EatSense: Human Centric, Action Recognition and Localization Dataset for Understanding Eating Behaviors and Quality of Motion Assessment.

Muhammad Ahmed Raza, Longfei Chen, Li Nanbo, Robert B. Fisher
The University of Edinburgh
School of Informatics

July 13, 2023

1 abstract

Current datasets for computer vision-based action recognition and localization cover a wide range of classes and challenging scenarios. However, these datasets don't cater to healthcare applications that involve long-term monitoring, tracking minor changes in movements over time for healthcare purposes, or completely modeling a specific human behavior that includes multiple sub-actions. Specifically, there are no existing datasets for research on either health monitoring on atomic-action-based eating behavior or for a full range of eating sub-actions that fully segment the main action. Addressing these gaps is valuable for extending research on the health monitoring of elderly people and is needed for creating richer and more complete descriptions of actions. This paper introduces a new benchmark dataset named EatSense that targets both the computer vision and healthcare communities and fills in the aforementioned gaps. EatSense is recorded while a person eats in an uncontrolled dining setting. The key features of EatSense are the introduction of challenging atomic actions for action recognition, the significantly diverse durations of actions that make it difficult for current temporal action localization frameworks to localize, the capability to model comprehensive eating behavior in terms of a sequence of action-based behaviors, and the simulation of minor variations in motion or performance. We conduct extensive experiments on EatSense with baseline deep learning-based approaches for benchmarking and hand-crafted feature-based approaches for explainable applications. We believe this dataset will benefit future researchers in building robust temporal action localization networks, behavior recognition, and performance assessment models for eating.

Keywords: EatSense, Eating Vision Dataset, Atomic-Action Recognition, Change in movement detection

2 Introduction

There are many extensive publicly available datasets for action recognition, temporal action localization and monitoring the daily activities of people [1]. These datasets contain various action classes for recognition, and temporal segments for localization and provide performance benchmarks for several commonly used algorithms [2]. Although the current datasets are extensive in terms of total recording hours, and many different and difficult scenarios, these still lack the capability to model a specific behavior or detect a change in motion to model decay in the motor movement of the subjects.

This leads us to why it is important to model behavior and identify minor changes. Modeling a person's behavior, such as their eating habits, can provide us with a more comprehensive understanding of their routine. Moreover, the ability to detect minor changes in motion can be incredibly valuable in situations where long-term monitoring is required, particularly for older individuals or for assessing changes in athletic performance [3],[4].

Most publicly available datasets have shortcomings such as: firstly, they contain only trimmed individual clips or sparse annotations in an untrimmed video instead of dense action labels, and secondly, they do not have the sub-actions (atomic actions) level of annotations; instead, they only offer the high-level action label, e.g., eating or drinking.

In this paper, we present a new densely annotated dataset named EatSense that is recorded while a person eats at a dining table in a real-world uncontrolled environment. EatSense is an unobtrusive, human-centric, upper-body-focused dataset that provides the capability to model eating behavior along with the ability to study the change in motion/motor deterioration. The change in motion is simulated by attaching weights to the wrists of the subjects while they eat[1]. Adding wrist weights to human movement can simulate increased muscle stiffness, leading to changes in movement patterns and kinematics. The concept of using weights to imitate upper-body decay has been confirmed in [5]. Likewise, in [6], a similar idea is employed to exhibit various gait abnormalities. In the past, different methods to temporarily induce palm stiffness and limit fine motor control of fingers have been explored [7], [8]. EatSense contains 27 subjects from 13 nationalities, hence introducing diversity in eating styles, tools used (fork, chopsticks, etc), and food selection. The contributions of this paper are:

- A new untrimmed dataset named EatSense for action recognition, temporal action localization and quality of motion assessment is presented.
  - We provide frame-wise, dense labels ($\approx$ 114.1 actions per video sequence) with three levels of abstractions (see section 4.2).
  - We provide detailed comparisons with other publicly available datasets where, unlike many datasets, EatSense contributes to both the computer vision and health-care communities (see table 4).
  - We provide experimental test benchmarks using recent approaches for action recognition and temporal action localization (see section 5).
  - The full dataset including the RGBD and skeleton data is publicly available.
- From an explainability point of view, hand-crafted features from the upper-body poses of the subject were extracted using domain knowledge and interactions between humans and objects,
  - to demonstrate that the dataset can be used for tasks where explainability is the key, such as healthcare applications where information about individual joints is vital to understand or diagnose/track/predict a problem (see section 6).
  - to demonstrate effective modeling of eating sub-action recognition using EatSense with reasonable accuracy from interpretable features (see section 6.2).
  - to demonstrate the application capability of EatSense for quality of motion assessment (see section 6.3).

[1]The weights are not intended to be a model for aging, but to demonstrate that minor changes in motion are detectable.

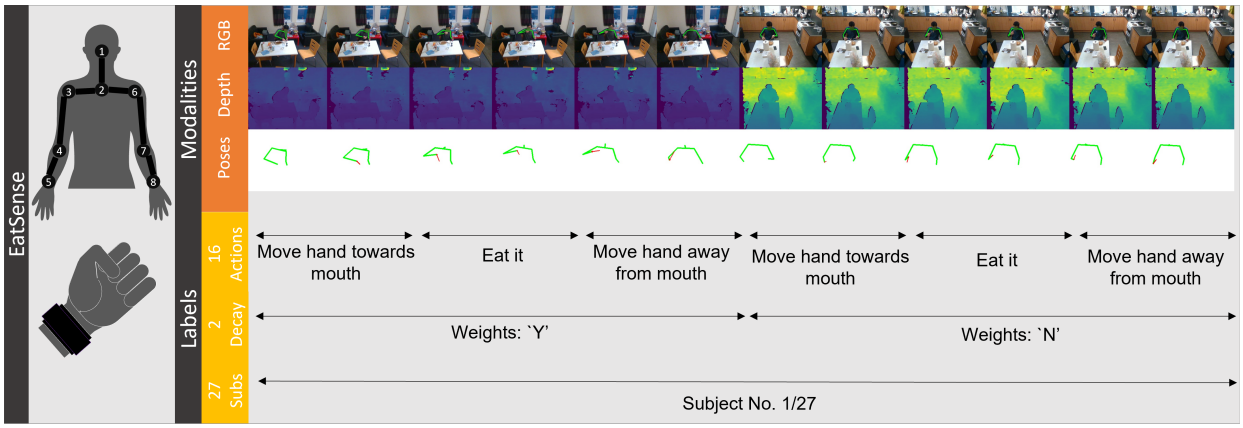

Figure 1: EatSense dataset is an eating sub-action analysis dataset, comprising multiple modalities, dense annotations and multiple abstractions of labels.

3 Literature Review

A review of publicly available datasets for the domain of action recognition, action localization and activities of daily living, is presented. This section also includes a brief overview of some commonly used action recognition and localization approaches.

3.1 Public Datasets

There are several publicly available action classification datasets. Some of the large-scale datasets can be divided into four categories according to their targeted application, i.e., datasets for action recognition (trimmed video datasets), datasets for temporal action localization (untrimmed video datasets), activities of daily living and quality of motion assessment datasets.

3.1.1 Trimmed Video Datasets

Datasets that contain one action per video sequence are classified as action recognition datasets as they do not present the possibility of temporally localizing the ongoing activity. NTU-RGB-D 120 [9] is one of the most extensive action-recognition benchmark datasets. It contains 114,480 video sequences of 106 subjects, and 120 action classes, which include numerous daily routine actions, group actions and medical conditions. It was collected via multiple sensors such as RGB, depth and infrared.

Kinetics-700 [10] is another large-scale dataset that contains 700 action classes, and 650,317 video clips collected from youtube with at least 450 clips for each action class. The action labels include a variety of actions including many actions involved in daily lifestyle such as 'digging', 'pouring milk' and 'drinking' etc. Goyal et. al. presented Something-Something v2 [11], which is an ego-centric dataset, that is focused on human hand gestures such as putting something on something or turning something upside down. It contains 220,847 videos recorded at 12 frames per second (fps), with 174 classes. This dataset was recorded in a setting where a person strictly performs a pre-defined set of actions with daily use objects.

HMDB51 [12] is another dataset brought together by a collection of youtube videos and digitized movies with at least 101 videos per class with a total of 7,000 videos manually annotated. HMDB51 contains 51 action labels which can generally be divided into 5 types, i.e., general facial actions, facial actions with object manipulation, general body movements, body movements with object interaction and body movements for

human interaction. Jhuang et. al. presented J-HMDB [13] which is a subset of the HMDB51 dataset with 21 classes where the authors provide annotations for human joints. These joint positions are further utilized to estimate ground-truth segmentation and optical flow.

As can be observed from this section, for many large-scale datasets (with hundreds of classes) the targeted applications are computer vision-based action recognition with a single clip-wide label. Thus, these datasets have a very limited capability to be used for healthcare or behavior understanding/modeling applications. On the other hand, EatSense contains full-length eating sessions of individuals and also introduces a new healthcare monitoring/motor function decay assessment dataset.

3.1.2 Untrimmed Video Datasets

There are various publicly available datasets that can be used to investigate the action localization problem, each with different label settings. These include: 1) video clips labeled with a single action, 2) videos marked with sparse labels, where there are long periods of inactivity between two actions, 3) videos with dense labels covering the entire video, meaning there are no unmarked sections, or the videos contain overlapping labels at any given time.

ActivityNet-1.3 [14] contains 203 actions that people do on a regular basis for example 'shovelling snow' or 'cleaning shoes'. These actions are broadly classified into vehicles, housework, animals, interior maintenance and exterior maintenance. ActivityNet is a large-scale dataset with an average of 137 untrimmed videos per action class for 849 hours of videos. It contains classes with sparse ground truth labels with 1.54 actions per clip of 1.9 minutes in length. FineGym [15] contains 530 sub-actions for example 'vault' and 'floor exercise' in untrimmed videos. This is a human-centric dataset, with a single subject in the field of view, collected from videos available on youtube where subjects (regardless of who are they) are performing various gymnastics.

PKU-MMD [16] is an extensive video dataset designed for action recognition and multi-modality action analysis. It is divided into two phases with 51 and 49 action labels respectively. The action labels can be categorized into two groups, consisting of 41 daily actions and 10 interaction actions. The dataset comprises a total of 1076 long videos (approximately 4 minutes) and 2000 short videos (approximately 2 minutes), all recorded from multiple perspectives.

There are several datasets available that contain dense labels. Two such datasets are AVA [17] and Sphere-H130 [18]. AVA contains 80 actions in 430 clips, each 15 minutes in length cropped from various films. Hence, this dataset includes instances with multiple subjects interacting with the environment or with each other. The Sphere-H130 action dataset contains 130 sequences of 13 actions of about 70 minutes in total performed by 5 subjects in a home setting. However, in this dataset, the subjects strictly perform a specific set of actions, hence it lacks real-world diversity.

UCF-101-24 [19] is another extensive action recognition dataset with realistic RGB videos downloaded from youtube. This contains 101 action categories with 13320 videos (27 hours) in total. These 101 categories can be broadly classified into five types, 1) Human-Object Interaction, 2) Body-Motion Only, 3) Human-Human Interaction, 4) Playing Musical Instruments, and 5) Sports. Epic-Kitchens [20], unlike Something-Something-v2, is a non-scripted dataset recorded where the subjects had the instruction to do the tasks in a kitchen however they like. Epic-Kitchens contains 4,053 classes on over 100 hours of high-definition kitchen recording sessions.

In conclusion, the datasets for action localization tasks have extensive sets of untrimmed videos and action labels, but many of them still lack dense temporal labels and others are short of consistent sets of sub-actions involved in one large action, hence they are rarely used for long-term behavior modeling of an individual. EatSense aims to fill these gaps, as it contains dense labels for the videos and 16 sub-actions involved in the eating action.

3.1.3 Eating Related Datasets

In a recent study, Tang et al. [21] introduced a dataset for intake gestures during meals, which is part of the Clemson Cafeteria Database (CCD) [22]. The video data was captured in a university cafeteria, with 276 participants consuming a total of 374 different foods and beverages. The dataset is composed of three different gesture classes, including bite, drink, and others. In another research work, Shengjie et al. [23] recorded an ego-centric dataset using a head-mounted camera in a free-living environment, where they formulated a binary classification problem to distinguish between eating and non-eating activities. Finally, for those interested in gesture detection, Neves et al. [24] presented a comprehensive review of approaches used in eating gesture detection.

OREBA (Objectively Recognizing Eating Behavior and Associated Intake) [25] is a dataset designed to provide researchers interested in detecting intake gestures (single gesture denoted as: Intake, Intake-Eat, Right, Spoon) with a large amount of data collected from multiple sensors during communal meals. This dataset includes recording via a 360-degree camera positioned at the front to record videos, as well as a sensor box that contains a gyroscope, an IMU and an accelerometer attached to both hands. Other research studies have also presented small-scale datasets such as Accelerometer and audio-based Calorie Estimation (ACE) [26], Clemson [27] and Food Intake Cycle (FIC) [28] that primarily focus on characteristics of intake gestures, such as chews, and swallowing behaviors.

Men et al. introduced a dataset [29] that aimed to differentiate between high-level actions related to the type of food being consumed, such as 'eat a steak' or 'drink from a plastic bottle'. The primary goal of this dataset was to estimate the frequency of self-feeding and to gain insights into eating/drinking behavior. They utilized Microsoft Kinect to capture skeleton motions. Mobiserv-AIIA [30] was designed for the evaluation of specialized meal intake to prevent undernourishment or malnutrition. The dataset contains captured videos recorded in a controlled laboratory environment with multiple cameras set up at various angles. Mobiserv-AIIA does not contain atomic actions, rather it focuses on high-level actions such as 'eat', 'drink' and 'slice' for different meals (breakfast, lunch and fast food) and using various tools (spoon, fork or glass of water, etc.).

To summarize, previous studies have presented various datasets and conducted research that deals with recognizing actions/gestures such as eating, drinking and swallowing, etc. but is limited in the sense that they do not explicitly highlight the most common sub-actions involved in the eating process. Onofri et. al. [31] explain that activity recognition-based behavior analysis algorithms require two categories of knowledge, namely contextual knowledge and prior knowledge. Also, most past (vision-based) datasets lack prior knowledge as they do not contain sub-action information. Hence, they fail to provide the capability to explore the complete behavior (based on many sub-actions involved while eating) of individual subjects. EatSense addresses this gap, as it contains the 16 most common sub-actions for the whole eating process.

3.1.4 Activities of Daily Living Datasets

The MSR-Action3D dataset [32] includes the 3D location of 20 joints in each frame. The dataset contains 20 actions such as 'tennis serve' and 'golf swing' etc. The MSR-DailyActivity dataset [33] was designed to model daily actions performed by a person while sitting on a couch. It contains 320 samples of 16 daily activities such as 'play guitar' and 'eat'.

Some trimmed and untrimmed video-based datasets such as Sphere-H130 [18], ActivityNet-1.3 [14] and Something-Something v2 [11] described in previous sections contain actions performed in a daily routine.

3.1.5 Quality of Motion Assessment Datasets

Many datasets not only focus on the ongoing activity but also quantify the quality of motion of the subject. Sphere-Walking [34] was designed for motion quality assessment via gait analysis. In this dataset, 6 subjects

were recorded while they climb up a flight of stairs and each of them was scored by health professionals. Init Gait DB [35] is a benchmark dataset for gait impairment research, recorded in a controlled laboratory environment. Eight different gait styles were simulated where the movement of limbs and posture of the human body was altered. It was recorded from multiple view angles using RGB cameras.

The walking gait dataset [6] is also a gait analysis-based dataset that simulates 9 walking gait patterns. These were simulated by adding a thick sole to one shoe or tying weights at the ankle. This was recorded via a Microsoft Kinect where the subject walked on a treadmill with two flat mirrors behind them. Sphere and other datasets rely on whole or lower-body gait analysis. Moreover, research such as [36], [37] and [38] presents a comprehensive overview of publicly available gait analysis databases. [36], [39] and [40] discuss challenges and solutions to gait analysis techniques in depth.

To the best of our knowledge, none of the current datasets concentrates specifically on the evaluation of human motion quality in relation to action-based eating behaviors, with a particular emphasis on the movement of the upper body joints.

3.2 Action Classification and Localization

Vision-based frameworks for action detection are classified into action recognition and temporal action localization.

3.2.1 Action Classification

In general, vision-based action classification/recognition[2] frameworks can be divided into two categories based on modalities, i.e., video-based and skeleton-based action recognition. For skeleton-based action recognition, many researchers have explored Recurrent Neural Network (RNN) or Long Short-Term Memory (LSTM) based approaches [41], [42], [43] with hand-crafted spatial features. These, however, ignore the spatial connectivity of the human body.

To incorporate human joint connectivity, some researchers proposed using Graph Convolutional Networks (GCN) or heatmaps for action recognition. In [44] Duan et al. proposed PoseConv3D (a.k.a. PoseC3D), which used a 3D heatmap volume as an input for a 3D-CNN network which made it less prone to joint estimation noise and thus more robust for action recognition. Yan et al. proposed the Spatio-Temporal Graph Convolutional Network (ST-GCN) [45], which established both spatial and temporal graph connections. Adaptive Graph Convolutional Network (AGCN) based approaches exploit the hierarchical structure of GCNs where different layers contain multi-level semantic information and thus incorporate long-range dependencies of the joints for action recognition [46], [47] and [48]. Recently, Chen et al. [49] proposed a feature aggregation topology (channel-wise topology refinement graph convolution - CTR-GC) that effectively aggregates joint features in various channels and dynamically learns different topologies.

Here, we only discuss a few RGB (2D/3D CNN) based action classifiers. Temporal Segment Networks (TSN) [50] first divides the video into snippets, uniformly and sparsely sample the frames and then average pool the samples to merge per-frame predictions. Temporal Pyramid Network (TPN) [51] introduced spatial and temporal modulation blocks to align semantics and adjust the tempo among multiple levels of features extracted from the backbone. Temporal Adaptive Network (TANet) [52] presented a temporal adaptive module that generates temporal kernels to capture global context information which, when used alongside a 2D CNN, produces a powerful action recognition framework.

There have been several studies on action recognition methods used for healthcare research. In [53] Gul et. al. proposed a YOLO-based action classifier along with a dataset to identify eight abnormalities such as 'backward fall' and 'chest pain', etc., collected via a camera set up in a live environment. In [54] Woznowski et. al. present complete activities of daily living hierarchical ontology with two video annotation

[2]Note that both action recognition and action classification refer to categorizing trimmed videos.

strategies based on granularity, i.e., atomic labels and high-level annotations for human action recognition in healthcare.

On action recognition for eating behaviors, Sharma et al. [55] presented a convolutional neural network (CNN)-based method for detecting hand-to-mouth gestures during extended periods, ranging from 0.5 to 15 minutes, to identify eating periods. The researchers utilized prior knowledge of other gestures to improve the detection of eating episodes. The Clemson all-day dataset was used, which contains data collected using IMU sensors placed around the subject's wrists. Okamoto et al. [56] presented a system for recognizing eating and drinking actions, which also categorizes the food items consumed. The system detects the mouth region to extract relevant information about nutritional intake.

However, the techniques related to healthcare primarily distinguish between different abnormalities instead of detecting minor changes in a before-and-after scenario. Furthermore, most of the previous techniques have a limitation in that they employ deep features to differentiate between two abnormalities instead of using explainable features. The use of explainable features could be more advantageous for healthcare professionals to understand the underlying causes of abnormalities. Additionally, explainable features could still be somewhat dependable in cases where the algorithm struggles to differentiate between two abnormalities.

3.2.2 Temporal Action Localization

The Background Suppression Network (BSN) [57] predicted the score of an action at any time instance and the score of the start and end of that particular action. It generated flexible proposals by keeping the temporal positions that are high for the score of the start and end of an action. However, these proposals were evaluated separately, which completely ignored the global context of the video. Boundary-Matching Network (BMN) [58] on the other hand, aggregated the features of all proposals and evaluated them simultaneously hence keeping the global context of the video intact. Most of the past algorithms including BSN and BMN used an external classifier to predict action categories from video proposals and relied heavily on anchor windows. Recently, ActionFormer [59] was presented, which combined a transformer with a temporal feature pyramid network to get multi-scale features and recognized action categories without explicitly generating action proposals (hence no external classifier) or pre-defined anchor windows.

4 The EatSense Dataset

The motivation for selecting eating for performance monitoring is that we wanted to select an activity that is performed on a regular basis so we can explore behavioral and upper-body movement changes that healthy person goes through in their daily lives. Eating is one of the most common and frequent actions in one's daily routine as compared to any other action that can change over time. Moreover, individuals tend to persist with their eating habits despite encountering minor physical impediments that may affect their mobility. Lastly, we believe eating can be exploited to acquire and evaluate changes in motor movement. As people grow older, their motor movement gradually gets restricted over time which also affects their ability to eat properly.

The main focus of this paper is to introduce a new dataset that addresses several research gaps related to human eating behavior and healthcare applications. The dataset offers a detailed labeling system with up to 16 action subclasses, including short-time actions, and involves the localization of sub-actions in videos with an average of 114.1 sub-actions in 11.5-minute segments. Additionally, the dataset emphasizes human-centric behavior understanding, particularly related to hand gestures and posture during eating. Finally, the dataset allows recognition of decay in motor movement, which is simulated through wrist weights to create small changes in upper-body movements. The dataset can be found at the link provided in the footnote[3].

[3]https://groups.inf.ed.ac.uk/vision/DATASETS/EATSENSE/

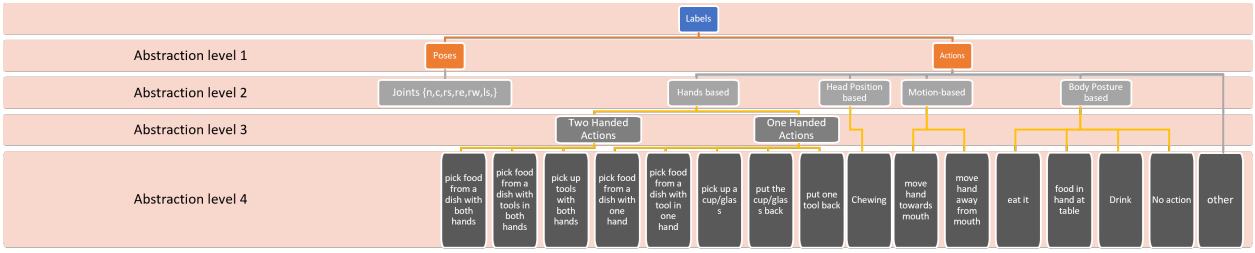

Figure 2: The level of abstractions used in the dataset for labelling each of the 16 actions.

This study focuses only on the research questions mentioned earlier, but there might be other possible uses of the dataset that are not thoroughly discussed in this paper and could be investigated further for understanding human eating behavior and healthcare applications.

4.1 Data Collection

An RGB-D Intel RealSense D415 camera was deployed in a dining room environment. This is an inexpensive depth camera that provides good 3D depth estimation in an indoor environment [60]. The depth maps were used to translate 2D poses from 2D to a 3D frame of reference. The camera was mounted at an oblique angular view facing the dining table, with the restriction that there is only one person in each frame. The recording was done in multiple locations with varying camera-to-subject distances and backgrounds. The frames were discarded if someone other than the person eating walked by or entered the field of view of the camera. The subjects were allowed to eat as they please without any interference or input from the recording team. Fig. 1 shows the setting of the camera system in the dining room environment. It also shows one sample from the dataset for both with and without wrist weights. Special wrist/ankle weights with velcro stitches were used. The weights were always attached to the wrists of subjects, which are denoted as joints numbered 5 and 8 in Fig. 1. The placement is also shown in the bottom thumbnail at the left of the figure.

4.2 Data Labelling

The dataset is labelled with four levels of abstraction.

4.2.1 Data Labelling: Poses

For the first level of abstraction, the skeleton of the upper-body pose was estimated. The 3D joint locations of 8 joints are represented as - nose (1), chest (2), right shoulder (3), right elbow (4), right wrist (5), left shoulder (6), left elbow (7) and left wrist (8). Sometimes both left and right limbs are denoted as one entity, e.g., the right and left wrist collectively is represented as w. HigherHRNet [61] is used to estimate the location of the 2D joints, which were then projected into 3D space by using the depth map measurement. The choice of HigherHRNet was made empirically from a pool of commonly used pose estimators for ground truthing the data.

As pose estimation is a crucial step in skeleton-based action recognition, an experiment to find the most suitable algorithm for the proposed dataset was conducted. First, a total of 100 images were sampled uniformly from the set of videos. Second, 2D poses in the images were then carefully labeled by hand. Third, the images were then used as input to deep learning-based pose estimation algorithms, i.e., OpenPose [62], darkpose [63], deeppose [64], higherHRNet [61] and vipnas [65]. To evaluate the results, two metrics were utilized: mean average precision (mAP) in 2D space and mean squared error (MSE) in 3D space. For

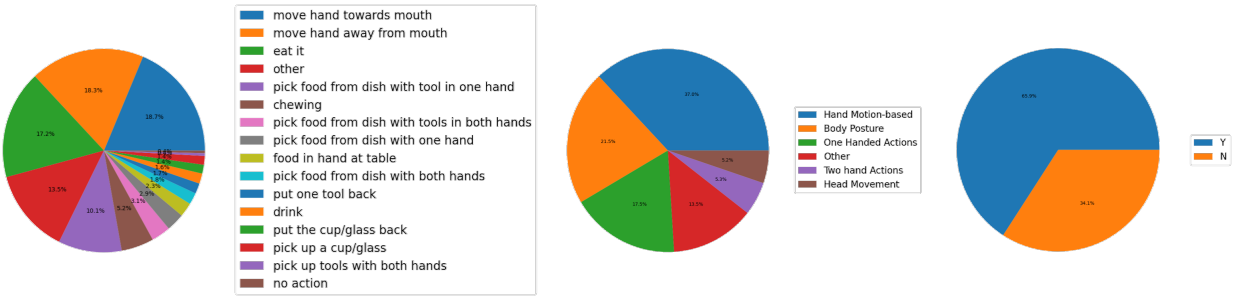

Figure 3: Distribution (in percentages) of various labels according to their occurrence in the dataset. Left) distribution of individual 16 sub-actions. Middle) distribution of actions based on abstraction-level 1 for the labels. Right) occurrence percentage of videos with weights 'Y' and 'N'.

mAP, the intersection over union (IoU) also known as object key-point similarity (OKS) in the case of key points, was calculated by measuring the distance between the predicted and ground truth key points using the Gaussian kernel. The dataset contains scenarios where arms cross each other or the subject puts their hand on their lap (under the table), thus the body joints are sometimes occluded. The experiments showed that when a joint was not visible in the camera, OpenPose was unable to detect the pose of the individual.

Both the 2D joint locations predicted by each of the classifiers and manually labeled 2D joint locations were then projected into the 3D space using the depth map. To quantify error on common grounds, MSE (3D) was calculated with only the visible joints. Table 1 shows the mAP with the threshold of IoU [0.5,1]. It also depicts the MSE for each of the pose estimators. In the experiments that follow, we chose to use HigherHRNet because its MSE was considerably lower (9.7×10^{-3} m) than the alternatives for approximately the same mAP.

Table 1: mAP (@IoU=0.50) and MSE (3D) of the skeleton estimation as compared to hand-labelled ground-truth skeletons.

Algorithm	mAP	MSE (m)
OpenPose [62]	23.4	5.7e-2
Deeppose [64]	59.6	1.18e-2
Darkpose [63]	64.6	1.79e-2
Vipnas [65]	63.0	1.6e-2
HigherHRNet [61]	63.1	9.7e-3

4.2.2 Data Labelling: Actions

For the second level of abstraction, the eating actions were broadly divided into five categories based on joint location and motion. The categories are hands-based, motion-based, head position-based, body posture based and others.

In the third level of abstraction, each of these categories is further divided into sub-actions, that include several atomic actions (that last for less than or equal to a second). Our dataset approximately follows the Zipf law as can be seen in Fig. 3 left. Experiments on actions with few examples can be unreliable, thus we include and present experiments only on the actions that have at least 40 instances. The levels of

abstractions for actions are shown in figure 2. Ground truthing was done manually with the help of a video image annotator (VIA) [66]. For sub-actions where two possible labels were correct, the preference was given to actions being performed by hand. For example, for simultaneous actions '*chewing*' and '*food in hand at table*', '*food in hand at table*' was given preference. Other examples include, '*chewing*' plus '*move hand away from mouth*' was marked as '*move hand away from mouth*'. Moreover, when a person was talking, using a mobile phone or any other activity not included in the list, it was marked as '*others*', irrespective of any eating action going on simultaneously. For example, '*talking*','*chewing*' or '*food in hand at table*' are all marked as '*others*'. Also, whenever everything is at rest, frames were marked as '*no action*'.

To imitate limited motor movements, the researchers captured at least two sets (up to 4 sets) of videos for each participant. In the first set, a weight was tied to each of the participant's wrists to restrict their movements[4], while in the second set, the participant was not weighed down and could move normally. The videos were labeled as 'Y' or 'N' depending on whether the weight was added or not, respectively.

Numerous volunteers actively participated in the labeling process, presenting a significant challenge in maintaining consistency across the labels. To address this concern, we devised a two-step quality control system aimed at achieving reasonably accurate labeling. Initially, we instructed the volunteers to label eating actions using a comprehensive guide that outlined naming conventions for actions and provided detailed explanations on the appropriate usage of each specific label. Subsequently, one of the authors diligently reviewed the labeled videos to ensure the consistent quality of the labels was upheld.

4.3 Data Statistics

The dataset contains 135 video sequences (53 without weights and 82 with weights) of 27 subjects with different cultural backgrounds to ensure diversity in ages, ethnicity, body size, gender, and eating behaviors. These were recorded with a resolution of 640x480 at 15 frames per second (fps). Actions performed in each of the individual videos are shown at the top fig. 4. Table 2 summarizes the average time taken by an instance of each action and the total number of instances of the actions. Moreover, for quality of motion assessment, the ratio of occurrence of non-weight 'N' is 36.6% and weight 'Y' is 63.4%. The percentage distribution of different levels of label abstractions is shown in Fig. 3.

Eating behaviors and food choices vary across different cultures and regions. Chopsticks are commonly used for eating in East Asian countries, while hand-to-mouth eating with no tool is prevalent in South Asian regions, where people prefer to eat rice or flatbreads directly with their hands. The subjects were chosen specifically to maximize diversity and generalizability. The EatSense dataset includes subjects from thirteen different nationalities, and Table 3 provides information on their ethnicity by region, age groups, and the tools they used for eating. As the choice of tools is dissimilar between different subjects, the actions performed are also bound to be subjective. The dataset also conforms to the above-mentioned convention, as can be visibly seen at the bottom fig. 4.

4.4 Dataset Properties

EatSense has several attractive properties that distinguish it from other existing datasets.

Each of these videos is densely labeled, which means there are no unlabelled temporal patches unlike most of the existing large-scale datasets. Also, the two-stage quality control of labels ensures clean and consistent labels across all of the videos. The current state of the dataset can be easily extended by recognizing tools and foods and checking for human-object interactions, to detect what types of food a person eats for a complete nutritional analysis. Unlike other existing datasets, where a background/environment plays a significant

[4]Adding weights is not intended to be a model for aging or a neurological disorder, but to demonstrate that changes in motion performance can be detected.

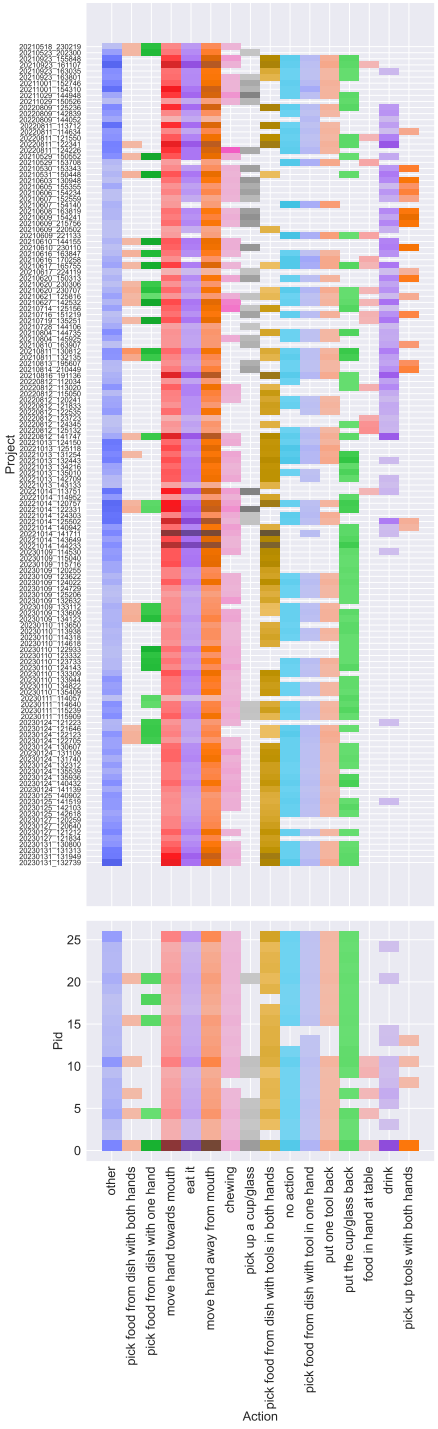

Figure 4: Top) Actions performed in each of the individual videos. The vertical axis shows the name of each of the individual videos, which has the format '$\{date\}\_\{unix-time\}$', collectively marked by the keyword 'Project' in the dataset. Bottom) shows the actions performed by individual subjects. The variations in the color mean the frequency of occurrence of each action. It has subject IDs (Pid) on the vertical axis and actions on the horizontal axis. Vectorized image, best viewed zoomed in.

Table 2: Average time in seconds taken by an instance of the action and total number of instances of the action for each of the actions in the EatSense dataset

Actions	Instances	
	total no.	avg. time
chewing	795	6.165
drink	247	2.723
eat it	2630	0.717
food in hand at table	344	3.868
move hand away from mouth	2792	0.625
move hand towards mouth	2851	0.844
no action	64	9.007
other	2057	7.043
pick food from dish with both hands	282	5.342
pick food from dish with one hand	440	3.741
pick food from dish with tool in one hand	1548	3.943
pick food from dish with tools in both hands	467	6.449
pick up a cup/glass	213	1.218
pick up tools with both hands	65	1.880
put one tool back	253	1.067
put the cup/glass back	214	1.618

Table 3: The table shows the diversity of subjects divided into five age groups. This shows the tools, foods, and ethnical origins of all 27 subjects involved in the dataset. Pid refers to person IDs.

Age-Groups	Pids	Genders	Tools	Foods	Ethnicity
below 30	2, 3, 4, 5, 6, 9, 10, 11, 12, 13	8M, 3F	Fork and Spoon, Fork, Spoon, No tool, Fork and Knife	Rice, Noodles, Soup, Shewarma, Apple, Toast, Only-Drinks, Roti, Egg, Steak, Sandwich, Pizza, Salad	South Asian, East Asian, British
30-39	7, 8, 15, 23, 25, 27	4M, 2F	No tool, Spoon	Shewarma, Rice	East Asian, European
40-49	19, 21, 22	1M,2F	Fork and Spoon, Fork and Knife	Rice	South-American, British
50-59	17, 25, 26	3F	Fork and Spoon, Spoon	Rice	British
Above 60	1, 16, 18, 20	3M,1F	Fork, No tool, Spoon, Fork and Knife	Rice, Roti, Soup, Wafers, Steak, Toast	American, British

role in distinguishing between different actions, EatSense has relatively consistent backgrounds and human posture-centric action instances.

Table 4: Brief comparison of the proposed EatSense dataset against publicly available datasets used in action recognition, localization and healthcare research. C# stands for the number of classes (action classes, anomaly classes in case of Init Gait DB), BUC stands for human behavior understanding capability, HCC stands for healthcare capability, UV stands for the untrimmed videos, S# stands for the number of subjects which is marked multiple(M) for datasets lacking specific numbers, Lbs indicates the type of labels single (S), sparse, (Sp) and dense (D). TC stands for targetted community, i.e., computer vision (CV) and healthcare (HC). Settings refer to a controlled (C) or uncontrolled (UC) environment for recording. The last column indicates the average number of actions present and the average video duration. N/A denotes that the required statistic wasn't available in the manuscript and the authors didn't respond.

Datasets	C #	BUC	HCC	UV	S#	Setting	Lbs	TC	Avg. # act.	Avg. vid. dur.
Epic-Kitchens-100 [20]	4053	✗	✗	✓	M	UC	D	CV	128.5	8.5 m
NTU-RGB-D [41]	120	✗	✗	✗	106	UC	S	CV	1	7.21 s
Kinetics [10]	700	✗	✗	✗	M	UC	S	CV	1	10 s
HMDB [12]	51	✗	✗	✓	M	UC	S	CV	1	∼2 s
J-HMDB [13]	21	✗	✗	✓	M	UC	S	CV	1	∼2 s
Something-Something-v2 [11]	174	✗	✗	✗	> 1300	C	S	CV	1	4.03 s
PKU-MMD (P1) [16]	51	✗	✗	✓	66	UC	D	CV	20	3∼4 m
PKU-MMD (P2) [16]	49	✗	✗	✓	13	UC	D	CV	7	1∼2 m
Activity-Net 1.3 [14]	200	✗	✓	✓	M	UC	Sp	CV	1.54	1.9 m
THUMOS14 [67]	20	✗	✗	✓	M	UC	D	CV	15.4	1.1 m
UCF-101-24 [19]	101	✗	✗	✓	M	UC	Sp	CV	1.4	5.1 s
FineGym [15]	530	✗	✗	✓	M	UC	Sp	CV	42	2h
FineDiving [3]	52	✗	✗	✓	M	UC	Sp	CV	3.26	6.9^5 s
AVA [17]	80	✗	✗	✓	M	UC	D	CV	1380	15 m
MSR-DailyActivity [32]	16	✗	✓	✗	10	C	S	CV, HC	1	6 s
Sphere H-130 [18]	13	✗	✓	✓	5	C	D	CV, HC	13	5 m
Mobiserv-AIIA [30]	13	✗	✓	✓	12	C	Sp	CV, HC	N/A	N/A
Init Gait DB [35]	7	✗	✓	✗	10	C	S	HC	N/A	N/A
OREBA [25]	2	✓	✓	✗	100	UC	S	CV, HC	N/A	N/A
CCD [21]	5	✓	✓	✗	264	UC	S	CV, HC	N/A	N/A
EatSense (ours)	16	✓	✓	✓	27	UC	D	CV, HC	114.1	11.5m

As discussed before, EatSense contains ground truth for multiple levels of abstraction. Each of the actions is chosen by a tree-like labeling strategy, based on figure 2. Frame-level labels of actions and poses are provided. When a person eats while sitting at a dining table, their lower body is occluded and does not play any significant part in detecting eating activities. Hence, EatSense is an upper-body posture-focused dataset. EatSense is also rich in data that could be used for human-health analysis. For example, it contains a layer of labels that simulate decay/decline in the motor movement of a person over time. Continuous monitoring of eating actions and looking for decay in motor movement could potentially be helpful for identifying a serious health situation.

Table 4 shows a brief comparison of how EatSense compares to other related datasets in both the computer vision and AI healthcare communities. The table includes commonly used action recognition/localization datasets, such as Thumos14 [67], FineGym, NTU-RGB-D, and AVA alongside commonly used healthcare-based datasets, for example, MSRDailyActivity3D, Init Gait DB and OREBA. The table presents various

characteristics of the EatSense dataset that offer multiple possibilities for research. These include the ability to develop models for action recognition, action localization with dense labels, eating behavior analysis, decay in motor movement assessment, upper-body-focused models, etc.

5 Experimental Evaluation on Baseline Approaches

We evaluate EatSense using two action understanding methodologies, i.e., action recognition and temporal action localization methods. The PyTorch implementation was utilized for each of the algorithms listed below. Each algorithm was trained for 150 epochs, beginning with a learning rate of 1×10^{-2}, and for every 30^{th} epoch, the learning rate was multiplied by a factor of $\frac{1}{10}$. Unless specified otherwise, the rest of the training protocols for the techniques used were consistent with those in the original papers.

5.1 Trimmed Video Analysis

The analysis of trimmed videos can be divided into two stages: single-modality experiments, which examine individual aspects such as skeleton, flow, and RGB, and multi-modality experiments, which combine RGB and flow. In trimmed video analysis, it is assumed that the videos have already been segmented into separate clips for each action. These techniques explore intra-action relationships independent of the past or future action occurrence, hence only recognizing the ongoing action. Trimmed video analysis avoids the problem of deciding when one action finishes and the next starts.

5.1.1 Dataset Splits

To generate a mix of actions for a classification analysis, first, the data was divided into clips of individual activities. Second, stratified sampling (without replacement) on the action clips was done. Third, using these sampled clips, five stratified splits were generated. Out of these five splits, three splits were used for training and one each for validation and testing for this recognition task. The splits were permuted for five-fold cross-validation. For evaluation on graph-connected networks that take poses as the input, a set of poses ($m \times 8 \times 3$ vector) for each frame, was also shuffled with the constraint that the same set of actions is selected as in the original five splits, where m is the number of frames.

5.1.2 Classifiers

Various deep learning-based networks with different input modalities were evaluated for the classification of the trimmed videos. For skeleton-based classifiers, graph-based approaches with deep features such as graph-convolutional networks (e.g., CTR-GCN [49], 2s-AGCN [48], ST-GCN [45]) and 3D heatmap volume (e.g., PoseConv3D [44]) were utilized for evaluation of both 2D and 3D joint information. We also demonstrate recognition from RGB, optical-flow (motion features) and combined RGB+Flow modalities by using TANet [52], TPN [51] and TSN [50]. Furthermore, a comparison was conducted between fine-tuning pre-trained algorithms and training the same algorithms from scratch.

5.1.3 Results

To measure the performance of the models on EatSense, we compute the Top-1 and macro (mean class accuracy) over all 16 classes. Table 5 shows the top-1 (clip) and macro (class) accuracies of the networks with pre-trained models and table 6 demonstrates the performance when trained from scratch.

[5] assuming fps=15. Fps is not mentioned in the FineDiving paper.

Table 5: The table displays the Top-1 and Macro accuracies achieved by deep networks with three modalities as input on the trimmed videos. The 'Pre-train dataset' column indicates the dataset on which the particular algorithm was pre-trained. NTU-60 and Kin-400 refer to NTU-RGB-D-60 and kinetics-400 respectively. The row 'Average' shows the average accuracy achieved by all the tested algorithms on their respective modalities.

Algorithm	Pre-Train Dataset	Modality	Top-1 Acc.	Macro Acc.
CTR-GCN 2D [49]	NTU-60	Pose	85.1	71.2
PoseConv3D [44]	NTU-60	Pose	79.1	54.9
2s-AGCN [48]	NTU-60	Pose	82.6	66.3
ST-GCN 2D [45]	NTU-60	Pose	67.4	38
ST-GCN 3D [45]	**NTU-60**	**Pose**	**89.8**	**71.9**
Average		Pose	80.8	60.5
TANet [52]	**Kin-400**	**RGB**	**87.5**	**80.5**
TPN [51]	Kin-400	RGB	87.4	79.8
TSN [50]	Kin-400	RGB	83.3	70.0
Average		RGB	86.1	77.0
TANet [52]	Kin-400	Flow	84.5	**72.6**
TPN [51]	Kin-400	Flow	82.9	66.4
TSN [50]	**Kin-400**	**Flow**	**87.2**	72.3
Average		Flow	84.8	70.5
TANet [52]	Kin-400	RGB+Flow	88.3	80.1
TPN [51]	Kin-400	RGB+Flow	88.5	81.5
TSN [50]	**Kin-400**	**RGB+Flow**	**90.2**	**82.9**
Average		RGB+Flow	89.0	81.5

5.1.4 Discussion

On trimmed action clips as shown in table 5 with pose as the modality, CTR-GCN performs well as compared to others because it utilizes a channel-wise topology, i.e., it dynamically learns and effectively aggregates features. ST-GCN 2D does not perform well as 2D poses have low-quality motion features compared to 3D. Claiming the first position, ST-GCN 3D performs significantly better regarding both top-1 and macro accuracy. Overall, almost all deep learning-based graph convolutional networks (GCN) tend to achieve an accuracy of over 70% and class-wise accuracy of over 50%. Moreover, when training from scratch (as shown in table 6), ST-GCN 3D still performs better than others.

For RGB as a modality, both trained from scratch and pre-trained on Kinetics-400, TANet and TPN achieve nearly the same top-1 and macro accuracy with a slight difference in the performance. However, TANet (when trained from scratch) achieves the best classwise accuracy (macro) of 72.6% and top-1 83.6% because it specializes in capturing long-term temporal dependencies, unlike TSN and TPN. TANet, however, does not perform as well with the optical flow as input. This may be related to how optical flow encodes motion information in the video. Optical flow is a dense representation of motion, whereas the encoding of the motion information in RGB frames is more sparse and may be more difficult to extract.

On the other hand, TSN is designed to explicitly model temporal information by dividing the video into segments and aggregating features from each segment. This makes it well-suited for processing dense motion information like optical flow. Hence, TSN provides good performance for optical flow input as compared to

Table 6: The table displays the Top-1 and Macro accuracies achieved by deep networks with three modalities as input on the trimmed videos. 'None' represents the baseline algorithms that were trained from scratch. The row 'Average' shows the average accuracy achieved by all the tested algorithms on their respective modalities.

Algorithm	Pre-Train Dataset	Modality	Top-1 Acc.	Macro Acc.
CTR-GCN [49]	None	Pose	84.9	71.1
PoseConv3D [44]	None	Pose	79.2	56.2
2s-AGCN [48]	None	Pose	83.6	65.6
ST-GCN 2D [45]	None	Pose	45.7	25.5
ST-GCN 3D [45]	**None**	**Pose**	**90.1**	**77.9**
Average		Pose	76.7	59.2
TANet [52]	**None**	**RGB**	**83.6**	**72.6**
TPN [51]	None	RGB	83.5	68.5
TSN [50]	None	RGB	82.4	61.8
Average		RGB	83.1	67.6
TANet [52]	None	Flow	82.9	**68.5**
TPN [51]	None	Flow	81.7	64.8
TSN [50]	**None**	**Flow**	**83.9**	67.5
Average		Flow	82.8	66.9
TANet [52]	None	RGB+Flow	85.7	71.7
TPN [51]	None	RGB+Flow	86.2	74.9
TSN [50]	**None**	**RGB+Flow**	**88.7**	**79**
Average		RGB+Flow	86.8	75.2

TANet or TPN. Additionally, TSN uses a multi-scale temporal sampling strategy that allows it to capture temporal information at multiple scales. This may be particularly beneficial for processing optical flow, which encodes motion information at multiple spatial and temporal scales.

Lastly, as shown in table 5, for mixed modality (RGB+Flow) the top-1 accuracy achieved by all three algorithms is nearly the same. Overall, TSN performs better in terms of top-1 accuracy and macro accuracy because it has the capacity to effectively capture the temporal evolution of actions by dividing the clip into short segments and sampling frames from each of the segments.

Table 6 shows the performance of the above-mentioned algorithms when trained from scratch. In the table, TANet demonstrates superiority in RGB modality, while TSN excels in optical flow. Overall, the experimental findings indicate that the baseline algorithms exhibit similar accuracy levels, even when trained from scratch. However, as can be observed by the averages, mentioned in the tables 5 and 6, utilizing pre-trained models enhance the accuracies achieved by the baseline algorithms for all the modalities.

5.2 Untrimmed Video Analysis

One of the main contributions of EatSense is its dense labeling, i.e., there are no unlabelled patches in the 11.5-minute (on average) long videos. This gives the opportunity to develop skeleton-based (pose-based) action localization frameworks. As there are no off-the-shelf implementations of such a technique at present, we provide evaluation benchmarks on RGB images. We present the evaluation of EatSense on untrimmed

videos using temporal action localization algorithms that exploit images to study inter-action temporal relationships.

5.2.1 Dataset Splits

For untrimmed videos, the videos were randomly divided into three splits, training on 96, validation on 24 and testing on 15 videos.

5.2.2 Localizers

In action localization frameworks, videos (usually in non-overlapping snippets) are input into a (pre-trained) visual encoder that represents the video as a high-level feature set, that is further processed for action recognition and localization. For temporal localization (deciding when the video stream changes from one activity to another) tasks, we extracted visually encoded features using the TSN [50] and TSP [68] pipelines. For EatSense, we used overlapping snippets to get the dense high-level feature set to detect atomic actions. These high-level features were then used with BMN [58] and ActionFormer [59], respectively, to generate segment proposals for evaluation using the EatSense dataset. In the end, the proposals generated by BMN were then classified using TSN (trained and evaluated as in section 5.1.2) whereas ActionFormer implicitly categorizes the generated proposals into action classes.

5.2.3 Results

For the performance evaluation on EatSense, mean average precision is used as a metric for action localization tasks. The results are shown in table 7, where @0.1, @0.3 and @0.5 denote the temporal IoU (tIoU) threshold levels.

Table 7: The table shows the mean average precision for the action localization task using deep networks on untrimmed videos.

Algorithm	Feature	Mean Average Precision			
		@0.1	@0.3	@0.5	Avg.
ActionFormer [59]	TSP	**14.04**	**7.91**	**3.43**	**8.46**
BMN [58] + TSN	TSN	2.27	1.12	0.60	1.33

5.2.4 Discussion

Untrimmed videos in EatSense are particularly challenging for current action localization networks as can be seen from the table 7. The performance of localization algorithms decays with respect to higher tIoUs. This is because EatSense contains actions of lengths varying in the range of [0.62,9] seconds, i.e., the smallest atomic action lasts for 9.3 frames on average. On the other hand, we only provide benchmarks for action localization on RGB data as (to the best of our knowledge) there aren't any specialized off-the-shelf skeleton-based temporal action localization frameworks available.

6 Applications with Explainable Features

EatSense targets a wide array of applications as highlighted previously in section 4. It allows atomic/sub-action recognition and localization and provides the capability to study minor changes in motion to determine

decay in the motor movement of the upper body of a person, etc. In this section, we also explore some major application scenarios of EatSense to demonstrate the flexibility of the dataset in understanding the role of individual joints for action recognition and deterioration assessment. For this purpose, we use hand-crafted features derived from the 8 upper body joints which are discussed in the following sub-sections.

6.1 Descriptive Features

After 2D human pose estimation, using HigherHRNet, depth maps recorded by the RGBD camera were used to project 2D poses to 3D space. The estimated 2D poses and projected 3D points are absolute positions and they tend to change whenever the environment or position of the camera changes. To avoid this issue, we set the origin of the camera coordinate system to the 3D location of the subject's chest joint. We calculated the relative positions of each joint by subtracting the 3D position of the chest from the 3D joint absolute position, as given in eq. 1, where $\vec{p}_j^{abs}$ refers to the absolute 3D position of the joint j, where $j = \{1, ..., 8\}$ and $\vec{p}_c^{abs}$ refers to the absolute position of the chest. The relative joint positions are then used as a feature for classification tasks.

$$\vec{p}_j^{rel} = \vec{p}_j^{abs} - \vec{p}_c^{abs} \tag{1}$$

6.1.1 Additional Spatial Features

Motion of the joints while performing various eating sub-actions is highly correlated. To exploit the relations between both arms, the Euclidean distances between both wrists and both elbows were estimated as given in eq. 2, where i represents either the elbow or wrist joint and r or l denotes right or the left one, respectively. For a more meaningful representation of a 3D point in space, the joints' relative positions (note that this gives the distance of the joint from the chest) are converted into a polar coordinate system as given in eq. 3. To explore the inherent long-range dependencies of the joints, i.e., the stretch and contraction of the arms, the product of the polar coordinates of the joints were calculated as given in eq. 4 where p_j^{polar} represents the relative polar joint position excluding the chest joint (skeleton origin).

$$d_{a,i} = ||\vec{p}_{r,i}^{rel} - \vec{p}_{l,i}^{rel}||_2 \tag{2}$$

$$p_j^{polar} = \sqrt{(p_{x,j}^{rel})^2 + (p_{y,j}^{rel})^2 + (p_{z,j}^{rel})^2} \tag{3}$$

$$p^{prod} = \prod_{j \in \{j=1,...,7 \; and \; j \neq c\}} p_j^{polar} \tag{4}$$

To get the orientation of the arms at any time instance, joint triplet angles for elbow (e) and shoulder (s) sockets were calculated (w and c represent wrists and body centre in the equations).

$$\begin{aligned} p_e^{\theta} &= cos^{-1} \frac{(\vec{p}_s^{rel} - \vec{p}_e^{rel}) \cdot (\vec{p}_w^{rel} - \vec{p}_e^{rel})}{\|\vec{p}_s^{rel} - \vec{p}_e^{rel}\| \times \|\vec{p}_e^{rel} - \vec{p}_w^{rel}\|} \\ p_s^{\theta} &= cos^{-1} \frac{(\vec{p}_c^{rel} - \vec{p}_s^{rel}) \cdot (\vec{p}_e^{rel} - \vec{p}_s^{rel})}{\|\vec{p}_c^{rel} - \vec{p}_s^{rel}\| \times \|\vec{p}_s^{rel} - \vec{p}_e^{rel}\|} \end{aligned} \tag{5}$$

Lastly, to exploit the interactions of the human posture with stationary objects in the scene, we find the distance of the joints from the plane of the table. The plane of the table was estimated using a least squares method from 3D table locations. These pixel locations were marked manually on the table in the first frame in each of the videos and then propagated through the video with the assumption that the table is fixed and does not change position during one eating session.

6.1.2 Additional Temporal Features

Temporal features are imperative for robust action recognition, but these features become pivotal if motion quantification is needed. In the temporal domain, velocities and acceleration of the joints were estimated, as given in eq. 6 and 7, respectively. Subscripts $t+k$ represent joint (j) positions at frame $t+k$, where k controls the number of frames in the temporal estimation window. Due to the real-world recording environment and no control over the subject, performance or the order of actions, these measurements were found to be particularly noisy. To accommodate this, features with varying window sizes, i.e., using k from 0 to 5 for both acceleration and velocities, were included in the feature space.

$$\vec{v}_j = \vec{p}^{rel}_{t+k+1,j} - \vec{p}^{rel}_{t-k-1,j} \tag{6}$$

$$\vec{a}_j = \vec{p}^{rel}_{t+k+2,j} + \vec{p}^{rel}_{t-k-2,j} - 2\vec{p}^{rel}_{t,j} \tag{7}$$

To put emphasis on the causality of the current posture, it is important to look for the immediate past values. For this purpose, we also use three lag relative positions (i.e., $\vec{p}^{rel}_{t-1,j}, \vec{p}^{rel}_{t-2,j}, \vec{p}^{rel}_{t-3,j}$). Furthermore, a weighted sum over the three lags was also included as a feature. The weights of the past three values were set to put more stress on recent past values. The weights and the moving window equation are given in eq. 8. The *movw* stands for moving window and the subscript shows the size of that window.

$$\vec{p}^{movw}_{3,j} = 0.5 \times \vec{p}^{rel}_{t-1,j} + 0.35 \times \vec{p}^{rel}_{t-2,j} + 0.15 \times \vec{p}^{rel}_{t-3,j} \tag{8}$$

6.2 Classification with Hand Crafted Features

This section investigates eating sub-action recognition with explainable (hand-crafted) features in a frame-by-frame setting. The features discussed in the previous paragraphs represent both the spatial and temporal domains of the skeleton. There are many features, and not all are useful. A feature selection process was applied (described in Section 6.2.3). The selected features were represented as vectors, which were the input to the various classifiers described below. The features were then used to classify the labeled 16 different actions in EatSense.

6.2.1 Dataset Splits

For classification using hand-crafted features, we used the same strategy of stratified sampling of the data as mentioned in subsection 5.1.1. Clips in the whole dataset were stratified sampled (without replacement) into 5 subsets of the data splits while maintaining nearly the same percentage of occurrence of each individual action in each subset. However, these clips were then expanded into individual frames for frame-by-frame analysis.

6.2.2 Classifiers

For classification using videos and hand-crafted features, various machine/deep learning methods were explored including Light Gradient Boosting Method (LightGBM) [69], Adaboost, Multi-layer Perceptron[6] (MLP), K-Nearest Neighbour (KNN) and Quadratic Discriminant Analysis (QDA). Additionally, as our data has an imbalanced set of classes, LightGBM with the focal-loss objective function was also tested. Default values of the hyper-parameters of the techniques mentioned above were used unless stated otherwise.

[6]Input layer of size 30, four fully connected hidden layers with 50,75,45,25 neurons followed by RELU and batch-normalization respectively. Finally, the output layer is size 16.

6.2.3 Experiments

To deal with the numerous features of the data, we employed a forward sequential feature selection search to identify the most influential features using stratified splits (all five splits) with five-fold cross-validation. The top 30 features from the feature selector are presented in Fig. 5. From the results, we discovered that 30 features are the most helpful for classification, and decided to use them for further experiments. For frame-by-frame analysis, two experiments were carried out. (i) Training and testing on stratified splits (5-fold cross-validation). (ii) Training on stratified splits and testing on unseen full videos.

6.2.4 Results

Table 8 shows the results of both experiments mentioned above. It shows the mean top-1 and standard deviation of the accuracy achieved by each of the algorithms with 5-fold cross-validation in the first three columns. The second experiment (shown in the last two columns) shows the accuracy of the trained model results on the unseen videos. It is clear that LightGBM outperforms the others in both experiments. As the action ground truth is labelled manually, the start and end frame times of the actions are prone to be noisy. To account for human variation and overcome the temporal classifier offset in the labels, a windowed search of sizes ranging from ±1 to ±45 frames (3 seconds) was applied to look for the correct label in that particular range of frames, between the ground truth and the predictions. Table 8 shows the accuracy achieved with the window size ±1 (which corresponds to ± 0.2 seconds).

Table 8: Raw frame-wise activity classification accuracy of testing on stratified splits and on unseen full videos. The stratified splits results show the mean and standard deviation of the results over the 5 test splits. Top-1, Macro and Std. show the frame-wise mean classification accuracy, mean classwise balanced accuracy and the mean standard deviation (post-5-fold CV). UV and FL stand for unseen videos and focal loss respectively.

Algorithm	5-fold CV			Unseen Accuracy	
	Top-1	Macro	Std.	UV	±1
MLP	67.7	47.7	1.3	22.2	31.3
N. Neighbors	59.5	43.1	1.2	32.6	42.1
Decision Tree	50.8	24.4	**0.7**	20.8	24.7
Random Forest	42.3	7.6	3.3	14.3	16.9
Neural Net	42.4	7.4	2.0	11.9	12.2
AdaBoost	47.3	24.4	1.7	25.7	34.2
Naive Bayes	25.7	17.2	1.4	7.3	11.7
QDA	27.5	23.1	3.3	26.2	38.2
LGBM	67.5	47.6	1.1	36.6	44.6
LGBM (FL)	**69.5**	**48.8**	0.9	**38.9**	**44.9**

6.3 Quality of Motion Assessment

As an initial proof-of-concept for detecting minor changes in motion, we investigate a binary classification problem. EatSense provides labels 'Y' and 'N' for the classification of people eating with and without weights {with weights ('Y'), without weights ('N')} on the wrist, respectively. This could potentially simulate decay in the motor movement of the elderly over time.

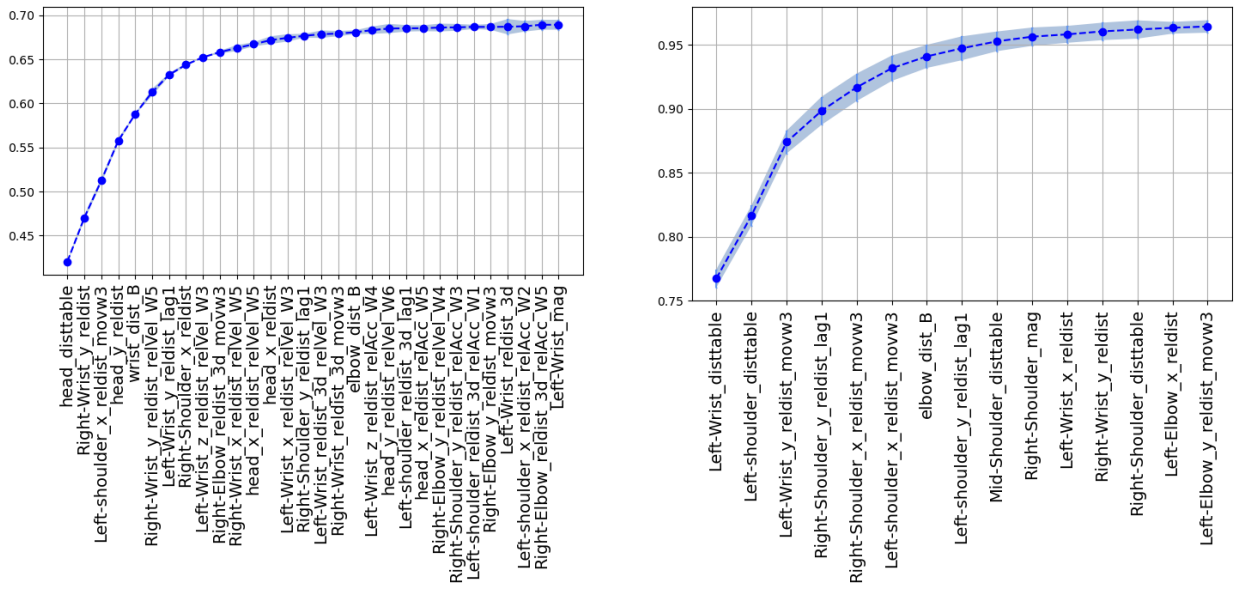

Figure 5: The left and the right figures are the forward sequential feature selection plots for both action recognition classification and motion quality assessment. The vertical axis shows the corresponding accuracy as features are added to the data. The horizontal axis lists the features as they are added to the classification process. The shaded region shows the error bounds of accuracy determined by cross-validation on different sets.

6.3.1 Dataset

On stratified splits (all five splits), a forward sequential feature selector with five-fold cross-validation was used to identify the most contributing features for this classification. The feature selection plot for only the top 15 features is shown in Fig. 5. The top 10 features were found to be of the greatest importance.

6.3.2 Classifier and Experiments

As lightGBM outperforms other algorithms in frame-by-frame action classification applications with hand-crafted features, thus lightGBM (LGBM) was used to distinguish between with and without weight cases. The experiments on the assessment of the quality of motion using 10 features were divided into two sub-experiments. (i) combined subjects: training and testing on all[7] data in stratified splits (with 5-fold cross-validation). (ii) Cross-Subjects: feature analysis to check how well the best 10 features generalize with respect to individual subjects in terms of separability of 'Y'/'N' classes using the t-SNE plot shown in Fig. 6.

6.3.3 Results

Table 9 shows the results of overall and subject-wise mean accuracy along with standard deviation (experiment (i)). The column 'Mean Acc.' shows the average of the accuracies achieved during 5-fold cross-validation

[7]both train and test splits contain all subjects.

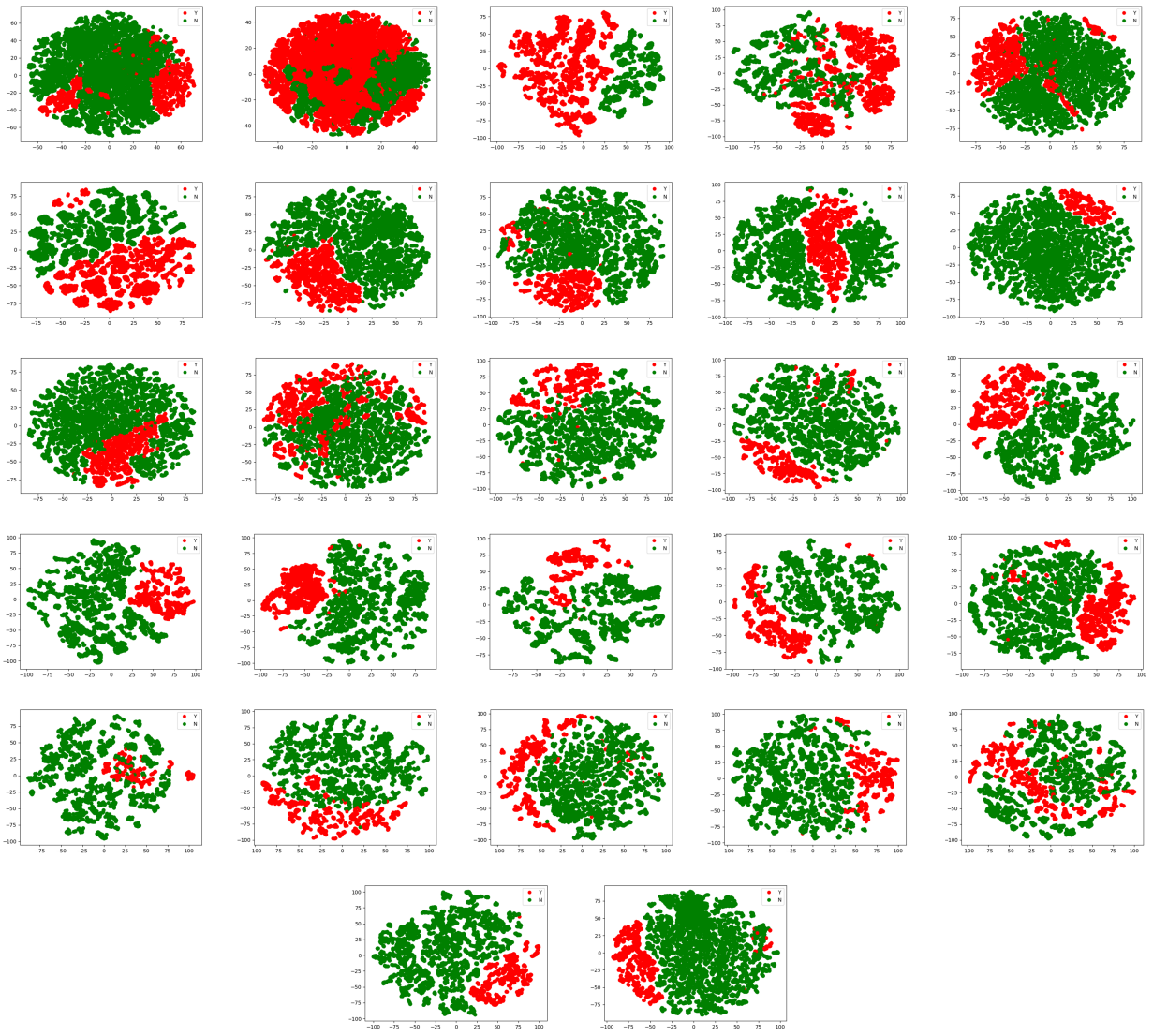

Figure 6: T-SNE plots for individual subjects with 10 features mapped to 2D space after feature selection with all 27 subjects. Green is for no weights ('N'), and red is for weights ('Y') attached to the wrists of the subjects. The first row from left to right, depicts subjects S_1 to S_5 and the rest of the rows are arranged in a similar fashion.

and 'Std.' shows the standard deviation.

Table 9: The mean weight/no weight ('Y'/'N') classification accuracy per subject (from the 5-fold CV) with LightGBM as the classifier. $\{S_1, ..., S_{27}\}$ denote each of the ten subjects.

Subjects	Mean Acc.	Std.
S_1	99.1	0.33
S_2	99.1	0.14
S_3	99.3	0.70
S_4	91.2	5.05
S_5	97.4	1.72
S_6	97.7	1.48
S_7	99.0	0.65
S_8	97.8	1.12
S_9	99.0	0.63
S_{10}	99.5	0.41
S_{11}	97.4	0.86
S_{12}	91.9	6.79
S_{13}	96.5	2.71
S_{14}	98.2	1.26
S_{15}	99.4	0.37
S_{16}	99.6	0.23
S_{17}	99.1	1.31
S_{18}	95.7	2.34
S_{19}	98.2	1.02
S_{20}	98.4	2.54
S_{21}	96.7	4.71
S_{22}	91.4	5.18
S_{23}	96.7	2.09
S_{24}	97.6	1.60
S_{25}	94.3	3.56
S_{26}	99.0	1.68
S_{27}	99.1	0.72
Overall	97.4	1.71

6.4 Discussion

For a frame-by-frame analysis with explainable features, it is observed in Table 8, LightGBM without the focal loss objective function achieves high frame-wise accuracy of 67.5% for unseen video, (44.6% if ± 1 temporal window analysis is applied to account for human variation in ground truth labelling) and overall class-wise (macro) accuracy of 47.6%. However, the use of the focal loss objective function with LightGBM increases the classwise (macro) accuracy by 1.2%.

For quality assessment of motion with combined subjects table 9 shows that LightGBM works very well and achieves overall 97.4% mean accuracy with five-fold cross-validation. Moreover, to validate the generalization of features across multiple subjects, the mapping of the 10 features selected by the forward sequential feature selector to a 2D plane (Fig. 6) using t-SNE clearly shows that both classes are separable

from each other, where green and red (0 and 1) represent 'N' (no weights) and 'Y' (with weights) respectively. This shows a change in motion is certainly detectable with low-level features.

7 Conclusions

This paper presents the new benchmark dataset EatSense that includes atomic actions, dense multiple abstraction levels of frame-level labels and with/without weight cases to simulate deterioration in motor movement. EatSense can be used as a generic training benchmark dataset for action recognition tasks specifically designed for the eating process. Furthermore, EatSense also has the capability to be used as a generic test benchmark suite for temporal action localization and action recognition.

We provide a systematic analysis of the performance of the dataset with many deep learning frameworks on multiple modalities for trimmed videos. We also discuss the performance of temporal action localization on untrimmed videos in EatSense. However, the performance of current temporal action localization algorithms is not very good and EatSense proves to be more challenging than the rest of the datasets publicly available. This highlights the need for developing new approaches for untrimmed video understanding. Furthermore, we demonstrate the application capability of EatSense even where a low-level understanding of the individual joints or hand-crafted features is required.

Future research includes extending the action recognition classes to include actions such as 'wipe mouth' and 'mix'. The deterioration classification currently is for two classes {with weights, without weights}, i.e., decayed and normal. We plan to extend the deterioration classification to a more fine-grained scale.

Ethics approval was obtained for data collection and distribution.

References

[1] M. B. Shaikh, D. Chai, Rgb-d data-based action recognition: A review, Sensors 21 (12) (2021) 4246.

[2] Z. Sun, Q. Ke, H. Rahmani, M. Bennamoun, G. Wang, J. Liu, Human action recognition from various data modalities: A review, IEEE transactions on pattern analysis and machine intelligence.

[3] J. Xu, Y. Rao, X. Yu, G. Chen, J. Zhou, J. Lu, Finediving: A fine-grained dataset for procedure-aware action quality assessment, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 2949–2958.

[4] G. Bertasius, H. Soo Park, S. X. Yu, J. Shi, Am i a baller? basketball performance assessment from first-person videos, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 2177–2185.

[5] Z. Lei, B. Y. Tan, N. P. Garg, L. Li, A. Sidarta, W. T. Ang, An intention prediction based shared control system for point-to-point navigation of a robotic wheelchair, IEEE Robotics and Automation Letters 7 (4) (2022) 8893–8900. doi:10.1109/LRA.2022.3189151.

[6] T. N. Nguyen, H. H. Huynh, J. Meunier, 3d reconstruction with time-of-flight depth camera and multiple mirrors, IEEE Access 6 (2018) 38106–38114. doi:10.1109/ACCESS.2018.2854262.

[7] R. D. Rondinelli, W. Dunn, K. M. Hassanein, C. A. Keesling, S. C. Meredith, T. L. Schulz, N. J. Lawrence, A simulation of hand impairments: effects on upper extremity function and implications toward medical impairment rating and disability determination, Archives of physical medicine and rehabilitation 78 (12) (1997) 1358–1363.

[8] S. Ishikawa, S. Okamoto, K. Isogai, Y. Akiyama, N. Yanagihara, Y. Yamada, Wearable dummy to simulate joint impairment: severity-based assessment of simulated spasticity of knee joint, in: Proceedings of the 2013 IEEE/SICE International Symposium on System Integration, IEEE, 2013, pp. 300–305.

[9] J. Liu, A. Shahroudy, M. Perez, G. Wang, L.-Y. Duan, A. C. Kot, Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding, IEEE transactions on pattern analysis and machine intelligence 42 (10) (2019) 2684–2701.

[10] J. Carreira, E. Noland, C. Hillier, A. Zisserman, A short note on the kinetics-700 human action dataset, arXiv preprint arXiv:1907.06987.

[11] R. Goyal, S. Ebrahimi Kahou, V. Michalski, J. Materzynska, S. Westphal, H. Kim, V. Haenel, I. Fruend, P. Yianilos, M. Mueller-Freitag, et al., The" something something" video database for learning and evaluating visual common sense, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 5842–5850.

[12] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, T. Serre, Hmdb: a large video database for human motion recognition, in: 2011 International conference on computer vision, IEEE, 2011, pp. 2556–2563.

[13] H. Jhuang, J. Gall, S. Zuffi, C. Schmid, M. J. Black, Towards understanding action recognition, in: International Conf. on Computer Vision (ICCV), 2013, pp. 3192–3199.

[14] F. Caba Heilbron, V. Escorcia, B. Ghanem, J. Carlos Niebles, Activitynet: A large-scale video benchmark for human activity understanding, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[15] D. Shao, Y. Zhao, B. Dai, 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.

[16] L. Chunhui, H. Yueyu, L. Yanghao, S. Sijie, L. Jiaying, Pku-mmd: A large scale benchmark for continuous multi-modal human action understanding, arXiv preprint arXiv:1703.07475.

[17] C. Gu, C. Sun, D. A. Ross, C. Vondrick, C. Pantofaru, Y. Li, S. Vijayanarasimhan, G. Toderici, S. Ricco, R. Sukthankar, et al., Ava: A video dataset of spatio-temporally localized atomic visual actions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 6047–6056.

[18] L. Tao, T. Burghardt, S. Hannuna, M. Camplani, A. Paiement, D. Damen, M. Mirmehdi, I. Craddock, A comparative home activity monitoring study using visual and inertial sensors, in: 2015 17th International Conference on E-health Networking, Application & Services (HealthCom), IEEE, 2015, pp. 644–647.

[19] K. Soomro, A. R. Zamir, M. Shah, Ucf101: A dataset of 101 human actions classes from videos in the wild (2012). doi:10.48550/ARXIV.1212.0402.
URL `https://arxiv.org/abs/1212.0402`

[20] 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.

[21] Z. Tang, A. Hoover, A new video dataset for recognizing intake gestures in a cafeteria setting, in: 2022 26th International Conference on Pattern Recognition (ICPR), 2022, pp. 4399–4405. doi:10.1109/ICPR56361.2022.9956550.

[22] A. Hoover, Data description: Clemson cafeteria dataset, Online, URL: http://cecas. clemson. edu/ahoover/cafeteria.

[23] S. Bi, D. Kotz, Eating detection with a head-mounted video camera, in: 2022 IEEE 10th International Conference on Healthcare Informatics (ICHI), 2022, pp. 60–66. doi:10.1109/ICHI54592.2022.00021.

[24] P. A. Neves, J. Simões, R. Costa, L. Pimenta, N. J. Gonçalves, C. Albuquerque, C. Cunha, E. Zdravevski, P. Lameski, N. M. Garcia, et al., Thought on food: A systematic review of current approaches and challenges for food intake detection, Sensors 22 (17) (2022) 6443.

[25] P. V. Rouast, H. Heydarian, M. T. Adam, M. E. Rollo, Oreba: A dataset for objectively recognizing eating behavior and associated intake, IEEE Access 8 (2020) 181955–181963.

[26] C. A. Merck, C. Maher, M. Mirtchouk, M. Zheng, Y. Huang, S. Kleinberg, Multimodality sensing for eating recognition., in: PervasiveHealth, 2016, pp. 130–137.

[27] Y. Shen, J. Salley, E. Muth, A. Hoover, Assessing the accuracy of a wrist motion tracking method for counting bites across demographic and food variables, IEEE journal of biomedical and health informatics 21 (3) (2016) 599–606.

[28] K. Kyritsis, C. Diou, A. Delopoulos, Modeling wrist micromovements to measure in-meal eating behavior from inertial sensor data, IEEE journal of biomedical and health informatics 23 (6) (2019) 2325–2334.

[29] Q. Men, H. Leung, Y. Yang, Self-feeding frequency estimation and eating action recognition from skeletal representation using kinect, World Wide Web 22 (2019) 1343–1358.

[30] A. Iosifidis, E. Marami, A. Tefas, I. Pitas, K. Lyroudia, The mobiserv-aiia eating and drinking multi-view database for vision-based assisted living, Journal of Information Hiding and Multimedia Signal Processing 6 (2) (2015) 254–273.

[31] L. Onofri, P. Soda, M. Pechenizkiy, G. Iannello, A survey on using domain and contextual knowledge for human activity recognition in video streams, Expert Systems with Applications 63 (2016) 97–111.

[32] W. Li, Z. Zhang, Z. Liu, Action recognition based on a bag of 3d points, in: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops, 2010, pp. 9–14. doi:10.1109/CVPRW.2010.5543273.

[33] J. Wang, Z. Liu, Y. Wu, J. Yuan, Mining actionlet ensemble for action recognition with depth cameras, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 1290–1297. doi:10.1109/CVPR.2012.6247813.

[34] A. Paiement, L. Tao, S. Hannuna, M. Camplani, D. Damen, M. Mirmehdi, Online quality assessment of human movement from skeleton data, in: British Machine Vision Conference, BMVA press, 2014, pp. 153–166.

[35] J. Ortells, M. T. Herrero-Ezquerro, R. A. Mollineda, Vision-based gait impairment analysis for aided diagnosis, Medical & biological engineering & computing 56 (9) (2018) 1553–1564.

[36] D. Sethi, S. Bharti, C. Prakash, A comprehensive survey on gait analysis: History, parameters, approaches, pose estimation, and future work, Artificial Intelligence in Medicine (2022) 102314.

[37] M. Moro, G. Marchesi, F. Hesse, F. Odone, M. Casadio, Markerless vs. marker-based gait analysis: A proof of concept study, Sensors 22 (5). doi:10.3390/s22052011.
URL `https://www.mdpi.com/1424-8220/22/5/2011`

[38] Y. Makihara, M. S. Nixon, Y. Yagi, Gait recognition: Databases, representations, and applications, Computer Vision: A Reference Guide (2020) 1–13.

[39] Y. Sun, J. S. Hare, M. S. Nixon, Detecting heel strikes for gait analysis through acceleration flow, IET Computer Vision 12 (5) (2018) 686–692.

[40] L. Wang, G. Zhao, N. Rajpoot, M. S. Nixon, Special issue on new advances in video-based gait analysis and applications: challenges and solutions, IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 40 (4) (2010) 982–985.

[41] J. Liu, A. Shahroudy, M. Perez, G. Wang, L.-Y. Duan, A. C. Kot, Ntu rgb+d 120: A large-scale benchmark for 3d human activity understanding, IEEE Transactions on Pattern Analysis and Machine Intelligence 42 (10) (2020) 2684–2701. doi:10.1109/TPAMI.2019.2916873.

[42] J. Liu, A. Shahroudy, D. Xu, G. Wang, Spatio-temporal lstm with trust gates for 3d human action recognition, in: B. Leibe, J. Matas, N. Sebe, M. Welling (Eds.), Computer Vision – ECCV 2016, Springer International Publishing, Cham, 2016, pp. 816–833.

[43] S. Song, C. Lan, J. Xing, W. Zeng, J. Liu, An end-to-end spatio-temporal attention model for human action recognition from skeleton data, in: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, AAAI'17, AAAI Press, 2017, p. 4263–4270.

[44] H. Duan, Y. Zhao, K. Chen, D. Lin, B. Dai, Revisiting skeleton-based action recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 2969–2978.

[45] S. Yan, Y. Xiong, D. Lin, Spatial temporal graph convolutional networks for skeleton-based action recognition, in: Thirty-second AAAI conference on artificial intelligence, 2018.

[46] M. Li, S. Chen, X. Chen, Y. Zhang, Y. Wang, Q. Tian, Actional-structural graph convolutional networks for skeleton-based action recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019.

[47] L. Shi, Y. Zhang, J. Cheng, H. Lu, Skeleton-based action recognition with directed graph neural networks, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019.

[48] L. Shi, Y. Zhang, J. Cheng, H. Lu, Two-stream adaptive graph convolutional networks for skeleton-based action recognition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019.

[49] Y. Chen, Z. Zhang, C. Yuan, B. Li, Y. Deng, W. Hu, Channel-wise topology refinement graph convolution for skeleton-based action recognition, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2021, pp. 13359–13368.

[50] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, L. V. Gool, Temporal segment networks: Towards good practices for deep action recognition, in: European conference on computer vision, Springer, 2016, pp. 20–36.

[51] C. Yang, Y. Xu, J. Shi, B. Dai, B. Zhou, Temporal pyramid network for action recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[52] Z. Liu, L. Wang, W. Wu, C. Qian, T. Lu, Tam: Temporal adaptive module for video recognition, in: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2021, pp. 13708–13718.

[53] M. A. Gul, M. H. Yousaf, S. Nawaz, Z. Ur Rehman, H. Kim, Patient monitoring by abnormal human activity recognition based on cnn architecture, Electronics 9 (12) (2020) 1993.

[54] P. Woznowski., R. King., W. Harwin., I. Craddock., A human activity recognition framework for healthcare applications: Ontology, labelling strategies, and best practice, in: Proceedings of the International Conference on Internet of Things and Big Data - IoTBD,, INSTICC, SciTePress, 2016, pp. 369–377. doi:10.5220/0005932503690377.

[55] S. Sharma, A. Hoover, Top-down detection of eating episodes by analyzing large windows of wrist motion using a convolutional neural network, Bioengineering 9 (2) (2022) 70.

[56] K. Okamoto, K. Yanai, Grillcam: A real-time eating action recognition system, in: International Conference on Multimedia Modeling, Springer, 2016, pp. 331–335.

[57] T. Lin, X. Zhao, H. Su, C. Wang, M. Yang, Bsn: Boundary sensitive network for temporal action proposal generation, in: Proceedings of the European conference on computer vision (ECCV), 2018, pp. 3–19.

[58] T. Lin, X. Liu, X. Li, E. Ding, S. Wen, Bmn: Boundary-matching network for temporal action proposal generation, in: Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 3889–3898.

[59] C. Zhang, J. Wu, Y. Li, Actionformer: Localizing moments of actions with transformers, arXiv preprint arXiv:2202.07925.

[60] E. Curto, H. Araujo, An experimental assessment of depth estimation in transparent and translucent scenes for intel realsense d415, sr305 and l515, Sensors 22 (19) (2022) 7378.

[61] B. Cheng, B. Xiao, J. Wang, H. Shi, T. S. Huang, L. Zhang, Higherhrnet: Scale-aware representation learning for bottom-up human pose estimation, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[62] Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, Y. A. Sheikh, Openpose: Realtime multi-person 2d pose estimation using part affinity fields, IEEE Transactions on Pattern Analysis and Machine Intelligence.

[63] F. Zhang, X. Zhu, H. Dai, M. Ye, C. Zhu, Distribution-aware coordinate representation for human pose estimation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[64] A. Toshev, C. Szegedy, Deeppose: Human pose estimation via deep neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.

[65] L. Xu, Y. Guan, S. Jin, W. Liu, C. Qian, P. Luo, W. Ouyang, X. Wang, Vipnas: Efficient video pose estimation via neural architecture search, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 16072–16081.

[66] A. Dutta, A. Zisserman, The VIA annotation software for images, audio and video, in: Proceedings of the 27th ACM International Conference on Multimedia, MM '19, ACM, New York, NY, USA, 2019. doi:10.1145/3343031.3350535.
URL `https://doi.org/10.1145/3343031.3350535`

[67] Y.-G. Jiang, J. Liu, A. Roshan Zamir, G. Toderici, I. Laptev, M. Shah, R. Sukthankar, THUMOS challenge: Action recognition with a large number of classes, http://crcv.ucf.edu/THUMOS14/ (2014).

[68] H. Alwassel, S. Giancola, B. Ghanem, Tsp: Temporally-sensitive pretraining of video encoders for localization tasks, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 3173–3183.

[69] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, T.-Y. Liu, Lightgbm: A highly efficient gradient boosting decision tree, Advances in neural information processing systems 30.