines small and large receptive field in contrast to MSTCN[38]. While MSTCN++ has the ability to look at multiple temporal fields, it still lacks the ability for efficient temporal reasoning. This drawback was resolved in DTGRM where they used Graph Convolution Networks(GCN) and model temporal relations in videos. While models like MSTCN++ and MSTCN use smoothing loss to avoid over-segmentation errors, this method introduced an auxilliary self-supervised task to encourage the model to find correct and in-correct temporal relations in videos.DTGRM

and MSTCN++ both work predict frames directly and there is no concept of detecting action boundaries. Since in our dataset and task it is necessary to understand when an action starts and ends, and also since the motion is so fine-grained, it is important to accurately detect action boundaries. To resolve this problem, we also employed another method, ASRF, which alleviates over-segmentation errors by detecting action boundaries. For analysis of the importance of the method based on the modality of the data, we trained each method on 3 different modalities: I3D features extracted on RGB frames, I3D features extracted on flow frames, and a third modality consisting of concatenating the I3D features of the first two modalities.

Evaluation metrics: For evaluation, the metrics we employ are framewise accuracy(Acc), framewise accuracy without background(Acc-bg), segmental edit distance (Edit), and segmental F1 score measured at overlapping thresholds of 10%, 25% and 50% denoted by F1-10, F1-25 and F1-50 respectively. The overlapping threshold is based on the metric of Intersection over Union (IoU) ratio. We added the framewise accuracy without background as a metric since a major portion of the frames in the dataset is background.

We did 2 forms of analysis: event-based analysis, where models were trained using all 9 classes including background. The second analysis was done based on a subset of the labels. More specifically we relabeled "left hand right ear" and "right hand left ear" to just ear and similarly for shoulder. This resulted in only 3 classes: ear, shoulder and background.

Table 3.2 illustrates the performance of all the 3 segmentation models for all the classes. It can be observed that ASRF shows the best performance as compared to MSTCN++ and DTGRM, as it is able to predict action boundaries. Note that the metric of *Acc-BG* is more important than *Acc* as the dataset has a lot of background frames. More importantly, the flow modality produced better results as compared to RGB or even concatenated I3D features, and this is further discussed in the analysis subsection. Other fine-grained activity datasets like [39] have shown similar observations where they illustrate RGB values contribute less due to subtle differences between classes as compared to coarse-grained classes.

We have also investigated how data modality plays a role on capturing temporal dynamics in actions that are very fine. For this, we chose a subset of classes like touching ears and shoulders because they are spatially separated by a very low margin as compared to touching hip and knee. Table 3.3 illustrates the performance of all the 3 segmentation models for this subset of classes. It is evident that flow plays an important role in understand these motion patterns.

Method	Modality	Acc	Acc-BG	Edit	F1-10	F1-25	F1-50
MSTCN++	RGB	79.59	28.43	67.94	69.53	67.13	55.07
	Flow	82.73	36.10	71.15	74.13	71.81	60.13
	Both	82.11	36.24	71.68	74.71	72.15	59.90
DTGRM	RGB	81.10	26.75	64.83	68.59	66.09	54.75
	Flow	83.55	35.76	71.91	75.35	72.52	61.05
	Both	**83.67**	35.68	70.50	75.07	72.50	60.74
ASRF	RGB	63.34	36.42	55.73	59.90	54.64	42.79
	Flow	80.45	**48.66**	**73.94**	**78.04**	**75.56**	**63.66**
	Both	80.98	44.03	73.45	77.10	74.81	63.27

Table 3.2: Performance of Action Segmentation models for All Classes.

Method	Modality	Acc	Acc-BG	Edit	F1-10	F1-25	F1-50
MSTCN++	RGB	82.41	33.95	71.92	73.51	70.38	59.34
	Flow	86.00	51.36	**80.44**	82.45	80.03	69.76
	Both	85.87	48.08	78.87	81.76	79.90	69.00
DTGRM	RGB	84.00	34.22	72.39	75.65	73.00	61.79
	Flow	**86.69**	50.80	79.09	82.62	80.85	70.23
	Both	86.09	43.45	74.76	80.08	77.43	66.66
ASRF	RGB	83.28	44.22	75.23	77.34	75.74	65.88
	Flow	85.02	**56.55**	79.95	**82.90**	**80.86**	**70.50**
	Both	84.63	49.12	77.79	80.64	78.93	68.69

Table 3.3: Performance of Action Segmentation models.(Set Level: Ear and Shoulder)

3.5.3 Analysis

3.5.3.1 Comparison with human scores

Using the segmentation results and the times when the instruction was made by accessing the video, the relevant accuracy and rhythm scores for each subject was produced by the system for all the 5 tasks and all subjects. The Bland-Altman plots(Fig. 3.3 and 3.4) shows the comparison of the system scores and human scores for the activities performed by the kids. The Y-axis indicates the difference of the scores generated by machines and humans and the X-axis indicates the mean of the scores using both methods. Every point in the scatter plot indicates the measurement for a user which may overlap. For each plot, The blue line indicates the mean of difference of human and machine scores (estimated bias) which is 3.04 for accuracy metric and 3.25 for rhythm. The red lines refers to interval (mean $\pm$ 1.96 $\times$ standard deviation) which signifies the limits of agreement between human and machine scores. For accuracy the limit of agreement is [6.55,0.47] and [6.66,0.16] for rhythm. Ideally the mean of differences should be closer to 0. This shows that there is

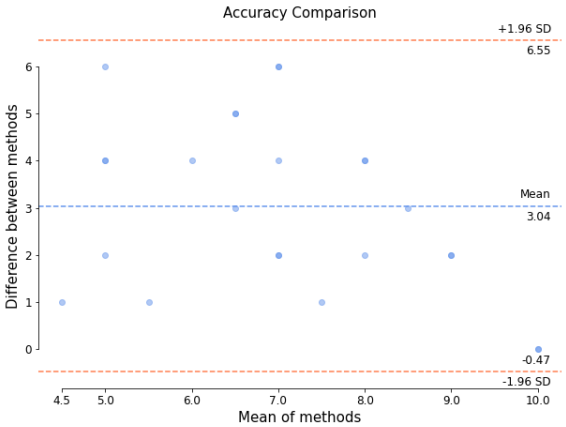


Figure 3.3: Accuracy Score comparison of Human and machine scores

a potential of improvement in reliably detecting the touches.

3.5.3.2 What cannot be handled by the current models

To illustrate the limitations of the current methods, we provide the segmentation results of some cases where the network failed.

1) Self correction: The network cannot always handle the scenario where the user self-corrects the touch. For example in Fig.3.5, the subject first intends to touch his shoulder. Then, in the middle of moving the hand towards the shoulder, he corrects himself and proceeds to touch the hip. Such cases are essential in this dataset and the target application, as the subject has to utilize their working memory to decide which body part to touch based on the instruction and type of the rule they are told to follow. The task was intentionally designed by the psychologist experts so that subjects can get easily confused and need to self-correct. The segmentation results show that the best system ASRF incorrectly predicted *lhrs* as it failed to understand that the touch did not happen.

2) Confusion between ear and shoulder: While the pose system clearly illustrated that the system cannot handle spatially fine-grained poses like touch ear v/s touch shoulder for some cases, this issue was echoed in action segmentation results as well, which used much more sophisticated I3D features.

3) Intense motion: Sometimes out of confusion and haste to complete the task, the subject performs touching of body parts at a high speed, and that makes it challenging to predict.

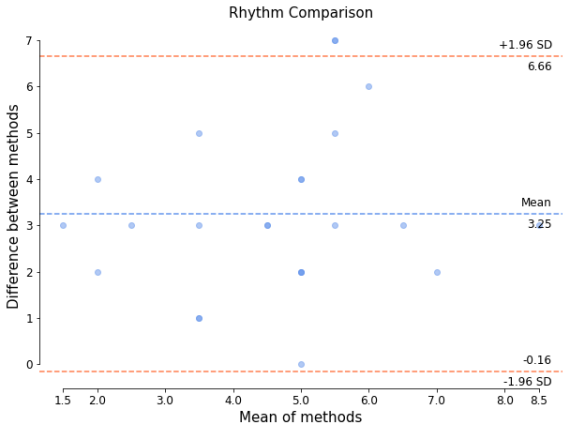


Figure 3.4: Rhythm Score Comparison of Human and machine scores

4) Occlusions: As the subjects perform action very quickly, this results into scenarios where the activities overlap and hand from the previous action occludes the other hand which is being used to perform the next action, causing the network difficulty to track.

3.5.4 Conclusion

In this paper, we introduce a system for Cross-Your-Body task focusing on cognitive assessment using kids as subjects. The system differs from existing works as there is a direct comparison between the scores provided by human experts and machines. The recorded data provides diverse activities which have high intra-class variability and low inter-class variability. It also includes many unique and realistic actions that involve uncoordinated motion patterns that vary in pace and has occlusions. We have empirically investigated significance of pose and I3D features and different data modalities by viewing it as an action segmentation problem. Many interesting findings show that the current state of the art systems find it difficult to recognize these activities. Our system demonstrates creation of 2 fundamental metrics required to measure several executive functions and shows promising potential for future research. This system can be used as a non-intrusive solution for cognitive assessment in kids where there is no need of an expert to manually score the cognitively demanding tasks.

Acknowledgements

This work was partially supported by National Science Foundation grants IIS 1565328.

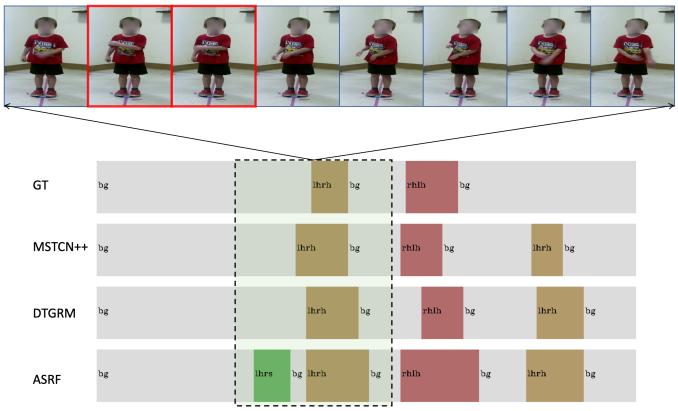


Figure 3.5: Segmentation Results for different methods. ASRF incorrectly predicts lhrs(highlighted section) when the subject hovered his hand around shoulder and then decided to touch right hip

Part 2:
Timestamp supervised action segmentation

4 Timestamp Supervised Action Segmentation Using Human Object Interaction

4.1 Abstract

This paper focuses on temporal action segmentation using timestamp supervision where only one frame is annotated for each action segment. The main idea and contribution is to use information from Human Object Interaction (HOI) to improve action segmentation accuracy. This information is obtained from an off-the-shelf pre-trained HOI detector, that requires no additional HOI-related annotations in our experimental datasets. Our approach generates pseudo-ground truth by expanding the annotated timestamps into intervals including neighboring frames where a human is continuously interacting with an object. This pseudo-ground truth allows the system to specifically exploit the spatio-temporal continuity of human interaction with an object to segment the video. Our experiments quantitatively show the advantages of leveraging HOI information, as our framework outperforms state-of-the-art methods on three challenging datasets with varying viewpoints, providing improvements of up to 10.9% in F1 score and up to 5.3% in frame-wise accuracy.

4.2 Introduction

The amount of video data available on the internet is growing at an ever-increasing rate. Analyzing these videos is important for diverse real-world applications in surveillance, video suggestions, sport analytics, etc. This potential has motivated the design of various approaches [40, 37, 41, 42, 43] for action recognition in trimmed video clips in recent

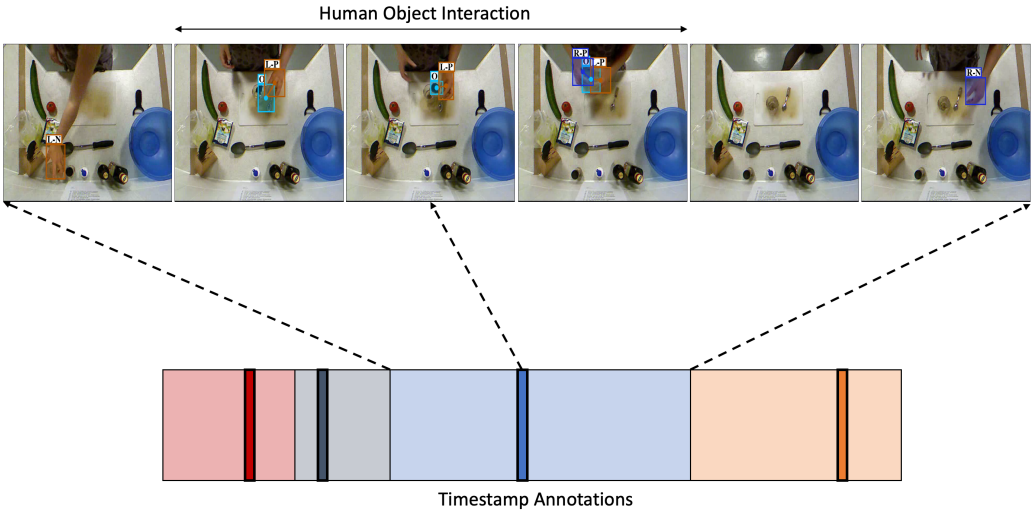


Figure 4.1: The continuity of human object interaction carries important information about the continuity of an action. The blue bounding boxes in the video indicate the spatial locations of objects that the human is interacting with. In timestamp supervision only one arbitrary frame per action segment is known (indicated by vertical bars in the segmented video), but the action label of that frame can be propagated to neighboring frames based on patterns of human-object interaction around that frame.

years. However, in the real world videos are usually untrimmed and contain several actions of varying lengths.

Action segmentation is the task of temporally segmenting untrimmed videos and producing an action label for every frame[44, 45, 46, 38]. Fully supervised action segmentation methods require as training data the start and end frame of each action in each training video. However, manually annotating these action boundaries is time-consuming and simply not scalable to large datasets.

To alleviate the manual annotation bottleneck, some action segmentation approaches[47, 48, 49, 50, 51, 52] utilize weaker supervision, in the form of an ordered sequence of actions present in the video, without specifying the start and end frames of each action. Similarly [53, 32, 54] use action sets to segment the video temporally. While these methods have significantly lighter annotation requirements, they attain much lower accuracy than their fully supervised counterparts. This gap in accuracy has led to an alternative type of supervision called time-stamp supervision[55] where, in addition to the ordered sequence of actions, the training data also contains a single frame number for each action, thus placing significant constraints on when each activity may be happening.

In this paper, we focus on timestamp supervision, given its promising combination of lighter annotation requirements and accuracy that is closer to that of fully-supervised methods. Within that context, we propose extracting and using human object interaction information to improve accuracy. Our approach extends the supervisory signal of single-frame timestamps to intervals around those timestamps, by identifying neighboring frames where human object interaction occurs continuously, and labeling such frames with the same action.

Figure 4.1 illustrates this idea using an example. In that figure, for the action of *add pepper* in a video, the human takes the container, adds the pepper and puts it back. Detecting the time interval of interaction between the human and the pepper container allows us to propagate the *add pepper* action label from the single frame included in the training data to all frames in that interval.

The main contributions of the paper are as follows:

- The key novelty is the idea of using HOI information to improve action segmentation accuracy. Furthermore, we show that in practice this idea does not require any extra training data for new action recognition datasets.

- We provide a specific architecture that serves as an example illustrating how to exploit HOI information in action segmentation under timestamp supervision. The proposed architecture demonstrates the feasibility and benefits of using HOI information in this setting.

- The proposed architecture outperforms state-of-the-art methods on action segmentation using timestamped supervision. We evaluated our system on three datasets 50salads[56], MPII Cooking 2[57], GTEA[58]. The system can be generalized to varying environments and viewpoints (egocentric and third person).

In principle, our idea of using HOI information requires additional, HOI-specific training data in order to train an HOI detector. In practice, we have used the same pre-trained off-the-shelf HOI detector in all our experimental datasets. Thus, these extra HOI-specific annotations can be treated as a one-off cost (that has already been paid if one uses an off-the-shelf HOI detector), as opposed to being an additional cost for each new action recognition dataset.

In the experimental results, our system provides across-the-board improved accuracy in all three datasets for all metrics, compared to the state-of-the-art timestamp-supervised

action segmentation methods. The source code and extensive documentation will be made public upon acceptance.

4.3 Related Work

4.3.1 Weakly Supervised Methods.

Weakly supervised methods for action segmentation have used diverse approaches such as connectionist temporal classification[47], energy-based learning[51] or Dynamic Time Warping[52]. Some methods are iterative, and alternate between generating pseudo-ground truth using the current model and refining the current model using the pseudo-ground truth [59, 48, 49, 60]. These methods suffer from relatively high inference time, and they cannot generate transcripts (i.e., action sequences) that have not seen during training, which makes such methods a poor fit for datasets where the number of possible action sequences is combinatorially large. Souri *et al.*[61] make the inference time faster by predicting the transcript alongside the frame-level predictions using mutual consistency. The features traditionally used are derived from either Improved Dense Trajectories (IDT) or I3D.

4.3.2 Timestamp Supervision.

The accuracy attained by weakly supervised action segmentation methods is significantly lower to that of fully supervised methods. Timestamp supervision has recently been explored as a way to bridge this accuracy gap, while still not requiring the same annotation burden as full supervision. Moltisanti *et al.*[55] trained a fine-grained acition classifier by employing a plateau function sampling distribution centered around temporal timestamp annotations. This method showed promising result on temporal action localization for trimmed videos. Later Ma *et al.*[62] mined action and background frames to extend the action localization system. Recently Li *et al.*[63] proposed a timestamp supervision method which uses the model predictions and the annotated timestamps to estimate action change. They also proposed a confidence loss that forces model confidence to monotonically decrease as the distance to timestamp increases. This system showed significantly improved results compared to weakly supervised methods trained using only action sequence information and no timestamps.

4.3.3 Human Object Interaction.

The task of human object interaction(HOI) detection is to localize a human and an object in their respective bounding boxes and then to specify the interaction between them, by outputting a tuple <human bounding box,object bounding box, object class, action class> given an image. This is an active research area[64, 65, 66, 67] and further literature on image based HOI can be found in state-of-the-at HOI papers[67]. In the video domain, Gupta *et al.*[68] formulated a bayesian approach that helps integrate various perceptual tasks involved in understanding human-object interactions. Also Kopula *et al.*[69] formulated the problem as a graph where the edges represented affordance and relation between human actions and objects and nodes represented objects. Also work from Nagarajan *et al.*[70] finds *interaction hotspots* on the objects and learns object affordances using the videos without manually annotated segmentations. These interaction hotspots are pixel-level segmentations that provide information of object affordance. Similarly environment affordance was utilized in applications involving action anticipation[71], and Xiao *et al.*exploits action/object relations for recognition in trimmed videos. Another method [72] on image-level HOI detection is designed to detect hands and objects when they are in contact. That system not only predicts the hand in contact with the object but also finds the bounding box of the object that is in contact with the hand. This system is technically related to [73] but instead of predicting triplets <human, verb, object>, they propose an alternative representation based on physical contact and interaction. The system is trained to recognize hands and active (touching) objects irrespective of object or activity class and thus can be generalized to other domains. However these approaches work on single images or trimmed videos, and no prior work has used HOI for action segmentation.

4.3.4 The Proposed Method in the Context of Related Methods.

With respect to the action segmentation methods discussed above, our method falls under timestamp supervision. The key feature differentiating our method from existing action segmentation methods is the use of information from human object interaction. Our method integrates HOI information within the timestamp supervised action segmentation framework of Li *et al.*[63], and the experiments show that using HOI information leads to across-the-board improved accuracy compared to the original results of [63].

The proposed method uses an HOI detection module as a black box, so any HOI method can be plugged in. Our implementation uses the off-the-shelf pre-trained system described in [72]. Consequently, our method can be applied to novel action recognition

datasets without needing any additional HOI annotations for those datasets.

4.4 Temporal Action Segmentation

Given a sequence of video frames $X = [x_1, ..., x_T]$ where T is the length of the video, the goal in temporal action segmentation is to predict action class label for each frame $a_{1:T} = [a_1, ..., a_T]$. In Section 4.4.1 we explain the problem formulation for action segmentation using timestamp supervision. In Section 4.4.2 we describe the proposed framework for learning from timestamp supervision using Human Object Interaction. Then we provide the details of loss function in Section 4.4.3.

4.4.1 Timestamp Supervision

In a fully supervised setup, each training video $X = [x_1, ..., x_T]$ is accompanied by frame-wise labels $[a_1, ..., a_T]$. However for timestamp supervision, the model is only provided with a single frame annotation per action segment during training. For a training video X containing T frames and N action segments, where $N << T$, labels $A_{TS} = [a_{t_1}, ..., a_{t_N}]$ specify one frame for each of the N segments. It is reported in [74] that it is 6 times faster to annotate a single frame per action than to annotate the start and end frames of each action.

4.4.2 Action Segmentation and HOI

Compared to other weaker forms of supervision such as transcripts (i.e., sequences of actions), timestamps provide not only the action class label but also a concrete temporal location when the activity is happening. This information allows us to explore and exploit patterns around that time frame. Commonly used datasets such as Breakfast[75], 50salads[56], MPIICooking2[57], GTEA[58], all display a human performing activities that involve interacting with objects. If we detect an interval of continuous human object interaction around a specific timestamp, we can assume that all frames in that interval belong to the same action as the timestamped frame. This approach creates HOI-influenced pseudo-groundtruth that enhances any other available real or pseudo-ground truth.

Many HOI detectors predict the action verb and spatial location of the interaction. There may be benefits to using the action verb information, but that may also require HOI training data more related to the specific action recognition dataset. To keep training requirements minimal, our current method does not use any action verb labels, and therefore does not require the HOI module to produce such labels.

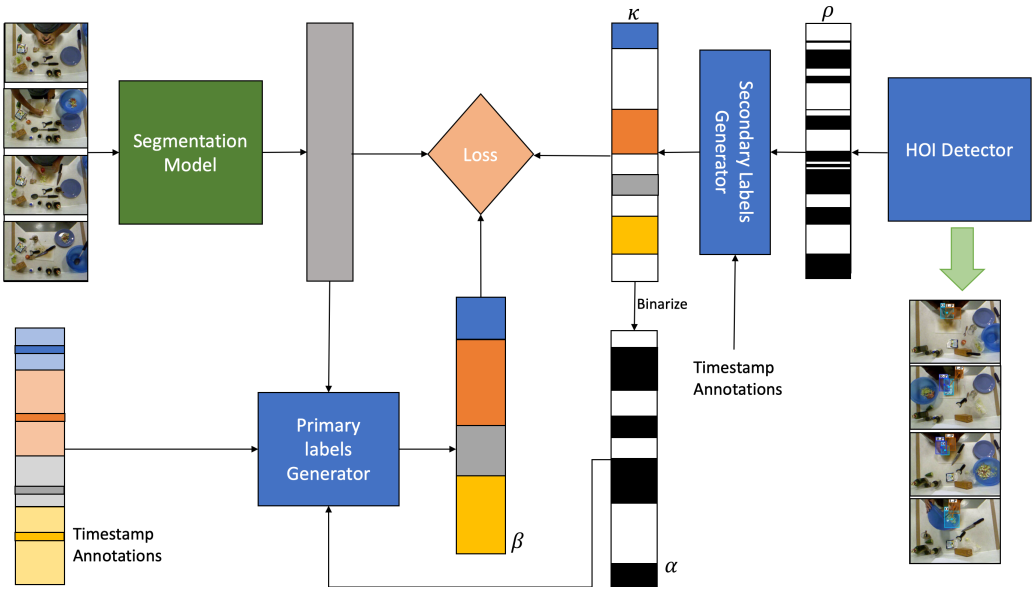


Figure 4.2: The proposed training framework. The secondary labels generator creates new pseudo ground-truth, κ using the HOI detections ρ and existing timestamp annotations. The binarized pseudo ground-truth(α) also provides new supervisory signal to the primary label generator for generating frame-wise labels β.

In our implementation, we use the off-the-shelf pre-trained HOI detector of Shan *et al.* [72]. Given an image, the model predicts hand sides and contact states either with the hands or surrounding objects. Hand side values are *left* or *right*, and hand state is represented as a 2D one-hot vector. There are five contact states: *none*, *self*, *other*, *portable*, and *non-portable*. The contact state is represented as a 5D one-hot vector. Alongside these categorical outputs, the model also produces bounding boxes around the hands and the interacting objects. In our method, we considered only those frames which had an interaction with a portable object. So, every frame with a detected contact state of portable is considered as a valid HOI frame, and the object bounding box b_t is stored. Here $t \in [1,T]$ and T is the length of the video.

4.4.2.1 HOI-Influenced Pseudo-Ground Truth

In the architecture diagram on Figure 4.2, the secondary label generator uses HOI information to generate pseudo-ground truth action labels. In this subsection we describe how the secondary label generator works.

The inputs are a video X, single-frame timestamp annotations $A_{TS} = [a_{t_1}, ..., a_{t_N}]$, and

a sequence of frame-level HOI predictions ρ. The output is pseudo-ground truth κ. As shown in Figure 4.3, we start with a window of τ frames around a given timestamp frame t_i. We denote by b_{anchor} the mean center location of the detected object bounding boxes within that window of τ frames. Point b_{anchor} provides an approximate location of the human object interaction around timestamp t_i. Neighboring frames will be labeled with the same action if the location of the detected human object interaction in those frames stays close to b_{anchor}.

Frame-wise labels κ are initialized to ground-truth single-frame timestamp action labels a_{t_i} for a video. Then, for each anchor location t_i, the algorithm considers adjacent intervals forward and backward in time, with a hop of w frames at a time, to decide whether to propagate label a_{t_i} to each of those intervals. We denote by $b_{i,j}$ the mean location of the object bounding box in frames $x_i, x_{i+1}, \ldots, x_j$. We denote by $\delta_{i,j}$ the distance between locations b_{anchor} and $b_{i,j}$. Given this notation, for a hop index h starting from 0 which increments by 1, $h \in \mathbb{R}$ and spatial threshold σ in pixels, the forward expansion of timestamp action a_{t_i} proceeds as follows:

$$\kappa_{[t_i+hw,t_i+(h+1)w]} = a_{t_i}, \text{ if } \delta_{[t_i+hw,t_i+(h+1)w]} < \sigma \tag{4.1}$$

The forward search terminates if $\delta_{[t_i+hw,t_i+(h+1)w]}$ for a hop h is greater than σ, if no valid HOI frames have been detected in hop h, or if the time search range reaches the end of the video.

Similarly the backward expansion of timestamp action a_{t_i} is as follows:

$$\kappa_{[t_i-hw,t_i-(h+1)w]} = a_{t_i}, \text{ if } \delta_{[t_i-hw,t_i-(h+1)w]} < \sigma \tag{4.2}$$

Once the forward and backward expansion of action timestamp a_{t_i} terminate, the next timestamp $a_{t_{i+1}}$ is considered for forward and backward expansion following the same logic.

4.4.2.2 Fine-tuning Action Changes

In the architecture diagram on Figure 4.2, the primary label generator, given a video X and timestamp annotations $A_{TS} = [a_{t_1}, ..., a_{t_N}]$, generates frame-wise labels $\hat{A} = [\hat{a_1}, ..., \hat{a_T}]$ such that $\hat{a}_{t_i} = a_{t_i}$ for $i \in [1, N]$ where N is the number of segments. In this subsection we describe the operation of the primary label generator.

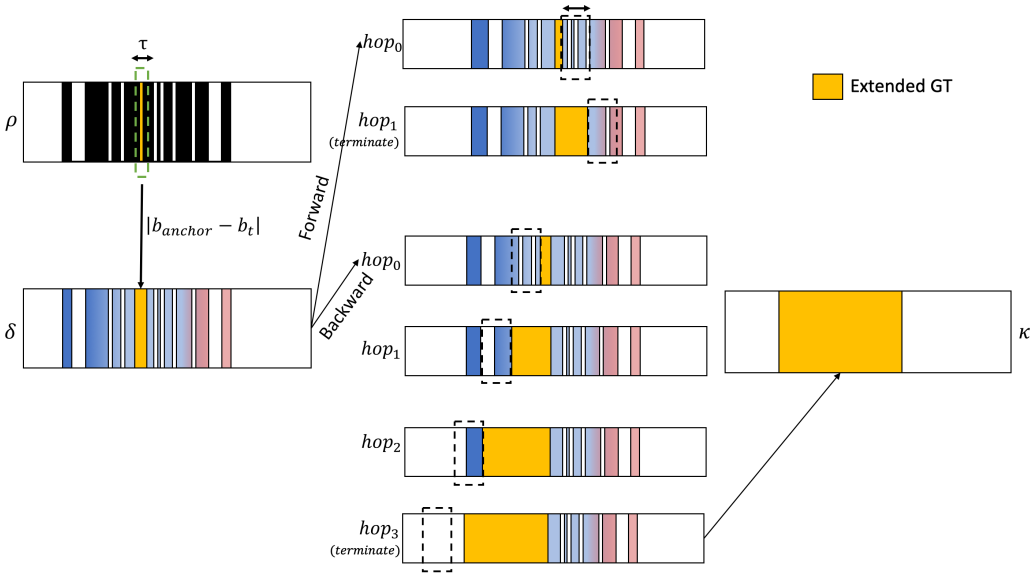


Figure 4.3: The proposed pseudo-ground truth generation method for a given action segment in a video. Timestamps are indicated in yellow. The black section in ρ indicates the frames where HOI was detected. After subtracting b_{anchor} from the bounding boxes of the neighbouring frames, the color spectrum in δ indicates magnitude difference from blue(low) to red(high) . hop_h indicates the progression of search window in forward and backward direction. Final pseudo ground-truth is indicated by the block κ.

Our formulation for this module builds on the method of [63], which trains a TCN model M for action segmentation. That TCN model is referred to as "segmentation model" in Fig. 4.2. To generate frame-wise labels, the method of [63] estimates the time t_{b_i} of action change between two consecutive timestamps t_i and t_{i+1}, as follows:

$$t_{b_i} = \underset{\hat{t}}{\operatorname{argmin}} \sum_{t=t_i}^{\hat{t}} d(h_t, c_i) + \sum_{t=\hat{t}+1}^{t_{i+1}} d(h_t, c_{i+1}) \tag{4.3}$$

$s.t.$

$$c_i = \frac{1}{\hat{t} - t_i + 1} \sum_{t=t_i}^{\hat{t}} h_t, \tag{4.4}$$

$$c_{i+1} = \frac{1}{t_{i+1} - \hat{t}} \sum_{t=\hat{t}+1}^{t_{i+1}} h_t \tag{4.5}$$

In the above, $d(.,.)$ signifies the Euclidean distance and h_t is the output of the penultimate layer of the TCN at time t. Intuitively, the algorithm divides the the frames between timestamps t_i and t_{i+1} into two clusters by finding the location t_{b_i} such that the average distance between the frame outputs and cluster centers is minimized.

In [63], this approach is implemented using a forward-backward algorithm. In the forward direction, frames from the last computed boundary $t_{b_{i-1}}$ to the timestamp t_i are assigned action label a_{t_i}, and these frames are used in estimating the next action boundary $t_{b_i,FW}$. For the backward direction, boundary estimate $t_{b_{i+1}}$, is used to predict the previous boundary $t_{b_i,BW}$. The average of the 2 estimates is used to find the final estimate t_{b_i}. As initial conditions, $t_{b_0} = 1$ and $t_{b_N} = T$, where T is the number of frames.

$$t_{b_{i,FW}} = \underset{\hat{t}}{\operatorname{argmin}} \sum_{t=t_{b_{i-1}}}^{\hat{t}} d(h_t, c_i) + \sum_{t=\hat{t}+1}^{t_{i+1}} d(h_t, c_{i+1}) \tag{4.6}$$

$$t_{b_{i,BW}} = \underset{\hat{t}}{\operatorname{argmin}} \sum_{t=t_i}^{\hat{t}} d(h_t, c_i) + \sum_{t=\hat{t}+1}^{t_{b_{i+1}}} d(h_t, c_{i+1}) \tag{4.7}$$

$$p = \frac{t_{b_{i,FW}} + t_{b_{i,BW}}}{2} \tag{4.8}$$

In [63], the value of p from Eq. 4.8 is used as the estimate for t_{b_i}. This is where our method diverges, and uses human-object interaction information to improve upon this estimate. Our modification is formulated as follows:

$$t_{b_i} = \begin{cases} p, & \alpha_p = 0 \\ f(p, G), & \alpha_p = 1 \end{cases} \tag{4.9}$$

$$f(p,G) = \begin{cases} min(G), & G \neq \emptyset \\ p, & G = \emptyset \end{cases} \tag{4.10}$$

$$G = \{t | t \in [t_{b_{i,FW}}, t_{b_{i,BW}}], \alpha_t = 0\} \tag{4.11}$$

Here $\alpha_t \in \{0,1\}$, for $t \in [0,T]$, indicates the interaction label obtained by binarizing the pseudo ground-truths κ at time t. Value 1 signifies interaction and 0 as no interaction. Figure 4.2 illustrates the binarized results α where the black segments indicate interaction and white segments indicate no interaction. Thus, we improve upon the architecture by adding a constraint that the detected boundary t_{b_i} is invalid if there is an ongoing human object interaction at that time. The boundary is re-adjusted to a temporal location where there is no interaction. During training, the final estimate t_{b_i} is estimated by the interaction label α_p. If interaction exists at time p then a subset of interaction values $\alpha_{[t_{b_i,FW}, t_{b_i,BW}]}$ is used to find a new action boundary. In the subset, the first time frame when there is no interaction is assigned as the new t_{b_i}. If there is interaction happening in all the frames in $\alpha_{[t_{b_i,FW}, t_{b_i,BW}]}$, then $t_{b_i} = p$.

4.4.3 Loss Function

We use the already successful combination of classification loss and smoothing loss used in traditional action segmentation techniques [38, 34, 76] and the novel confidence loss [63].

Classification Loss

For classification loss, we employed a cross entropy loss that computes the loss between the prediction action probabilities and the generated target labels as well as the generated pseudo ground-truths using HOI.

$$\mathscr{L}_{cls} = \frac{1}{T} \sum_t -log(\tilde{y}_{t,\hat{a}}), \tag{4.12}$$

Here $\tilde{y}_{t,\hat{a}}$ is the predicted probability from the model for target action label $\hat{a}$ at time t.

Smoothing Loss

To penalize for local inconsistencies in the the predicted action class probabilities we adopted the truncated mean square error as a smoothing loss[38]. This loss encourages the network to avoid over-segmentation errors

$$\mathscr{L}_{T-MSE} = \frac{1}{TC}\sum_{t,a}\tilde{\Delta}^2_{t,a}, \tag{4.13}$$

$$\tilde{\Delta}_{t,a} = \begin{cases} \Delta_{t,a}, & \Delta_{t,a} \leq \tau \\ \tau, & otherwise \end{cases} \tag{4.14}$$

$$\Delta_{t,a} = |log\tilde{y}_{t,a} - log\tilde{y}_{t-1,a}| \tag{4.15}$$

Where C is the number of action classes, $\tilde{y}_{t,a}$ is the action a probability at time t.

Confidence Loss

The confidence loss[63] enforces monotonicity on the model confidence and is defined as follows:

$$\mathscr{L}_{conf} = \frac{1}{T'}\sum_{a_{t_i}\in A_{TS}}(\sum_{t=t_{i-1}}^{t_{i+1}}\delta_{a_{t_i},t}), \tag{4.16}$$

$$\delta_{a_{t_i},t} = \begin{cases} max(0, log\tilde{y}_{t,a_{t_i}} - log\tilde{y}_{t-1,a_{t_i}}), if\, t \geq t_i \\ max(0, log\tilde{y}_{t-1,a_{t_i}} - log\tilde{y}_{t,a_{t_i}}), if\, t < t_i \end{cases} \tag{4.17}$$

Using this loss, the low confident regions which are surrounded by higher probability regions are encouraged to produce higher probabilities. This loss also penalizes outlier frames carrying high probabilities that are far from the annotated timestamp and that are not surrounded by high confidence regions.

The final loss of the action segmentation model is:

$$\mathscr{L}_{total} = \mathscr{L}_{cls} + \alpha \mathscr{L}_{T-MSE} + \beta \mathscr{L}_{conf} \tag{4.18}$$

Here α and β are the hyper-parameters that guide the contribution of each loss.

4.5 Experiments

In this section, we compare our method with the current state of the art systems for action segmentation using timestamp supervision. We also show the contribution of each component quantitavely and qualitatively.

4.5.1 Datasets

In our experiments, we have used three public datasets commonly used for evaluating action segmentation methods: 50salads[56], MPII Cooking 2[57], and GTEA[58]. We note that each dataset contains both video-level class labels, that describe the activity of an entire video, and segment-level class labels, that describe the fine-grained action that takes place at each segment. In these experiments, both for our method and for the competitors, we do not take video-level class labels into account. We are only concerned with labeling each frame with the correct fine-grained action label.

The **50Salads** dataset contains 50 videos and 17 fine-grained action classes. Each video on average contains 20 fine-grained action instances and is 6.4 minutes long. The videos display human subjects preparing different types of salads. There are 25 video-level class labels (different salads) overall, and every actor prepares two different salads.

The **GTEA** dataset contains 28 egocentric videos and 11 fine-grained action classes. There are 7 different video-level classes such as "preparing tea" and "hot dog", performed by 4 subjects. Each video contains 20 fine-grained action instances on average.

The **MPII Cooking 2** contains 243 high quality videos, ranging in length from 1 minute to 40 minutes, and 67 fine-grained action classes. It includes 29 subjects who prepare 58 different dishes (video-level class labels) like "making pizza" or "preparing cucumber".

4.5.2 Evaluation Metrics

We use evaluation metrics commonly used in action segmentation tasks [38, 34, 76]: frame-wise accuracy (Acc), segmental edit distance (Edit) and segmental F1 score at over-

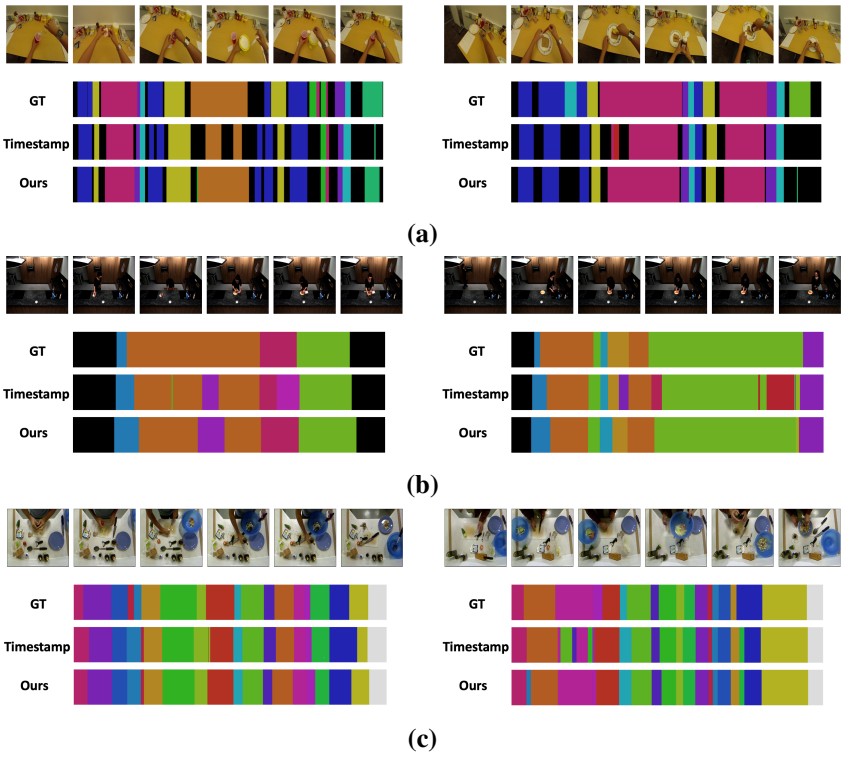


Figure 4.4: Qualitative results on (a) 50Salads, (b) MPII Cooking2 and (c) GTEA datasets. The current state of the art system (Timestamp) still suffers from over-segmentation in all datasets. On the contrary our approach gets better prediction by utilizing the frames where continuous human object interaction occurs.

lapping thresholds of 10%, 25% and 50%, denoted as F1@{10,25,50}. While frame-wise accuracy is the most commonly used metric in action segmentation research, it naturally places more importance on long-duration actions over shorter actions, and it lacks an explicit penalty for over-segmentation errors. Segmental edit score and F1 score penalize the over-segmentation errors and treat shorter and longer duration actions as equally important.

4.5.3 Implementation Details

For the action segmentation module of Fig. 4.2 we use the multi-stage temporal convolution network of Li *et al.*[63]. For HOI detection, if there are multiple objects detected, where the human is interacting with an object in each hand, the bounding boxes are merged to a bigger bounding box. We trained for 70 epochs using Adam optimizer. For the first 30 epochs the network was trained using only the annotated timestamps to minimize the impact of initialization. From epochs 30 to 50, the pseudo ground-truth κ generated using HOI (the output of the "secondary label generator" in Fig. 4.2) was used to train the network. After epoch 50, the generated labels created by identifying action change and HOI (the output of the "primary label generator" in Fig. 4.2) were used for training. The learning rate is 0.0005 and the batch size is 8. For the loss function, we used $\tau = 4$, $\alpha = 0.15$ and $\beta = 0.075$. We used the same I3D[37] features as in [38]. We trained all models using the same timestamp annotations as Li *et al.*[63], for fair comparison with other methods. For all 3 datasets the optimum σ and τ values were obtained based on the cross-validation performance, whereas the α and β values that we used were the same ones as in [63]. The value of σ used for each dataset, and further implementation details, can be obtained in the supplementary material.

4.5.4 Results

4.5.4.1 Comparison with the State of the Art System

In Table 4.1 we compare our method with the current state-of-the-art method of Li *et al.*[63] for action segmentation using timestamp supervision. Compared to [63], our approach consistently attains higher accuracy in all 3 datasets in all metrics. For GTEA, the F1 score at 50% overlapping threshold improves by 10.9%. The frame-wise accuracy improves by 5.3% when compared to [63] and is now 92.5% of the fully supervised approach. For 50Salads, the F1 score at 50% overlapping threshold improves by 5.9% and the frame-wise accuracy improves by 0.4% when compared to [63] and is now 97.45%

	F1 @ {10, 25, 50}			Edit	Acc
50Salads					
Timestamp[63]	73.9	70.9	60.1	66.8	75.6
Ours	**77.3**	**75.2**	**63.6**	**69.8**	**75.8**
Full Supervision	70.8	67.7	58.6	63.8	77.8
GTEA					
Timestamp[63]	78.9	73.0	55.4	72.3	66.4
Ours	**82.1**	**78.7**	**63.0**	**74.8**	**70.4**
Full Supervision	85.1	82.7	69.6	79.6	76.1
MPII Cooking2					
Timestamp[63]	42.7	38.7	28.7	41.1	50.1
Ours	**44.9**	**40.6**	**28.8**	**43.5**	**51.3**
Full Supervision	45.5	42.1	32.5	43.2	54.0

Table 4.1: Comparison between our method and current state of the art for timestamp action segmentation on the three datasets.

Supervision	Method	F1@{10,25,50}			Edit	Acc
Full	MSTCN++[33]	80.7	78.5	70.1	74.3	83.7
	BCN[76]	82.3	81.3	74.0	74.3	84.4
	ASRF[34]	84.9	83.5	77.3	79.3	84.5
Timestamps	Seg model + plateau [55]	71.2	68.2	56.1	62.6	73.9
	Timestamp[63]	73.9	70.9	60.1	66.8	75.6
	Ours	**77.3**	**75.2**	**63.6**	**69.8**	**75.8**

Table 4.2: Results with different levels of supervision on 50Salads.

of the fully supervised approach. For MPII cooking2, the F1 score at 25% overlapping threshold improves by 4.5% and the frame-wise accuracy is improved by 2.3% when compared to [63] and is now 95.1% of the fully supervised approach.

Tables 4.2-4.3 compare the performance of the system with state-of-the-art fully supervised methods as well as with other timestamp supervision methods. For fair comparison, all timestamp supervision methods use the same timestamp annotations. These tables illustrate that our method makes a significant step towards closing the accuracy gap between timestamp supervision and fully supervised methods.

Supervision	Method	F1@{10,25,50}			Edit	Acc
Full	MSTCN++[33]	88.8	85.7	76.0	83.5	80.1
	BCN[76]	88.5	87.1	77.3	84.4	79.8
	ASRF[34]	89.4	87.8	79.8	83.7	77.3
Timestamps	Seg model + plateau[55]	74.8	68.0	43.6	72.3	52.9
	Timestamp[63]	78.9	73.0	55.4	72.3	66.4
	Ours	**82.1**	**78.7**	**63.0**	**74.8**	**70.4**

Table 4.3: Results with different levels of supervision on GTEA.

4.5.4.2 Impact of loss with HOI

Table 4.4 shows the benefits of using HOI information. We show results using the original loss function of [63], and results obtained by incorporating two changes proposed in this paper: "pg" denotes the pseudo-ground truth generated using HOI, as described in Section 4.4.2.1. By "ft" we denote detecting action boundaries using the proposed "fine-tuning" equations 4.9-4.11 of Section 4.4.2.2, whereas versions not marked with "ft" detect action boundaries as described in [63].

For 50Salads, the F1 score @50% overlap increased by 2.5% when compared to [63] when adding the "pg" component, and increased further by 0.6% when using the "ft" approach. The qualitative results showcase how our approach corrected some of the over-segmentation errors in [63]. Similar improvements were seen in GTEA, where the F1 score @50% increased by 2.7% by using just pseudo-ground truth and by 4.9% with fine-tuning action changes using HOI. Similar gains were seen in MPII Cooking 2. Overall, our system yields consistently higher accuracy in datasets that have varying viewpoints (egocentric and third person).

4.5.4.3 Impact of fine-tuning.

Table 4.5 illustrates the benefits of re-adjusting the action change boundaries using HOI information. The terms "loss", "pg" and "ft" have the same meanings that we defined in discussing Table 4.4. Table 4.5 shows that, for the GTEA dataset, our proposed improvements lead to higher accuracy in almost all metrics. There are only two entries in that table (out of a total of 10) where the proposed components do not improve accuracy, but in both those cases the drop is marginal (0.3%). In the other eight entries, our components lead to improvements ranging from 0.6% to 4.9%.

	F1@{10,25,50}			Edit	Acc
50Salads					
loss[63]	73.9	70.9	60.1	66.8	75.6
loss+pg	76.5	74.4	62.6	69.3	75.7
loss+pg+ft	**77.3**	**75.2**	**63.6**	**69.8**	**75.8**
GTEA					
loss[63]	78.9	73.0	55.4	72.3	66.4
loss+pg	79.9	75.5	58.1	74.2	68.2
loss+pg+ft	**82.1**	**78.7**	**63.0**	**74.8**	**70.4**
MPII Cooking2					
loss[63]	42.7	38.7	28.7	41.1	50.1
loss+pg	44.4	40.0	28.3	42.1	50.5
loss+pg+ft	**44.9**	**40.6**	**28.8**	**43.5**	**51.3**

Table 4.4: Contribution of the original loss from Li *et al.*[63](loss), new pseudo ground-truth(pg) generation and fine-tuning(ft) of the action change using HOI

	F1@{10,25,50}			Edit	Acc
GTEA					
loss[63]	78.9	73.0	55.4	72.3	66.4
loss+ft	78.6	74.5	57.6	72.0	67.9
Improvement	-0.3	1.5	2.2	-0.3	1.5
loss+pg	79.9	75.5	58.1	74.2	68.2
loss+pg+ft	82.1	78.7	63.0	74.8	70.4
Impovement	2.3	3.1	4.9	0.6	2.2

Table 4.5: Improvement in performance for GTEA using labels generated by adding constraint of HOI to detect acton change

4.6 Conclusion

The main novel idea in this paper is that information from human-object interaction can be used to improve action segmentation accuracy under timestamp supervision. Our model extends the single frame timestamp annotations using the frame level predictions of a human-object interaction detector. We track the object that the human is interacting with around the timestamp, and we use that information to generate pseudo-ground truth action labels. We improve the existing frame-level action class generator by adding a constraint that an action boundary cannot exist around frames where the human is continuously interacting with the object. Results on three commonly used public datasets show that the key idea of using HOI information can indeed improve action segmentation accuracy noticeably. Our proposed architecture has outperformed the current state of the art, further closing the gap with models trained using full supervision.

4.7 Implementation Details

In this section we provide the implementation details for our model using the three datasets: 50salads[56], MPII Cooking 2[57], GTEA[58]. For all the three datasets, we train the model in 3 steps:

1. Until epoch 30 we use the single frame timestamps.

2. From epoch 31 to 50 we use the pseudo ground-truths generated from the timestamps and HOI.

3. From epoch 51 to 70 we train using the fine-tuned action boundaries using HOI.

Regarding Tables 4 and 5 of the main paper, which show variations of our method for ablation studies, this is how these variations correspond to the above steps:

- The "loss+pg" method of Table 4 corresponds to using only the first two steps above, and skipping the third step.

- The "loss+pg+ft" method corresponds to using all three steps as described above.

- The "loss+ft" method of Table 5 corresponds to using only steps 1 and 3, and skipping step 2. So, for the "loss+ft" version, from epoch 31 to 50 we train using the fine-tuned action boundaries using HOI, and we stop training when epoch 50 is done.

The best performance in all three datasets was dependent on the pseudo ground-truth generation parameters σ and τ as explained in the section titled *"HOI influenced Pseudo Groundruth"* of the main paper. For each dataset, the values for σ and τ were chosen automatically, using cross-validation. These are the values we used:

- For 50Salads, $\sigma = 30$ and $\tau = 30$.
- For GTEA, $\sigma = 10$ and $\tau = 75$.
- For MPII Cooking 2, $\sigma = 15$ and $\tau = 15$.

4.8 Impact of frame selection on performance

The HOI pseudo ground-truths are generated around the annotated frame-level timestamp. To check the system's sensitivity on the initialization of these frame-level annotations, we randomly selected timestamp frames for each action segment and created 10 unique sets of timestamp annotations and their respective HOI pseudo ground-truths for every video. We trained independent models with these newly generated annotations and performed this experiment for GTEA and 50Salads dataset and Table 4.6 illustrates the mean and standard deviation for each performance metric for these 10 models trained on unique timestamps and HOI pseudo ground-truths. It can be seen that the mean values for these performance metric are still better than the current state of the art despite random initialization and shows the system can still perform better even if timestamps are annotated randomly.

4.9 Importance of Pseudo-Ground Truths Using HOI

The information below describes alternative training strategies that we have evaluated. These strategies consist of different choices and orderings among the following modules:

- **Module a**: This is the module described in Section 3.2.1 of the main paper, which generates and uses pseudo-ground truth labels κ. As described earlier, in the implementation details section of this supplementary document, this module is used (in the normal version of our method) for training in epochs 31 to 50.

Dataset	F1@{10,25,50}			Edit	Acc
50Salads	76.6 ±0.6	74.2 ±0.6	63.1 ±0.8	69.5 ±0.5	76.2 ±0.3
GTEA	81.4 ±0.9	78.0 ±1.2	61.5 ±1.4	75.5 ±1.3	70.2 ±0.6

Table 4.6: Variation in performance using 10 unique combinations of randomly generated frame-level annotations. Number to the left of ± indicates the mean for 10 runs and to the right indicates standard deviation

- **Module b**: This is the "fine-tuning" module described in Section 3.2.2 of the main paper. As described earlier, in the implementation details section of this supplementary document, this module is used (in the normal version of our method) for training in epochs 51 to 70.

- **Module b'**: This is a replacement of the "fine-tuning" module described in Section 3.2.2 with the original boundary detection method used in [63]. In the system overview of Figure 2 of the main paper, this variant would correspond to cutting the link between binary labels α and the "primary labels generator".

Changing the order between module a and module b In the standard version of our method, as explained earlier, we train using module a in epochs 31-50, and we train using module b in epochs 51-70. We evaluated switching this order. This variation is denoted on Table 4.7 as "b then a". Essentially, in this variation we train using module b in epochs 31-50, and we train using module a in epochs 51-70. Table 4.7 shows the results of this variation. We see that using module a first and module b second gave better performance for all three datasets.

Replacing our fine-tuning module with the original boundary detection of .[63]

We also evaluated a variant where we use module b' (the original boundary detection module of [63]) instead of our finetuning module (module b). We tried both possible orderings in training (module a in epochs 31-50 followed by module b' in epochs 51-70, and the other way around).

Table 4.8 illustrates the performance differences of these variations. It can still be seen that training the network first from epoch 31 to 50 using pseudo ground-truths from HOI helps the network perform better for all three datasets. We can also see, by comparing the "a then b" results in Table 4.7 with the "a then b' " results of Table 4.8 that module b, which is one of our contributions, leads to better accuracy than module b' which is the corresponding component in [63].

Training Type	F1@{10,25,50}			Edit	Acc
50Salads					
a then b	**77.3**	**75.2**	**63.6**	**69.8**	**75.8**
b then a	73.2	70.1	58.2	64.7	73.8
GTEA					
a then b	**82.1**	**78.7**	**63.0**	**74.8**	**70.4**
b then a	73.6	66.6	49.4	68.8	61.4
MPII Cooking2					
a then b	**44.9**	**40.6**	**28.8**	**43.5**	**51.3**
b then a	35.0	30.1	19.6	30.8	39.5

Table 4.7: Variation in training using labels generated by pseudo ground-truths generated using HOI (a) and boundary detection using HOI (b)

Training Type	F1@{10,25,50}			Edit	Acc
50Salads					
a then b'	**76.5**	**74.4**	**62.6**	**69.3**	**75.7**
b' then a	71.8	69.0	57.8	64.9	73.7
GTEA					
a then b'	**79.9**	**75.5**	**58.1**	**74.2**	**68.2**
b' then a	73.4	65.7	45.7	70.7	60.3
MPII Cooking2					
a then b'	**44.4**	**40.0**	**28.3**	**42.1**	**50.5**
b' then a	35.9	31.4	20.6	32.6	39.8

Table 4.8: Variation in training using labels generated by boundary detection without using HOI (b') and pseudo ground-truths generated using HOI (a)

σ (pixels)	F1@{10,25,50}			Edit	Acc
10	80.5	77.2	59.7	74.9	68.9
20	**82.3**	**78.8**	60.5	**76.9**	69.9
25	81.4	78.0	**61.2**	73.8	**70.0**
30	80.3	75.2	59.3	74.4	69.0
35	81.0	77.4	58.6	73.2	68.4
40	77.6	72.2	56.0	72.5	67.2

Table 4.9: Performance Impact on varying Spatial threshold, σ in pixels with Temporal window $\tau = 30$ for GTEA dataset.

τ (frames)	F1@{10,25,50}			Edit	Acc
15	**80.8**	**76.4**	**60.6**	**75.3**	**69.4**
30	80.3	75.2	59.3	74.4	69.0
45	80.4	76.0	58.4	74.8	68.1
60	79.2	73.3	58.0	73.4	68.6
75	77.2	73.6	56.6	70.9	67.3
90	78.4	75.3	57.4	73.6	68.3

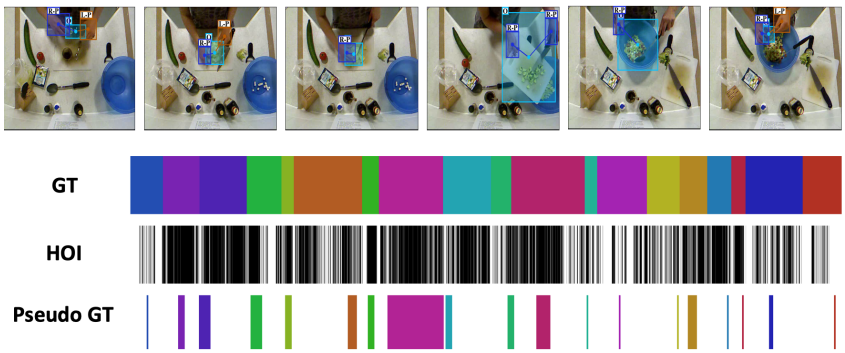
Table 4.10: Performance Impact on varying Temporal window, τ with Spatial threshold, $\sigma = 30$ for GTEA dataset

4.10 Impact of spatial and temporal thresholds

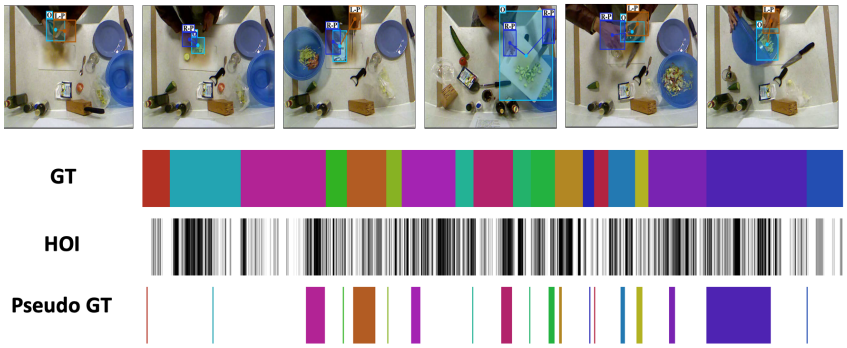
The pseudo ground-truth generated is controlled by 2 variables τ and σ. Variable τ controls the temporal window in which the algorithm finds the bounding box of interaction. Table 4.9 illustrates the impact of performance by keeping the temporal window constant at $\tau = 30$ frames and varying spatial threshold σ from 10 to 40 pixels. It can be seen that lower spatial thresholds of 10 or 20 pixels performed better as they ensure consideration of smaller movements during the interaction. Table 4.10 refers to the performance of varying the temporal window τ from 15 frames to 90 frames on GTEA dataset at a fixed spatial threshold σ of 30 pixels. It can be seen that the smaller window of 15 frames performs better as it will avoid overshoot of more frames to re-labelled incorrectly.

4.11 Accuracy of the generated pseudo ground-truth using HOI

We compared the quality of the pseudo ground-truths generated using HOI and single timestamp with the actual frame-wise ground-truth labels of the datasets. The metrics used were percentage count (%Count) of the frames where the algorithm labelled a frame



(a)



(b)

Figure 4.5: Human Object Interaction Detections (HOI) and the corresponding pseudo ground-truth generation for 50Salads Dataset

σ (pixels)	%Acc / %Count GTEA (τ=75)	%Acc / %Count 50Salads (τ=30)
10	**66.66/58.26**	95.50/8.79
20	54.28/72.06	94.94/15.21
25	48.73/78.88	94.80/18.05
30	42.15/82.89	**93.45/21.00**
35	39.29/86.43	93.04/24.55
40	36.36/89.23	91.83/28.18

Table 4.11: Variation frame-wise accuracy and count of frames using σ keeping τ constant for 50Salads and GTEA dataset. Highlighted values indicates the setup where the network gave best test performance.

with a valid action label. Note this analysis was not used to decide spatial and temporal thresholds. Thresholds were solely decided on the network's cross-validation performance. Using those valid labels, we measured how many of those frame-wise labels were accurate when compared to the ground-truth (%Acc).

Table 4.11 illustrates these values by using the variation of spatial threshold in pixels and keeping temporal window (τ) constant (75 frames for GTEA and 30 frames 50Salads). Increasing σ will enable the algorithm to track the HOI bounding box in neighboring frames at a coarser level, thus enabling the system include more frames in the same action, but the accuracy of these frames drops despite the increase in %Count.

Similarly Table 4.12 illustrates the same metrics using the variation of Temporal window τ and keeping spatial threshold σ constant (10 pixels for GTEA and 30 pixels for 50Salads). It can be seen that tracking the bounding boxes at longer lengths may cause the %Acc of the labels to reduce, but increase the %Count.

Thus, a good balance of accurate frames that last for longer duration is required and this will vary according to the dataset as some might have fine-grained actions, while others may long duration actions.

4.12 Impact of labels generated using HOI and action change

In Section 3.2.2 of the main paper, titled *Fine-tuning Action Changes*, we use the first frame of non-interaction frames in range $[t_{b_i,FW}, t_{b_i,BW}]$ to decide the boundary change location t_{b_i}. Table 4.13 illustrates performance of the network by picking up the last frame of non-interaction (last) as compared to the first frame of non-interaction (first) when the action boundary at t_{b_i} was at a location when HOI occured. It can be seen that using the first frame was the better strategy and gave better performance for all three datasets.

τ (frames)	%Acc/%Count GTEA (σ=10)	%Acc/%Count 50Salads (σ=30)
15	70.88/56.06	94.36/18.22
30	70.29/57.23	**93.45/21.00**
45	70.32/57.80	92.01/23.99
60	68.24/57.31	90.92/26.25
75	**66.66/58.26**	87.75/27.53
90	65.36/58.73	87.65/29.38

Table 4.12: Variation frame-wise accuracy and count of frames using using τ keeping σ constant for 50Salads and GTEA dataset. Highlighted values indicates the setup where the network gave best test performance.

4.13 Limitations

The proposed method makes several assumptions, and is limited by the extent to which those assumptions hold in a specific dataset. One assumption is that each video displays a single human performing activities involving interaction with objects. This assumption is relevant in many real-world applications, and it is true in commonly used datasets, such as the ones we have used in our experiments. At the same time, clearly there can be action recognition domains where this assumption does not apply. For example, this assumption would not apply for distinguishing between activities such as “walking” and “running”.

Also low resolution and dark condition videos, that appear for example in the Breakfast Dataset [75], will not benefit from this approach as the HOI detector fails to detect interactions. The pseudo ground-truth generation can be extended further by using off-the-shelf object detectors and tracking those bounding boxes from the extended interaction frames. Other temporal modelling systems like transformers can be used to improve the performance. The system currently utilizes the idea of interactions to generate the pseudo ground-truths. Future work can involve extraction of features inside the interaction bounding box which can provide more information to the network.

4.14 Improvement of Temporal Modeling using Transformers

To Improve the temporal modeling system, current literature provides evidence that transformers can be a viable solution instead of temporal convolution network[38]. To this end we replaced the TCN architecture with a transformer based[77] and Table 4.14 illustrates the improvement in performance for the existing method.

Type	F1@{10,25,0}			Edit	Acc
GTEA					
first	**82.1**	**78.7**	**63.0**	**74.8**	**70.4**
last	81.1	78.1	60.9	74.8	69.9
50Salads					
first	**77.3**	**75.2**	**63.6**	**69.8**	**75.8**
last	75.9	73.8	61.7	68.8	75.4
MPII Cooking 2					
first	**44.9**	**40.6**	**28.8**	**43.5**	**51.3**
last	45.4	40.4	27.7	42.3	49.7

Table 4.13: Variation in Fine tuning Boundary detection using action change detection and choosing first /vs last non HOI detected frame in range $[t_{b_i,FW}, t_{b_i,BW}]$

	F1@10	F1@25	F1@50	Edit	Acc
50 Salads					
Auth	73.9	70.9	60.1	66.8	75.6
Ours	77.3	75.2	63.6	69.8	75.8
Ours with transformer	**79.6**	**76.1**	**65.9**	**72.6**	**78.1**
GTEA					
Auth	78.9	73	55.4	72.3	66.4
Ours	82.1	78.7	63.0	74.8	70.4
Ours with transformer	**86.7**	**84.1**	**67.1**	**81.8**	**70.4**

Table 4.14: Performance Improvement using Transformers for temporal modelling

5 (3+1)ReC Dataset

5.1 Introduction

Current long duration activity understanding datasets suffer from several problems. Dataset like Breakfast[75] has good number of video-level labels and has third-person views. While the dataset has multiple views from which a person is being recorded, they suffer from bad video quality and has very low video resolution 320x480. Also the dataset suffers from large occlusions and also makes the algorithm difficult to track as there are many concerns regarding bad lightning conditions. While there are other datasets which resolve the bad resolution conditions like MPIICooking2[57], they are more tailored towards fine-grained activity recognition and they suffer from large class imbalance.

We created a new dataset Called 3+1ReC(Explain) which solves all these problems and also makes it more useful for activity recogntion focused towards multiple research areas. The dataset we created comprises of recording 30 unique actors, each perform 15 dishes in different Kitchen. We recorded high quality dataset having recording resolution of 1920x1080. The data is recording in Third Person and Egocentric view as well. There are 3 different third person view for the same dish along-with an egocentric view.

5.2 Data Acquisition and Protocol

The goal of the dataset is to facilitate high quality action understanding using the information of interactions around kitchen and not bias the system to perform a dish in the same manner. Every dish can be made by following sequential steps which are called as transcript. We ensured that every dish has unique set of sequences for the 30 subjects so that the dataset is not biased to certain sequences of actions. We have also ensured that the subjects are not instruted to perform actions in the same manner to avoid bias of spatio-temporal information while performing the cooking activities.

The subjects were recorded at multiple indoor locations and strict quality control was maintained as per what instructions need to be told to the participant before every recording session. Before recording every dish, we would verbally ensure if the subjects knew where the respective ingredients are located in the kitchen. Once they give us a confirmation, they would perform the sequences of steps according to the transcript and a volunteer would instruct what the next step is from the background.

The recording platform consisted of 3 Full HD WiFi cameras and one GoPro attached to the forehead. A sample of the camera setup can be seen in fig. 5.1. If the background of the kitchen was constant for more than one subject, the ingredients were moved in the kitchen and the camera locations were also moved to reduce bias.

We created unique transcripts for all the dishes for different subjects. Table 5.5 refers to a sample transcript for all the 15 dishes which was used to instruct the subject.

5.3 Dataset Properties

The dataset is unique in many aspects and we have compared it with the current datasets used by the research community. We divide the dataset properties in 3 categories: recording properties, environment settings and label distributions.

5.3.1 Recording Properties

Table 5.1 illustrates the unique recording properrties when compared to other existing datasets. In terms of video resolution this is the only dataset that ego-centric and third person view, both having resolution as 1920x1080. When compared to the length of the videos, (3+1)ReC has an average length of 3.6 min while the maximum length of the video is 10.3 min. The other competitors like Epic Kitchen [78] has very long videos (61.8min) but they lack first person view. Similarly EGTEA+ has longer first person view but they don't lack third person view. Our dataset has a healthy balance of multi-view(ego-centric and thirdperson) and has competitive longer length for long duration action modelling.

5.3.2 Environment settings

Table 5.2 illustrates the distribution of environment settings. While Epic Kitchen[78] has highest number of environment settings, they are only ego-centric videos. Also our dataset has highest number of videos (1799) and the number of tasks are highest as compared to all other datasets.

Dataset	ave video length (min)	Ego centric view	# third person views	Video Resolution
50 Salad	6.4	X	1	640 x 480
IKEA	1.9	X	3	1920 x 1080
Breakfast	2.3	X	2-5	320×240
Epic Kitchens 100	8.6 (61.8 max)	Y	X	1920x1080
GTEA	1.2	Y	X	720 x 404
EGTEA +	>19 (54.3 max)	Y	X	1280x960
1ReC	3.6 (10.3 max)	Y	X	1920x1080
3ReC	3.6 (10.3 max)	X	3	1920x1080
(3+1)ReC	3.6 (10.3 max)	Y	3	1920x1080

Table 5.1: Comparison of current datasets based on video recording properties

Dataset	# subjects	# env settings	# vids	#tasks
50 Salad	25	1	50	1
IKEA	48	5	1113	4
Breakfast	52	18	1712	10
Epic Kitchens 100	37	45	700	NA
GTEA	4	1	28	7
EGTEA +	32	1	86	7
1ReC	30	8	450	15
3ReC	30	10	1349	15
(3+1)ReC	30	10	1799	15

Table 5.2: Comparison of current datasets based on task and environment properties

Dataset	#actions (w/o bg)	# transcripts (w/ bg)	Ave #segments per video (with bg)	Max/Min #samples
Cooking 2	87*	272	95.5	1736(1736/1) (take out/apply plaster)
50 Salad	17	50	20	1.55(62/40) (place tomato/add dressing)
IKEA	31**	359	22.7	159.4(3348/21) (spin leg/lay down table top)
Breakfast	47	256	6.9	68.5(685/10) (pour_milk/stir_tea)
Epic Kitchens 100	3769^	700	128*	1890*(1890/1) (turn on tap/pour celery)
GTEA	10*	28	33	35(140/4 (take/fold)
EGTEA +	106	86	239	23.5(752/32) (read/close oil container)
1ReC	102	418	11.7	16.4(246/15) (Take_Spoon/Pour_Egg2Pan)
3ReC	102	441	11.7	16.4(738/45) (Take_Spoon/Pour_Egg2Pan)
(3+1)ReC	102	444	11.7	16.4(984/60) (Take_Spoon/Pour_Egg2Pan)

Table 5.3: Comparison of current datasets based on transcript properties

5.3.3 Label distributions

Table 5.3 refers to the label distributions based on the annotations for all the datasets. Compared to Breakfast dataset[75], we have a much better class distribution when compared to the max/min count for each label. For Breakfast dataset it was 685/10 while for our dataset it is 984/60. This will ensure sufficient number of samples in training and test split for training and testing. Also when compared to Epic Kitchens 100[78], which is 1890/1, our dataset even when compared to only ego-centric view, we have a better distribution of max/min of 246/15.

5.3.4 Annotations

We enabled our dataset for future research to be utilized for all supervision methods. For that we have created frame-level annotations that will help fully supervised methods. We also have transcript annotations that will enable researchers in weakly supervised setting. Most importantly we also provide timestamp annotations that will enable

Split	F1@10	F1@25	F1@50	Edit	Acc
1	29.2158	25.9172	18.6469	28.8902	34.4656
2	31.5951	27.9908	20.2454	30.2096	36.6419
3	61.7154	58.5529	49.7365	57.1333	61.7072
4	58.0175	53.7415	46.1613	53.9395	56.838
5	44.1012	40.9266	32.6898	44.4721	50.2286
Average	44.929	41.4258	33.49598	42.92894	47.97626

Table 5.4: Fully Supervised

Figure 5.1: Camera Setup for Recording

timestamp or semi-supervised training. For timestamp annotations we randomly sampled a single frame-level annotations from the respective action-segments. We also tried to keep the annotations among views of a same dish to be consistent so that it can also help multi-view action action recognition research. For that we tried to temporally synchronize multiple views of the same dish video and assign common ground-truth label. Videos which couldn't be synchronized because of frame drops for a same dish were annotated separately. The start of an action was assigned when the volunteer in the video showed intention of performing that action and there was a labelling buffer of 0.5 seconds that helped synchronize labels of other views for the same dish. With this approach videos that were synchronized would have start and end time errors for an action of less than 0.5 seconds.

Dish Name	Transcript
Avocado Toast	Take_CuttingBoard-Take_Bread-Toast_Bread-Take_Knife-Take_Avocado-Take_Bowl-Cut_Avocado-Take_Spoon-Spoon_Avocado-Mash_Avocado-Sprinkle_Seasoning-Spread_Avocado
Bana Pancake	Take_Banana-Take_Bowl-Pour_Oil-Peel_Banana-Add_Banana-Take_Spoon-Mash_Banana-Take_Egg-Crack_Egg-Whisk_Batter-Sprinkle_Seasoning-Whisk_Batter-Pour_Batter2Pan-Fry_Pancake-Take_Plate-Put_Pancake2Plate
Cereal	Take_Bowl-Take_Milk-Pour_Milk-Take_Cereal-Pour_Cereal-Take_Spoon-Mix_Cereal
Chocolate Milk	Take_Cup-Take_Milk-Pour_Powder-Pour_Milk-Pour_Sugar-Take_Spoon- Stir_Milk
Coffee	Take_Cup-Take_Milk-Pour_Milk-Pour_Coffee-Microwave_Cup-Pour_Sugar-Take_Spoon-Stir_Coffee
French Toast	Take_Bread-Take_Egg-Take_Milk-Take_Bowl-Add_Butter-Crack_Egg-Sprinkle_Seasoning-Pour_Milk-Take_Spoon-Whisk_Egg-Dip_Bread-Put_Bread2Pan-Fry_Toast-Take_Plate-Put_Toast2Plate
Fried Eggs	Pour_Oil-Take_Egg-Take_Bowl-Crack_Egg-Sprinkle_Seasoning-Take_Tomato-Wash_Tomato-Take_Knife-Cut_Tomato-Take_Spoon-Whisk_Egg-Pour_Egg2Pan-Fry_Egg-Take_Plate-Put_Egg2Plate
Fruit Salad	Take_Bowl-Take_Knife-Take_CuttingBoard-Take_Cucumber-Take_Strawberry-Take_Apple-Take_Banana-Peel_Banana-Wash_Cucumber-Wash_Strawberry-Wash_Apple-Cut_Cucumber-Cut_Strawberry-Cut_Banana-Cut_Apple-Sprinkle_Seasoning-Put_Fruit2Bowl-Mix_Salad
Hashbrown	Take_Bowl-Take_Potato-Take_Onion-Pour_Oil-Peel_Potato-Take_Grater-Grate_Potato-Grate_Onion-Squeeze_Hashbrown-Put_Hashbrown2Pan-Wash_Hands-Fry_Hashbrown-Take_Plate-Put_Hashbrown2Plate
Lemonade	Take_Cup-Pour_Water-Take_Lemon-Take_Knife-Take_CuttingBoard-Cut_Lemon-Pour_Sugar-Squeeze_Lemon-Take_Spoon-Stir_Lemonade
Oatmeal	Take_Bowl-Take_Milk-Pour_Oat-Pour_Milk-Take_Strawberry-Wash_Strawberry-Take_Banana-Take_Knife-Take_CuttingBoard-Peel_Banana-Cut_Banana-Add_Banana-Cut_Strawberry-Add_Strawberry-Sprinkle_Seasoning-Take_Spoon-Stir_Oatmeal
PeanutButter Sandwich	Take_Plate-Take_Bread-Toast_Bread-Take_Knife-Cut_Bread-Take_PeanutButter-Take_Jam-Take_Spoon-Spread_Peanutbutter-Spread_Jam-Put_Bread
Tea	Fill_Kettle-Boil_Water-Take_Teabag-Take_Cup-Add_Teabag-Pour_Water-Pour_Sugar-Take_Spoon-Stir_Tea
Vegetable Sandwich	Take_Bread-Take_Cucumber-Take_Tomato-Take_Onion-Wash_Cucumber-Wash_Tomato-Cut_Tomato-Cut_Cucumber-Cut_Onion-Cut_Bread-Put_Tomato-Put_Cucumber-Put_Onion-Sprinkle_Seasoning-Put_Bread
Orange Juice	Take_Orange-Take_Knife-Take_CuttingBoard-Cut_Orange-Take_Squeezer-Take_Cup-Squeeze_Orange-Pour_Juice

Table 5.5: Sample Transcripts for a subject for 15 Dishes

Figure 5.2: Single Object HOI Detection

To enable further research on HOI, we also ran human object interaction detector[72] to get the bounding box locations from individual frames. Fig 5.2 illustrates the HOI detection for the bounding for one of the third person view. The output stored were the locations of the right and left hand boxes and also the blue bounding box indicating the object where HOI happened.

Similarly Fig 5.3 illustrates the HOI detection for the bounding for the third person view. The HOI detector consistently showed good outputs for the ego-centric view as well.

Fig 5.4 illustrates the HOI detection for the bounding for the third person view where there were 2 objects the human was interacting with. The right hand was interacting with the wooden spatula while the left one was interacting with the pan. We believe this would also enable further multi-object human object interacting tracking for better activity modeling.

Split Index	Subject ID Range
1	P1-P5
2	P6-P10
3	P11-P15
4	P21-P25
5	P26-P30
6	P16-P20

Table 5.6: Subject Id-wise splits for cross-validation

Splits	F1@10	F1@25	F1@50	Edit	Acc
1	24.4023	18.0028	7.1027	24.709	20.5274
2	23.0229	16.9396	7.6743	24.3647	19.1439
3	41.8873	31.0063	11.0737	40.9283	33.1738
4	41.7433	30.8959	12.3971	43.6891	34.0279
5	39.3612	27.6146	9.8918	41.0297	27.8333
6	43.5104	32.5173	13.2102	45.1729	30.2173
Average	34.0834	24.89184	10.2249	36.64895	27.4872

Table 5.7: Results for Timestamp Supervision on Ego-centric Data

Splits	F1@10	F1@25	F1@50	Edit	Acc
1	29.2158	25.9172	18.6469	28.8902	34.4656
2	31.5951	27.9908	20.2454	30.2096	36.6419
3	61.7154	58.5529	49.7365	57.1333	61.7072
4	58.0175	53.7415	46.1613	53.9395	56.838
5	44.1012	40.9266	32.6898	44.4721	50.2286
6	63.3366	60.2328	50.5335	60.242	57.8541
Average	47.9969	44.5603	36.3356	45.8145	49.6226

Table 5.8: Results for Full Supervision on Ego-centric (3+1)Rec Data

Splits	F1@10	F1@25	F1@50	Edit	Acc
1	37.3250	30.7932	17.9368	45.1231	22.3924
2	33.9899	29.2628	18.4581	39.7927	17.9611
3	40.6667	34.0000	20.1111	44.0624	22.5466
4	43.7428	37.4282	25.1435	47.2812	28.1340
5	37.1610	31.5061	21.1194	40.2220	20.5108
6	42.7054	37.2275	24.9301	47.3061	25.1894
Average	38.5771	32.5981	20.5538	43.9646	22.7890

Table 5.9: Results for Weak Supervision on Ego-centric (3+1)Rec Data

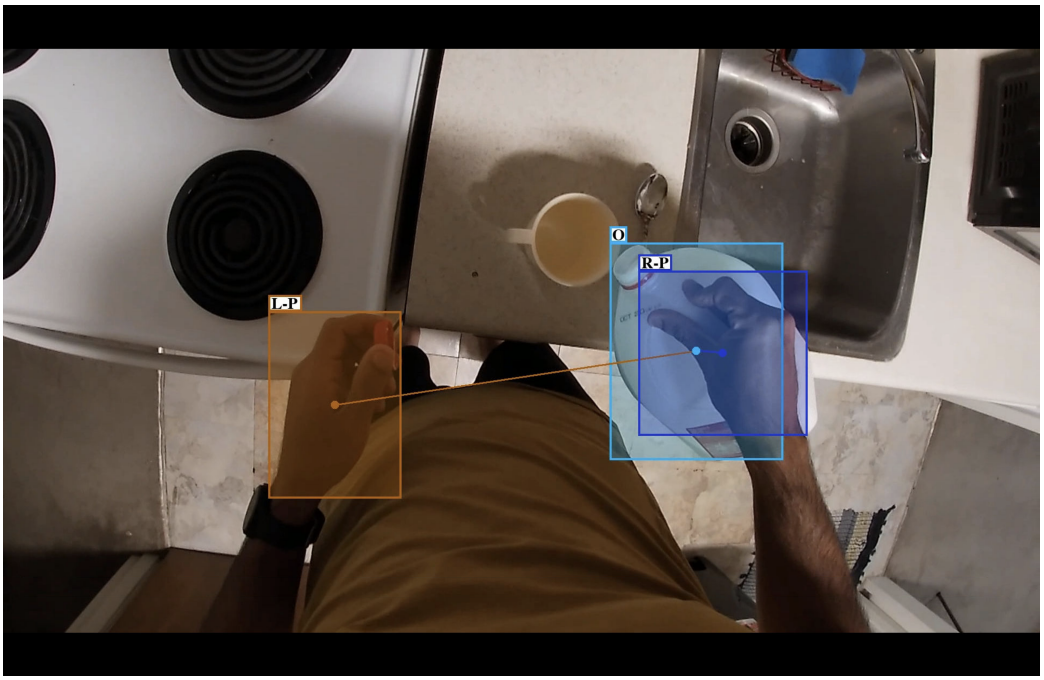


Figure 5.3: Ego centric HOI Detection

5.4 Baseline experiments and results

To provide baselines for future research we utilized a temporal Convolution network architecture [63] and trained them in Fully supervised and timestamp supervised fashion. Similarly for weakly supervised setting, we utilized mutual consistency network and loss[] to train. These method will act as a baseline for future research in many fields and using the HOI spatial predictions, a separate field of research can be made. Similarly since the annotations are synchronized, a mullti-view action segmentation approaches can be formulated which can utlize understanding multi-view analysis of human object interaction.

We used a 6 fold cross-validation settings and Table 5.6 refers to the 6 fold cross-validation settings. For faster analysis, we utilized only the ego-centricc views for training and testing. We Extracted the I3D features using the implementation[37] which was pretrained on Kinetics dataset. We chose a temporal window of 16 frames to compute the I3D features. The I3D extraction primarily takes 2 modalities of data: RGB and flow and we computed per frame flow features. As the I3D network cannot take a big input size like 1920x1080 dimension data, we divided each frame into 2 windows of width of 960 and extracted I3D features of these 2 windows. The I3D features extracted from each

Figure 5.4: Multi obj HOI Detection

modality is concatenated together to produce a feature representation $\mathbf{x}_i \in \mathbb{R}^{2048 \times T_i}$, where T_i is the length of the video i. As the length of the videos is very long, we reduced the dimensions of the data using incremental PCA as computing PCA on the entire dataset was memory consuming. We reduced the dimension from $\mathbb{R}^{2048 \times T_i}$ to $\mathbb{R}^{100 \times T_i}$.

For fully supervised setting we trained the TCN network for 70 epochs using the same conditions as mentioned by the author except we had a frame-level loss generated from the ground-truth from the annotations. For Timestamp and for network using HOI, we used the performance metrics and settings constant as mentioned in 4.5.3

Table 5.9 refers to the split-wise distribution of the performance for the weakly supervised setting. The performance in all metrics is consistent and the overall frame-level accuracy is low as mutual consistency is a weakly supervised network and it utilizes viterbi algorithm to decode the frame-level predictions at test time.

Table 5.8 refers to the split-wise distribution of the performance for the Fully supervised setting. The performance metrics are understandably high and the frame-level accuracy is still not as high as compared to other datasets. This shows that the dataset is challenging and there needs to be improvement in modeling the spatio-temporal features.

Table 5.7 refers to the split-wise distribution of the performance for the timestamp su-

Splits	F1@10	F1@25	F1@50	Edit	Acc
1	23.5336	17.9201	8.2044	27.0587	22.0824
2	24.1116	18.2636	8.0072	26.1095	21.2022
3	43.0724	31.1133	13.4176	46.435	30.0908
4	44.7776	34.083	15.7921	44.7295	30.7489
5	40.5024	30.2459	11.9309	41.5375	36.241
6	49.0329	37.5242	16.5377	49.4089	33.1053
Average	37.5051	28.1917	12.3150	39.2132	28.9118

Table 5.10: Results for Timestamp + HOI Supervision on Ego-centric (3+1)Rec Data

Supervision	F1@10	F1@25	F1@50	Edit	Acc
Timestamp	34.0834	24.8918	9.6279	34.9442	26.9413
Ours	**37.5051**	**28.1917**	**12.3150**	**39.2132**	**28.9118**
Full	47.9969	44.5603	36.3356	45.8145	49.6226

Table 5.11: Cumulative results for Timestamp and Full Supervision on Ego-centric (3+1)Rec Data

pervised setting. For split 1 and 2, the network shows lower performance when compared to other splits as the data was much challenging in the scenarios in terms of environment settings.

Table 5.10 refers to the split-wise distribution of the performance for the timestamp supervised setting using HOI. For split 1 and 2, the network showed similar lower performance.

Table 5.11 refers to the overall cross-validation results. It can be seen clearly that keeping the network consistent and adding pseudo-groundtruths using HOI, we can improve the timestamp supervised results with no additional cost of annotations.

5.5 Qualitative Comparison for Timestamp Supervision with HOI

Figure 5.5 illustrates a qualitative example of segmentation results using the timestamp's implementation when compared to our method which utilizes HOI. The left figure illustrates the result for dish "Fried Eggs" and the highlighted section is for the label "Fry Egg". It can be seen HOI provided better action segmentation performance. Similarly for the right image of for dish "Orange Juice" and the class label "Squeeze Orange" was segmented much better using HOI. The left and right top images illustrates predictions of HOi in blue boxes which helped in providing better pseudo ground-truths thereby improving

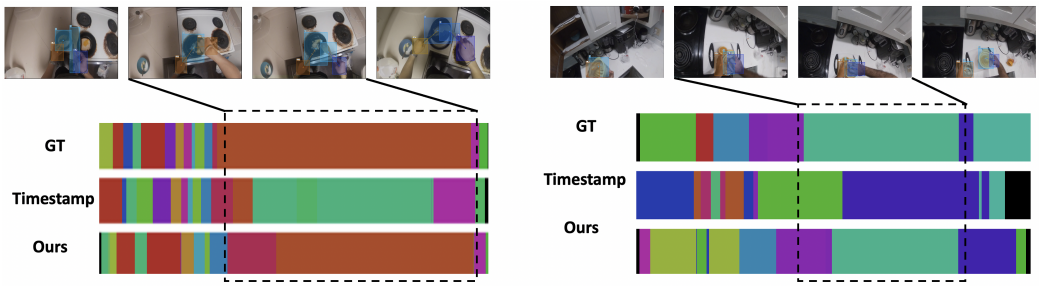


Figure 5.5: Qualitative results for Ego-centric prediction compared with ours

the performance of the system.

5.6 Conclusion

We created a new dataset named (3+1)Rec which is first of it's kind dataset that has third person and first person view for long term video analysis. The dataset provides high quality long-term videos with 15 video level labels or dishes and efforts were made to provide every dish with unique transcript. When compared to the current datasets, we provide not only timestamp labels but also full frame-level labels and label consistency is maintained for all the videos. To enable future multi-view action segmentation research, we also made sure that the labels for a dish are temporally consisted over all the 4 views. Since all the activities are performed by humans doing regular kitchen activities, there is a potential research avenue which can benefit analysing human object interaction and we have also provided human object interaction labels. This dataset can be used for full, weak and timestamp supervised setting and we have provided baseline by training them on the current state of the art.

6 Conclusion

The thesis makes several assumptions, and is limited by the extent to which those assumptions hold in a specific dataset. One assumption is that each video displays a single human performing activities involving interaction with objects. This assumption is relevant in many real-world applications, and it is true in commonly used datasets, such as the ones we have used in our experiments. At the same time, clearly there can be action recognition domains where this assumption does not apply. For example, this assumption would not apply for distinguishing between activities such as “walking” and “running”.

In the first section we created action recognition systems that track the body, rgb and flow information. This work was showcased how computer vision systems can be used to evaluate onset of cognitive disorders such as ADHD in kids through an unobtrusive, accessible and easy-to-use framework. We created a dataset having real-world usage with children performing fine-grained motion patterns having high intra-class variability. We created a system that will be useful in advancing reasearch in cognitive assessment of kids. This system produces scores that can directly be transferred to measure executive functioning which is a key perdictor for onset of ADHD in adolescent kids.

In the second section, we propose an approach to train an action segmentation model by employing the timestamp annotations and concept of human object interaction. Our model extends the single frame timestamp annotations using the frame level predictions of a human object detector. We track the object bounding boxes where the human is interacting around the timestamp, thereby pseudo ground-truths. We also use improved the existing frame-level action class generator by adding a constraint that an action boundary cannot exist around frames where the human is continuously interacting with the object. We improved the temporal modelling by replacing a temporal convolutional based model to a transformer. To enable future research on multi-view action segmentation and also utilize human object interaction information, we introduce a new dataset called (3+1)Rec and this dataset can be utilized in several supervision settings such as full, weak and timestamp. We believe that given it’s high quality, wide variety of environment settings and a

balanced class distribution, this dataset can be utilized in understand human actions in long duraton videos.

References

[1] D. H. Schunk and B. J. Zimmerman, "Social origins of self-regulatory competence," *Educational psychologist*, vol. 32, no. 4, pp. 195–208, 1997.

[2] F. J. Morrison, C. C. Ponitz, and M. M. McClelland, "Self-regulation and academic achievement in the transition to school," *Child development at the intersection of emotion and cognition*, vol. 1, pp. 203–224, 2010.

[3] J. R. Best and P. H. Miller, "A developmental perspective on executive function," *Child development*, vol. 81, no. 6, pp. 1641–1660, 2010.

[4] R. A. Barkley, "Behavioral inhibition, sustained attention, and executive functions: constructing a unifying theory of adhd." *Psychological bulletin*, vol. 121, no. 1, p. 65, 1997.

[5] E. Cormier, "Attention deficit/hyperactivity disorder: a review and update," *Journal of pediatric nursing*, vol. 23, no. 5, pp. 345–357, 2008.

[6] D. W. Dunn and W. G. Kronenberger, "Attention-deficit/hyperactivity disorder in children and adolescents." *Neurologic clinics*, 2003.

[7] C. Dendy, "Executive function"what is this anyway?"," *Retrieved September*, vol. 18, p. 2008, 2008.

[8] M. M. McClelland, C. E. Cameron, R. Duncan, R. P. Bowles, A. C. Acock, A. Miao, and M. E. Pratt, "Predictors of early growth in academic achievement: The head-toes-knees-shoulders task," *Frontiers in psychology*, vol. 5, p. 599, 2014.

[9] S. Gattupalli, D. Ebert, M. Papakostas, F. Makedon, and V. Athitsos, "Cognilearn: A deep learning-based interface for cognitive behavior assessment," in *Proceedings*

of the 22nd International Conference on Intelligent User Interfaces. ACM, 2017, pp. 577–587.

[10] E. Insafutdinov, L. Pishchulin, B. Andres, M. Andriluka, and B. Schiele, "Deepercut: A deeper, stronger, and faster multi-person pose estimation model," in *European Conference on Computer Vision*. Springer, 2016, pp. 34–50.

[11] J. Howcroft, S. Klejman, D. Fehlings, V. Wright, K. Zabjek, J. Andrysek, and E. Biddiss, "Active video game play in children with cerebral palsy: potential for physical activity promotion and rehabilitation therapies," *Archives of physical medicine and rehabilitation*, vol. 93, no. 8, pp. 1448–1456, 2012.

[12] A. Billard, B. Robins, J. Nadel, and K. Dautenhahn, "Building robota, a mini-humanoid robot for the rehabilitation of children with autism," *Assistive Technology*, vol. 19, no. 1, pp. 37–49, 2007.

[13] A. A. Rizzo, J. G. Buckwalter, T. Bowerly, C. Van Der Zaag, L. Humphrey, U. Neumann, C. Chua, C. Kyriakakis, A. Van Rooyen, and D. Sisemore, "The virtual classroom: a virtual reality environment for the assessment and rehabilitation of attention deficits," *CyberPsychology & Behavior*, vol. 3, no. 3, pp. 483–499, 2000.

[14] A. P. Association *et al.*, *Diagnostic and statistical manual of mental disorders (DSM-5®)*. American Psychiatric Pub, 2013.

[15] E. Hotham, M. Haberfield, S. Hillier, J. M. White, and G. Todd, "Upper limb function in children with attention-deficit/hyperactivity disorder (adhd)," *Journal of Neural Transmission*, vol. 125, no. 4, pp. 713–726, 2018.

[16] H. J. Kam, K. Lee, S.-M. Cho, Y.-M. Shin, and R. W. Park, "High-resolution actigraphic analysis of adhd: A wide range of movement variability observation in three school courses-a pilot study," *Healthcare informatics research*, vol. 17, no. 1, pp. 29–37, 2011.

[17] M. Muñoz-Organero, L. Powell, B. Heller, V. Harpin, and J. Parker, "Automatic extraction and detection of characteristic movement patterns in children with adhd based on a convolutional neural network (cnn) and acceleration images," *Sensors*, vol. 18, no. 11, p. 3924, 2018.

[18] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[19] J. Steward, D. Lichti, J. Chow, R. Ferber, and S. Osis, "Performance assessment and calibration of the kinect 2.0 time-of-flight range camera for use in motion capture applications," *FIG Working Week 2015*, pp. 1–14, 2015.

[20] E. Lachat, H. Macher, M. Mittet, T. Landes, and P. Grussenmeyer, "First experiences with kinect v2 sensor for close range 3d modelling," *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 40, no. 5, p. 93, 2015.

[21] L. Zhou, Z. Liu, H. Leung, and H. P. Shum, "Posture reconstruction using kinect with a probabilistic model," in *Proceedings of the 20th ACM Symposium on Virtual Reality Software and Technology*. ACM, 2014, pp. 117–125.

[22] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields," in *arXiv preprint arXiv:1812.08008*, 2018.

[23] W. H. Organization, "The world health report 2001: Mental health: new understanding, new hope," 2001.

[24] A. Riaz, M. Asad, S. M. R. Al Arif, E. Alonso, D. Dima, P. Corr, and G. Slabaugh, "Deep fmri: An end-to-end deep network for classification of fmri data," in *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*. IEEE, 2018, pp. 1419–1422.

[25] S. Jaiswal, M. F. Valstar, A. Gillott, and D. Daley, "Automatic detection of adhd and asd from expressive behaviour in rgbd data," in *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*. IEEE, 2017, pp. 762–769.

[26] H.-J. Dai and J. Jonnagaddala, "Assessing the severity of positive valence symptoms in initial psychiatric evaluation records: Should we use convolutional neural networks?" *PloS one*, vol. 13, no. 10, p. e0204493, 2018.

[27] M. Wellsby and P. M. Pexman, "Developing embodied cognition: insights from children's concepts and language processing," *Frontiers in psychology*, vol. 5, p. 506, 2014.

[28] M. D. Bell, A. J. Weinstein, B. Pittman, R. M. Gorman, and M. Abujelala, "The activate test of embodied cognition (atec): Reliability, concurrent validity and discriminant validity in a community sample of children using cognitively demanding physical tasks related to executive functioning," *Child Neuropsychology*, pp. 1–11, 2021.

[29] V. Krieger and J. A. Amador-Campos, "Assessment of executive function in adhd adolescents: contribution of performance tests and rating scales," *Child Neuropsychology*, vol. 24, no. 8, pp. 1063–1087, 2018.

[30] S. I. Sayed, K. Tsiakas, M. Bell, V. Athitsos, and F. Makedon, "Cognitive assessment in children through motion capture and computer vision: the cross-your-body task," in *Proceedings of the 6th international Workshop on Sensor-based Activity Recognition and Interaction*, 2019, pp. 1–6.

[31] L. Zou, J. Zheng, and M. J. McKeown, "Deep learning based automatic diagnoses of attention deficit hyperactive disorder," in *2017 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. IEEE, 2017, pp. 962–966.

[32] M. Fayyaz and J. Gall, "Sct: Set constrained temporal transformer for set supervised action segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 501–510.

[33] S.-J. Li, Y. AbuFarha, Y. Liu, M.-M. Cheng, and J. Gall, "Ms-tcn++: Multi-stage temporal convolutional network for action segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[34] Y. Ishikawa, S. Kasai, Y. Aoki, and H. Kataoka, "Alleviating over-segmentation errors by detecting action boundaries," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021, pp. 2322–2331.

[35] D. Wang, D. Hu, X. Li, and D. Dou, "Temporal relational modeling with self-supervision for action segmentation," *arXiv preprint arXiv:2012.07508*, 2020.

[36] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[37] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6299–6308.

[38] Y. A. Farha and J. Gall, "Ms-tcn: Multi-stage temporal convolutional network for action segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3575–3584.

[39] D. Shao, Y. Zhao, B. Dai, and D. Lin, "Finegym: A hierarchical video dataset for fine-grained action understanding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 2616–2625.

[40] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," *arXiv preprint arXiv:1406.2199*, 2014.

[41] A. Diba, M. Fayyaz, V. Sharma, M. M. Arzani, R. Yousefzadeh, J. Gall, and L. Van Gool, "Spatio-temporal channel correlation networks for action classification," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 284–299.

[42] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7794–7803.

[43] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 6202–6211.

[44] H. Kuehne, J. Gall, and T. Serre, "An end-to-end generative framework for video segmentation and recognition," in *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2016, pp. 1–8.

[45] C. Lea, M. D. Flynn, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks for action segmentation and detection," in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 156–165.

[46] Y. Zhao, Y. Xiong, L. Wang, Z. Wu, X. Tang, and D. Lin, "Temporal action detection with structured segment networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2914–2923.

[47] D.-A. Huang, L. Fei-Fei, and J. C. Niebles, "Connectionist temporal modeling for weakly supervised action labeling," in *European Conference on Computer Vision*. Springer, 2016, pp. 137–153.

[48] A. Richard, H. Kuehne, and J. Gall, "Weakly supervised action learning with rnn based fine-to-coarse modeling," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017, pp. 754–763.

[49] L. Ding and C. Xu, "Weakly-supervised action segmentation with iterative soft boundary assignment," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6508–6516.

[50] A. Richard, H. Kuehne, A. Iqbal, and J. Gall, "Neuralnetwork-viterbi: A framework for weakly supervised video learning," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018, pp. 7386–7395.

[51] J. Li, P. Lei, and S. Todorovic, "Weakly supervised energy-based learning for action segmentation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 6243–6251.

[52] C.-Y. Chang, D.-A. Huang, Y. Sui, L. Fei-Fei, and J. C. Niebles, "D3tw: Discriminative differentiable dynamic time warping for weakly supervised action alignment and segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3546–3555.

[53] A. Richard, H. Kuehne, and J. Gall, "Action sets: Weakly supervised action segmentation without ordering constraints," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018, pp. 5987–5996.

[54] J. Li and S. Todorovic, "Set-constrained viterbi for set-supervised action segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10 820–10 829.

[55] D. Moltisanti, S. Fidler, and D. Damen, "Action recognition from single timestamp supervision in untrimmed videos," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 9915–9924.

[56] S. Stein and S. J. McKenna, "Combining embedded accelerometers with computer vision for recognizing food preparation activities," in *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*, 2013, pp. 729–738.

[57] M. Rohrbach, A. Rohrbach, M. Regneri, S. Amin, M. Andriluka, M. Pinkal, and B. Schiele, "Recognizing fine-grained and composite activities using hand-centric features and script data," *International Journal of Computer Vision*, vol. 119, no. 3, pp. 346–373, 2016.

[58] A. Fathi, X. Ren, and J. M. Rehg, "Learning to recognize objects in egocentric activities," in *CVPR 2011*. IEEE, 2011, pp. 3281–3288.

[59] H. Kuehne, A. Richard, and J. Gall, "Weakly supervised learning of actions from transcripts," *Computer Vision and Image Understanding*, vol. 163, pp. 78–89, 2017.

[60] ——, "A hybrid rnn-hmm approach for weakly supervised temporal action segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 42, no. 4, pp. 765–779, 2018.

[61] Y. Souri, M. Fayyaz, L. Minciullo, G. Francesca, and J. Gall, "Fast weakly supervised action segmentation using mutual consistency," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[62] P. Lee, J. Wang, Y. Lu, and H. Byun, "Weakly-supervised temporal action localization by uncertainty modeling," *arXiv preprint arXiv:2006.07006*, 2020.

[63] Z. Li, Y. Abu Farha, and J. Gall, "Temporal action segmentation from timestamp supervision," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 8365–8374.

[64] Y.-W. Chao, Z. Wang, Y. He, J. Wang, and J. Deng, "Hico: A benchmark for recognizing human-object interactions in images," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1017–1025.

[65] Y.-W. Chao, Y. Liu, X. Liu, H. Zeng, and J. Deng, "Learning to detect human-object interactions," in *2018 ieee winter conference on applications of computer vision (wacv)*. IEEE, 2018, pp. 381–389.

[66] S. Gupta and J. Malik, "Visual semantic role labeling," *arXiv preprint arXiv:1505.04474*, 2015.

[67] M. Tamura, H. Ohashi, and T. Yoshinaga, "Qpic: Query-based pairwise human-object interaction detection with image-wide contextual information," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 10 410–10 419.

[68] A. Gupta, A. Kembhavi, and L. S. Davis, "Observing human-object interactions: Using spatial and functional compatibility for recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 31, no. 10, pp. 1775–1789, 2009.

[69] H. S. Koppula and A. Saxena, "Anticipating human activities using object affordances for reactive robotic response," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 1, pp. 14–29, 2015.

[70] T. Nagarajan, C. Feichtenhofer, and K. Grauman, "Grounded human-object interaction hotspots from video," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 8688–8697.

[71] T. Nagarajan, Y. Li, C. Feichtenhofer, and K. Grauman, "Ego-topo: Environment affordances from egocentric video," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 163–172.

[72] D. Shan, J. Geng, M. Shu, and D. F. Fouhey, "Understanding human hands in contact at internet scale," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9869–9878.

[73] G. Gkioxari, R. Girshick, P. Dollár, and K. He, "Detecting and recognizing human-object interactions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8359–8367.

[74] F. Ma, L. Zhu, Y. Yang, S. Zha, G. Kundu, M. Feiszli, and Z. Shou, "Sf-net: Single-frame supervision for temporal action localization," in *European conference on computer vision*. Springer, 2020, pp. 420–437.

[75] H. Kuehne, A. Arslan, and T. Serre, "The language of actions: Recovering the syntax and semantics of goal-directed human activities," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 780–787.

[76] Z. Wang, Z. Gao, L. Wang, Z. Li, and G. Wu, "Boundary-aware cascade networks for temporal action segmentation," in *European Conference on Computer Vision*. Springer, 2020, pp. 34–51.

[77] F. Yi, H. Wen, and T. Jiang, "Asformer: Transformer for action segmentation," *arXiv preprint arXiv:2110.08568*, 2021.

[78] D. Damen, H. Doughty, G. M. Farinella, S. Fidler, A. Furnari, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price *et al.*, "Scaling egocentric vision: The epic-kitchens dataset," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 720–736.