COMPOSITIONAL MODELLING OF FIRST-PERSON ACTIONS AS VERB-NOUN STREAMS USING LSTM BASED LATE FUSION STRATEGIES

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL
OF
TED UNIVERSITY

BY

ZEYNEP GÖKCE

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
INTERACTIVE COMPUTING AND INFORMATION SYSTEMS

JULY 2019

Approval of the Graduate School of TEDU

_____

Prof. Dr. Kezban Çelik
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

_____

Assist. Prof. Dr. Tayfun Küçükyılmaz
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

_____

Assist. Prof. Dr. Selen Pehlivan Tort
Supervisor

Examining Committee Members

Prof. Dr. Pınar Duygulu Şahin (Hacettepe University, Computer Engineering) _____

Assist. Prof. Dr. Selen Pehlivan Tort (TED University, Computer Engineering) _____

Assist. Prof. Dr. Venera Adanova (TED University, Computer Engineering) _____

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: ZEYNEP GÖKCE

Signature :

ABSTRACT

COMPOSITIONAL MODELLING OF FIRST-PERSON ACTIONS AS VERB-NOUN STREAMS USING LSTM BASED LATE FUSION STRATEGIES

GÖKCE, ZEYNEP
M.S., Department of Interactive Computing and Information Systems
Supervisor: Assist. Prof. Dr. Selen Pehlivan Tort

JULY 2019, 61 pages

Analysis of first-person videos involving human actions could help in the solutions of many problems. These videos include a large number of fine-grained action categories with hand-object interactions. In this thesis, compositional modeling of verb and object streams with various fusion strategies is proposed to recognize human actions in first-person videos. We utilize 3D Convolutional Neural Network model, C3D, for verb stream to model video-based features in multiple scales, and we utilize object detection model, YOLO, for object stream to model objects interacting with hand. Two fusion strategies are proposed to combine these two streams. In the first one, human actions are obtained by simple multiplication without learning. In the second, LSTM based models are proposed. Experimental results on EGTEA Gaze+ dataset obtained from these two different fusion strategies show that our composite models present promising results compared to the baseline action models.

Keywords: human action recognition, first-person videos, deep learning, computer

vision

ÖZ

BİRİNCİ-ŞAHIS HAREKETLERİNİN LSTM TABANLI GEÇ FÜZYON STRATEJİLERİ KULLANARAK FİİL-NESNE AKIŞLARI OLARAK BİRLEŞİMSEL MODELLENMESİ

GÖKCE, ZEYNEP
Yüksek Lisans, İnteraktif Bilişim Sistemleri Bölümü
Tez Yöneticisi: Dr. Öğr. Üyesi. Selen Pehlivan Tort

Temmuz 2019 , 61 sayfa

Birinci-şahıs videolarının analizi, insan hareketlerini içeren birçok probleme çözüm sunmaktadır. Bu videolar, el-nesne etkileşimlerine sahip çok sayıda detaylı eylem kategorisi içerir. Bu tezde, birinci-şahıs videolarındaki insan hareketlerini tanımlamak amacıyla çeşitli füzyon stratejileriyle fiil ve nesne akışlarının birleşimsel modellenmesi önerilmiştir. Fiil akışında, video tabanlı özellikleri çoklu ölçeklerde modellemek için 3 Boyutlu Konvolüsyonlu Sinir Ağı modeli, C3D, kullandık. Nesne akışında ise el ile etkileşimde bulunan nesneleri modellemek için nesne algılama modeli, YOLO, kullandık. Bu iki akışı birleştirmek için iki farklı füzyon stratejisi önerilmiştir. İlkinde, insan hareketleri herhangi bir öğrenme gerçekleştirmeden basit bir çarpımla elde edilmektedir. İkincisinde ise LSTM tabanlı modeller kullanılmıştır. EGTEA Gaze+ veri seti üzerinde iki farklı füzyon metodolojilerinden elde ettiğimiz deneysel sonuçlar, birleşik modellerimizin taban modeli olan C3D hareket modelinden daha başarılı olduğunu göstermiştir.

Anahtar Kelimeler: insan hareketlerini tanıma, birinci-şahıs videolar, derin öğrenme, bilgisayarlı görü

To my family.

ACKNOWLEDGMENTS

I would first like to express my deepest gratitude to my supervisor Assist. Prof. Dr. Selen Pehlivan Tort for her guidance, encouragements, support, toleration and continuous supervision throughout my master studies. I feel so lucky to have studied with her throughout my thesis.

I am grateful to have met all my colleagues from B343 room and instructors in TEDU. Especially, I would like to mention Prof. Dr. Tolga K. Çapın, Assist. Prof. Dr. Bilgin Avenoğlu, Assist. Prof. Dr. Gizem Kayar and Assist. Prof. Dr. Tayfun Küçükyılmaz for supporting me throughout my research. It was great to work with them.

I offer my regards and blessings to my dear friends for their encouragements and supports. Especially, I would like to special thank to Cemal for his invaluable support.

Last but not least, I would like to thank my family for always being supportive with deep love. None of this would have been possible without their love and support.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

2D	2 Dimensional
3D	3 Dimensional
CNN	Convolutional Neural Network
3D-CNN	3 Dimensional Convolutional Network
RNN	Recurrent Neural Network
LSTM	Long Short Term Memory
BiLSTM	Bidirectional Long Short Term Memory
YOLO	You Only Look Once
YOLOv2	You Only Look Once Version 2
YOLOv3	You Only Look Once Version 3
SIFT	Scale Invariant Feature Transform
SURF	Speeded-up Robust Features
HOG	Histogram of Oriented Gradients
HOF	History of Optical Flow
BPTT	Back Propagation Through Time
IOU	Intersersection Over Union
ROI	Region of Interest
FPV	First Person Videos
MHI	Motion History Image
MEI	Motion Energy Image
HAR	Human Action Recognition
STIP	Space-Time Interest Points
MIL	Multiple Instance Learning
SVM	Support Vector Machine

KNN	K-Nearest Neighbor
convLSTM	Convolutional Long-Short Therm Memory
R-CNN	Regions with CNN

CHAPTER 1

INTRODUCTION

With the increasing availability and popularity of the wearable cameras such as GoPro [1], Google Glass [2], SenseCam [3] and Tobii Eye-Tracking Glasses [4], recordings with these cameras have become a part of daily life. A camera wearer can record thousands of hours of personalized videos with these cameras and the recorded data is known as the first-person video. First-person videos capture what the camera wearers see and consist of temporal and visual information in the wearer's point of view. Since these videos are in the first-person perspective, it enables to monitor the day-long activity of the camera wearer, and enables to understand the interactions with the surrounding objects or other individuals.

Due to the recent growth in the number of first-person videos, evaluation of daily human activities gain popularity in first-person vision. This has led to new applications for health monitoring, robotics, autonomous driving, and entertainment. Daily recording in first-person view can be used to monitor patient activities. For instance, first-person acts are analyzed to detect early signs of dementia [5]. In robotic, first-person videos are useful to make the robot learn the structure of human motion from the first person view [6]. For driver-assistance systems, monitoring the driver's behavioral status is studied to provide necessary assistance for safe and comfortable driving [7]. Besides, tracking and understanding human actions in first-person view is important for feasible applications in virtual reality [8].

Unlike third-person videos with fixed camera view, the first-person setting has dif-

[1] https://gopro.com/en/tr/
[2] https://www.google.com/glass/start/
[3] https://www.microsoft.com/en-us/research/project/sensecam/
[4] https://www.tobiipro.com/product-listing/tobii-pro-glasses-2/

ferent characteristics with new challenges such as camera view transition, frequent illumination changes. Since the videos are recorded from the perspective of the camera wearer, the wearer's attention in first-person videos is on the hands and the target objects in these videos.

As one fundamental problem of computer vision, action recognition task is also an important topic with wide range of applications in first-person vision. Day-long videos in first-person view include fine-grained action categories performed by hands with object interactions (e.g., pour seasoning from seasoning container to salad). Intra-class similarity makes action categories hard to distinguish from each other and the recognition task requires detailed spatial-temporal analysis to reveal details. During analysis, hands and objects are the most significant clues to determine the performed action, and the action is recognized using both hand motion and object appearances. Moreover, it is evident that models based on both appearance and motion achieve state of the art results on first-person videos [9].

In this thesis, our aim is to construct a composite model which is made of verb and object (noun) streams to perform action recognition in first-person videos. There is a large number of distinct action categories with verb-object labels (i.e. *take tomato* action is decomposed into *tomato* as an object and *take* as a verb). Splitting action recognition problem into verb and object recognition problems helps to cover a large number of categories as it decomposes the action space into verb and object spaces. In our model, verb recognition is modelled used a verb stream that encodes spatial-temporal behavior of hands performing action, while the object recognition is based on an object stream modelling object occurrences in the video. In order to recognize the action, various fusion methodologies are proposed to combine these distinct verb and object streams.

In this chapter, we present our research problem with our contributions. First, we explain some of the challenges appearing in recognition from the first-person view in Section 1.1. Then, we summarize the contributions of this study in Section 1.2. Finally, we present the content of this thesis in Section 1.3.

1.1 Challenges

Action recognition problem in first-person view is one of the challenging problems of computer vision. Major issues of action recognition explained in [10, 11] are expanded with our observations. The challenges of our problem in first-person videos are listed as follows

- **Intra-class variations.** A large number of action categories appear in first-person videos and the same action can be performed differently by different persons. For instance, a *mix salad* action sample can be performed by various kinds of hand held instruments, can be performed in different speed and direction.
- **Inter-class similarities.** Since the range of hand motion is limited in the first-person view, patterns of hand movements can be so similar to each other. For instance, the hand movements for *open* and *take* actions are performed similarly in two steps that are (1) the hand reaches out to an object in the first step and (2) the hand is pulled up towards the wearer's body after hand-object interaction in the second step.
- **Head movements.** Due to the wearable camera, the instant head movements cause camera uneven transitions making the visual content hard to understand.
- **Temporal variations.** Some action categories last longer than some other action categories. For instance, *mix* action is a periodic action and its samples take a long time compared to *put* action samples.
- **Limited vision.** Since the videos are recorded by the first-person point-of-view, only the camera wearer's hands and objects in close range are visible. First-person view may not provide enough information about the background. In fact, the hand can be partially visible and its pose can get lost.

1.2 Contributions

The contributions of our study in first-person action recognition are listed as follows

- **Action recognition model based on verb and object.** In this thesis, we propose a novel compositional action recognition model with two complementary streams corresponding to verb and object models. We show that our proposed compositional model outperforms the conventional action recognition model trained using action categories without any decomposition.

- **Hand model as motion information.** To the best of our knowledge, the hand information is generally proposed to reach object information in literature the [12]. It is not directly used to represent the action. Since hand knowledge is crucial in action recognition in first-person videos, hand based verb model is proposed to present the motion as verb classification. It has been shown that combination of detectors in multiple scales increases the performance, although hand based verb model is not adequate on its own to represent motion knowledge.

- **Full supervision with hand and object annotations.** Since the action space is huge and contains extreme number of action samples, we populate the recognition model with full supervision in this thesis. The background models of our system including action, verb and object detectors are trained using provided annotations in multiple levels. Even the EGTEA Gaze+ dataset provides ground truth video labels, it does not have the annotations for frame, hand and object locations. We annotate frames with hand and object bounding boxes.

1.3 Thesis Organization

The rest of this thesis is composed of four chapters as summarized below.

In Chapter 2, the related literature is reviewed by focusing on general purpose methods for action recognition and focusing specially on first-person action recognition. In addition, the popular first-person video datasets are provided in detail. Technical review of the background architectures YOLO, 3D-CNN and LSTM used as part of our action recognition model is explained as well.

In Chapter 3, our proposed architecture is described in detail. Background detection models and proposed late fusion strategies for compositional model are explained

with training details.

In Chapter 4, the experimental evaluations are provided with our observations. Evaluations are presented for two main fusion strategies and supported through ablation studies. In addition, comparison with recent studies is given.

In Chapter 5, Our proposed model are summarized with its limitations. The future work, that can be applied to improve our proposed model, is explained through a detailed discussion.

CHAPTER 2

RELATED WORKS

Research on egocentric video analysis emerged with the development of wearable cameras and was applied to a wide range of problems. The focus of this thesis is on recognizing actions from egocentric videos. This chapter first gives a brief review of action recognition problem with recent background models (see surveys [13, 11, 14] for more details), and then explains the recent studies used to recognize actions from the first-person view. Background models used in our study; convolutional neural networks, recurrent neural network with its variations and YOLO object detection network, are explained as well in this chapter.

2.1 Third-Person Action Recognition Models

In computer vision, various models are proposed to analyze human actions in videos. For human action recognition, classical methods are based on widely used hand-crafted features [15, 16, 17, 18]. With the advent of deep learning, recent studies focus on deep learning based approaches. In this section, we summarize the models using hand-crafted features, and then deep neural models used for action recognition.

2.1.1 Traditional Model Based on Hand-Crafted Features

Human action recognition studies in the literature consist of different kinds of approaches from global representations to local representations using hand crafted features, and classical machine learning methods. While global approaches are based on human silhouettes in image and optical flows, the local approaches are based on local

descriptors.

In some earlier studies, human action can be represented using global descriptors extracted from the whole images or silhouettes of images. In these studies, silhouettes and optical flow based models were proposed as global approaches [19]. In silhouettes based approach [20, 21, 22], global hand-crafted descriptors are defined over silhouettes. One of these studies was proposed by Bobick et al. [23] as an action recognition model with two components including motion-history image (MHI) and motion-energy image (MEI). Instead of using silhouettes, Lu et al. [24] proposed optical flow based model using Lucas-Kanade-Tomasi Tracker [25] for body joints tracing in time-step. Action recognition approach is based on correspondences between the human body postures in video frames including body joints. Instead of application of silhouettes and optical flow based models separately, Tran et al. [26] address the problem using both silhouette and optical flow approaches for action recognition.

Although the global approach is proposed widely in the literature, local approaches became popular with the emergence of space-time interest points (STIPs) proposed in [27]. STIP descriptors adapted from Harris Corner detector [28] represents human actions by modelling spatial and temporal information of interest points. Later, human motion in videos was represented using STIP descriptors in many studies [29, 30]. Besides STIP, Wang et al. [31] proposed trajectory based approach using dense-trajectory descriptor including HOG [32], HOF [33] and MBH [34] features extracted along trajectories.

2.1.2 Deep Neural Network Models

After deep architectures were applied for image classification [35, 36], they are also tested for video classification including action recognition problems. Popular deep learning models for action recognition include Convolutional Neural Network (CNN), Recurrent Neural Network (RNN) and Long Short Term Memory (LSTM) which is one particular variant of RNN. These models are used in action recognition not only for classification but also for feature extraction.

Action model proposed in [37] extracts hand-crafted features over spatial-temporal

volumes by computing Harris Corner points and their histograms. Then, these features are feed into the deep learning model to train for action classes. In another study, Mo et al. [38] use extracted CNN features for action classifications.

Due to spatial-temporal nature of videos, new deep learning models were designed for video understanding. 3D-CNN based models [39, 40, 4] achieve good results since they learn both spatial and temporal information using 3-dimensional convolution filters. Recently, LSTM models are able to learn over long sequences of data as they model the temporal dynamics from the video. Since LSTM is a good alternative for sequence classification, it is also proposed for action recognition using a sequence of images in videos [41]. In [42], LSTM is applied over sequence of video clips each represented by 3D convolutional neural network based feature, named C3D [4].

Another approach is stream-based action recognition models consisting of multiple deep neural network models. One such model is [43] with two streams that include one spatial stream using a single frame as input and another temporal stream using multi-frame optical flow representations as input. It has been shown that performance is increased using two-stream convolution neural network models.

2.2 First-Person Action Recognition Models

Within the development of wearable cameras, action recognition problems in first-person videos have become popular in computer vision. In literature, a wide range of studies are proposed. Traditional methods are developed using hand-crafted features such as GIST [44], HOG [32], HOF [33], STIP [27], SIFT [45], and trajectory-based MBH [31], and traditional classification models such as K-NN and SVM. With the advent of deep learning models, recent egocentric models for action recognition are developed using models such as CNN, LSTM, C3D and stream networks. In this section, we explore traditional and deep learning based egocentric action models from the literature in three main parts which are appearance-based, motion-based and hybrid models.

2.2.1 Appearance Based Models

Appearance cues such as occurrences of objects and their locations, gaze information, and hands with locations and sizes are informative for egocentric video understanding and they are used in many studies.

Objects have crucial knowledge to describe first-person actions by revealing human-object interactions. According to Fathi et al.[46], the target object is generally visible in the center of video frames. But, the target object can also be obtained using gaze or hand location. The first step in first-person action recognition is localizing hands and objects in video frames to define the region of interests (ROIs) where the action occurs. Later, appearance-based features can be extracted over ROIs.

In many studies [47, 46], hand-object interactions are modelled over extracted ROIs to understand egocentric activities.

Spriggs et al. [48] propose a K-NN based technique for activity recognition using GIST [44] feature and Inertial Measurement Units (IMU) data. 57.8% performance is achieved over CMU-MMAC [49] dataset when IMU and video data are combined. Another object-based model by Pirsiavash and Ramanan [50] introduces temporal pyramid based model provided to define the usage of the objects in video for action recognition. HOG [32] features are extracted and used for object modeling. With linear SVM classifier, performance is achieved up to 77% using object information over ADL dataset. Fathi et al. [46] address the importance of objects in first-person videos and propose a method with two steps over GTEA dataset. The first step is segmenting videos into foreground and background regions using optical flow, SIFT features, color histograms. Foreground segments are further decomposed into hands and active objects. The second step performs recognition using Multiple Instance Learning (MIL) over object segments. According to Fathi et al. [47], fine-grained actions are categorized using hand interaction features (such as optical flow of both hand and object, hand pose, hand location, hand size, and left/right-hand relative location). As their previous work [46], hands, foreground objects, and background are segmented and Adaboost [51] classifier is used for recognition. The proposed model is tested on GTEA dataset with an accuracy of 45%.

Recently, Cartas et al. [52] present another object-based model using human-object interaction with two steps over GTEA dataset. First, the hand region is segmented to get object region in video frames. The object regions surrounding hands are detected using Multiscale Combinational Grouping method [53]. Then, a star-structured region model, R*CNN [54], is used for more than one region classification. Last, the output of R*CNN as contextual cue is given to LSTM to predict action category.

Gaze information is another important cue for action recognition in first-person videos. The gaze of camera wearer generally focuses on the point where the action is performed. Therefore, visual features extracted around the gaze regions are more informative compared to features extracted from other regions. Fathi et al. [2] use gaze information for action recognition in first-person videos. This is extended object-based model of their previous work [47] with addition of gaze appearance. The SVM classifier is used for action categorization using object-based, gaze-based appearance features and future manipulation features. Object-based features are extracted from object classifiers including object context around the gaze point. Gaze-based appearance features are computed using histogram of color and texture area around gaze points. Future manipulation features consist of the information of whether the object is manipulated by hand in a few frames ahead. Using gaze information that is given with GTEA Gaze+ dataset, the unrelated background objects are eliminated and 47% performance is achieved compared to 27% of [47] on the same dataset without using gaze information. Similarly, Li et al. [1] develop a model for gaze prediction in first-person videos using hand/head movement, hand location and pose.

2.2.2 Motion Based Models

First-person videos capture motion information from camera wearer, head and hand movements and eye movements. Besides object-based models, which is known as appearance-based models, motion-based models are also proposed to recognize the first-person actions in the literature.

Kitani et al. [55] model motion in first-person sports activities using motion histograms. The motion histograms are based on optical flow of the scene. Due to the unsupervised scenario, Drichlet process mixture models are proposed to get ac-

tion categories using the motion histograms. Li et al. [56] model motion information using Dense Trajectories [31] as a baseline descriptor.

2.2.3 Hybrid Models

Appearance and motion domains are composed, i.e. using stream-based models which are introduced by [57]. These kinds of studies are based on not only appearance information but also motion information, since fusing them is also more informative for first-person action recognition as it can be seen in the following studies [9, 58, 59, 60].

Ma et al. [9] model object appearance and motion information as a two-stream network. The first stream analyzes appearance in three steps; segmentation, localization and object recognition successively. FCN32 network architecture is designed for segmenting hand regions. Later, FCN32 network architecture trained for hand segmentation is fine-tuned for object localization. This step gives the pixel-based occurrence probability of object of interest as output. Pre-processing each probability map, it uses the centroid of the largest blob as the predicted object center and the object is cropped with fixed-size bounding box. Finally, the cropped object regions with labels are given to the Object Recognition CNN, CNN-M-2048 model [61]. The second stream analyzes motion information similarly but using optical flow features. Finally, fully connected layers of these two streams are concatenated and a new fully connected layer is added on top of the network to recognize activities.

Yansong Tang et al. [58] integrate depth knowledge besides appearance and motion information and test over RGB-D egocentric dataset (THU-READ). Tri-stream network is proposed to incorporate appearance, motion and depth knowledge, and to encode RGB images, optical flow, and depth images respectively. Action prediction is calculated by taking the average score of three streams.

Hahn et al. [59] propose a model using visual information from videos and textual information from the recipe of these videos as well. The proposed model has three steps which are action proposal, object recognition, and recipe alignment steps. In action proposal step, video frames are localized in terms of having action or not.

Bidirectional LSTM is used for frame classification with two classes, action or not-action. In the object recognition step, ResNet101 network [62] is trained for object classification along with frames having actions according to the action proposal step. Finally, in the recipe alignment step, the action category of the video is predicted using a natural language processing model (Stanford Dependency Parser [63]).

Recently, G. Kapidis et al. [60] introduce a multi-modal approach based on sequential learning using LSTM architecture to recognize egocentric actions on EPIC-Kitchens dataset [64]. Due to unavailability of hand annotations in EPIC-Kitchens dataset, a hand detector is trained using Yolov3 [65] object detection model on hand samples gathered from various datasets. Then, objects in video frames of EPIC-Kitchens dataset are detected using object detection model. The object is also interpreted as a binary object presence vector using YOLOv3. Finally, LSTM is trained for action recognition using both information of hands as motion knowledge and object presence vector as object knowledge.

2.3 Brief Overview of Egocentric Datasets

The availability of first-person videos has increased along with a wearable camera and other devices in recent years. This section includes an overview of popular and the most relevant egocentric datasets (see Table 2.1).

CMU-MMAC. The Carnegie Mellon University Multi-modal Activity database (CMU-MMAC) [49] consists of cooking and meal preparation activities which are making brownies, pizza, sandwich, salad, and scrambled eggs. These 5 cooking activities are performed by 43 subjects. Different modalities are recorded which are video, audio, IMU signals, motion capture, wearable sensors.

ADL. This dataset was introduced for detecting activities of daily living in first-person videos. It consists of 1 million frames of 10 hours of video. It contains 32 actions performed by 20 people in 20 different homes such as combing hair, makeup, brushing teeth, etc. The dataset was annotated with activities, object tracks, hand positions, and interaction events [50].

GTEA. Georgia Tech Egocentric Activities (GTEA) dataset contains 7 meal preparation activities which are Hotdog, Sandwich, Instant Coffee, Peanut Butter Sandwich, Jamand Peanut Butter Sandwich, Sweet Tea, Coffee and Honey, Cheese Sandwich recorded by a head-mounted camera and each performed by 4 subjects. There are 28 activity videos with 71 action categories such as *put water cup*, *take coffee* and etc. Each action category was also labelled using verb and noun categories [47, 46].

GTEA Gaze. This dataset was collected using Tobii eye-tracking glasses. In total, there are 17 video sequences with gaze information performed by 14 different subjects. Each action in these videos is represented with a verb and set of nouns such as *pouring milk into cup*. Pixel-level hand annotations are provided as well [2].

GTEA Gaze+. This dataset is a variant of GTEA Gaze+ and it was collected using SMI eye-tracking glasses. It consists of 7 meal-preparation activities which are American Breakfast, Pizza, Snack, Greek Salad, Pasta Salad, Turkey Sandwich and Cheese Burger with recipes. 37 activity videos performed by 26 subjects are provided with gaze tracking and action annotation. Each action consists of verb and set of nouns such as *put milk container*. Similarly, this dataset also provides pixel-level hand annotations. 44 action categories are used in [56] and the action categories are listed in web pages of dataset [66] for a fair comparison.

EGTEA Gaze+. This is also available large egocentric dataset. There are 86 egocentric activities consist of cooking videos. It includes 106 action categories which consist of 19 verb categories and 53 object categories. [12].

THU-READ. Tsinghua University RGB-D Egocentric Action Dataset provides 2 modalities which are RGB and depth modalities. It contains 40 actions which are performed by 8 subjects (6 males and 2 females). Each action is repeated 3 times by each subject. In total, there are 1920 action clips. These action clips are divided into 4 splits in subject-based; 3 splits are used for training, the other is used for testing [58].

BEOID. Bristol Egocentric Object Interactions Dataset is created using a head-mounted camera and eye tracker. It consists of 58 different activity videos performed by 8 different subjects in 6 different places which are kitchen, workspace, laser printer, corridor with a locked door, cardiac gym, and weight-lifting machine. The action labels

with 3D object information are available [67].

EPIC-Kitchens. The first-person dataset is recorded using a head-mounted camera in 32 different kitchens which are the wearers' kitchens at their home. There are 55 hours of recording with 39,594 action segments which consist of 125 verb classes along with 331 noun classes. This dataset also provides the 454,255 object bounding boxes [68].

UT Egocentric. Various indoor and outdoor activities such as cooking, eating, driving, and shopping are recorded by 4 different subjects. There are 10 videos recorded but 4 of the 10 videos are available due to the privacy reasons. Annotated binary masks of images with positive and negative classes are given for region of interest. There is no available annotation except for binary mask of region of interest [69, 70].

Charades-Ego. It consists of 157 different types of daily activities with 68,536 annotated samples which are recorded both in first-person and third-person views. There are 33 verb and 38 noun classes; the place information where the videos are recorded in is available per activity sample [71].

Datasets	Task	#Subject	#Action Categories	#Video sequence	#Action Segments	#Verb/Noun Categories	Object Labels	Hand Labels	Other Sensors
GTEA	Action	4	71 (61*)	28	525 (456*)	8/16	N/A	Pixel-Level	N/A
GTEA Gaze	Action	14	40 (25*)	17	331 (270*)	N/A	N/A	N/A	Gaze
GTEA Gaze+	Action	6	44	37	1958	N/A	N/A	Pixel-Level	Gaze
EGTEA Gaze+	Activity	32	106	86	10325	19 /53	N/A	Pixel-Level	Gaze
ADL	Activity	20	18	N/A	364	N/A	Bounding Box	N/A	N/A
THU-READ	Action	8	40	N/A	N/A	N/A	N/A	Pixel-Level	Depth
CMU-MMAC	Action	43	29	N/A	N/A	N/A	N/A	N/A	Audio, IMU Signals
BEOID	Action	5	34	58	742	N/A	3D Object Labels	N/A	N/A
EPIC-Kitchens	Action	N/A	149	432	39.594	125/331	Bounding Box	N/A	Audio
UT Egocentric	Activity	4	N/A	4/10 is available	N/A	N/A	N/A	N/A	N/A
Charades-Ego	Activity	71	157	7860	68536	33 / 38	N/A	N/A	N/A

Table 2.1: Comparative overview of first-person action datasets in detail. *Only subset of the dataset benchmarked in some studies [1, 2, 3]

2.4 Convolutional Neural Network

Convolutional neural network (CNN) is a deep learning algorithm, developed inspiring from the human brain cortex. CNN is similar to regular neural networks and consists of a set of layers that are an input layer, followed by hidden layers (convolutional, pooling, and fully connected layers) and an output layer. There are various CNN architectures constructed for image data such as LeNet [72], AlexNet [35], VGGNet [73], GoogLeNet [74], ResNet [62]. Unlike classical computer vision approaches, these models take images as input and construct visual concepts without any pre-processing.

LeNet5 architecture by LeCun et al. [72] was introduced as a pioneer of convolutional neural network with 7 convolution layers and developed for handwritten digit/character recognition on bank checks. Deeper CNN architectures have become popular with AlexNet [35] that won ImageNet ILSVRC competition in 2012. Another deep CNN architecture with 16 convolution layer, VGGNet [73], was constructed using 2 or 3 consecutive convolutional layers. Unlike previous neural network models, ResNet [62] was developed with residual blocks. Within these residual blocks, the output of a layer not only feeds into the next layer but also feeds into the 2-3 next layers. GoogLeNet [74] proposed inception modules as a wider and deeper model than others for image classification. Each inception module consists of multiple convolutional layers with various spatial sizes.

There are two types of convolution which are 2D convolution, and 3D convolution. While the input is convolved with the kernel only spatially in 2D convolution, the input is convolved with the kernel both spatially and temporally in 3D convolution. 3D convolution is generally preferred for video data processing since it models spatial-temporal information. 3D Convolutional Networks [75, 4] are CNN models with good performances in action recognition over videos.

Figure 2.1: 3D Convolutional Neural Network (C3D) [4] architecture with 8 convolutional, 5 pooling and 2 fully connected layers.

2.5 3D Convolutional Neural Network

For the reasons that appearance and motion information is important to describe videos and 3D-CNN learns spatio-temporal information of videos well [4, 75]), 3D-CNN is proposed for action recognition in videos.

In this part, 3D-CNN architecture, which is named C3D network in this paper [4], proposed by Tran et al. [4] is described in detail. There are 8 convolution layers, 5 pooling layers, 2 fully connected layers and a softmax layer in the proposed architecture (see Figure 2.1). All 3D convolution filters are 3x3x3 with stride 1x1x1. During training, five clips with 16x112x112 size are cropped randomly from each training video as spatially and temporally. Also, horizontal flipping is applied to each clip with 50% probability. Trained C3D model is used for feature extractor. For each clip with 8 frames overlapping temporally, C3D features extracted from fc6 layer. The video is represented with the 4096-dim feature taking an average of the fc6 feature maps. After following L2-normalization, C3D video descriptors are given to multi-class linear SVM classifier to predict action category. It has been also shown that C3D features combined with hand-crafted features such as dense trajectory increase the accuracy.

2.6 Recurrent Neural Network

The feed-forward neural networks such as CNN could not deliver high performance for sequential data such as time series data, videos, audio, and text, since it is assumed that inputs and outputs in each layer of the network are independent. No correlation between each point of the sequential data can be established in these networks. On the contrary, Recurrent Neural Network (RNN) is named as recurrent since (1) the calculation is repeated for each item of the sequential data using weight sharing strategy and (2) outputs in each time step depends on the calculation from the previous time steps. As shown in Figure 2.2, besides the input data x_t, the content units h_{t-1} showing the previous output also affect the network at time *t* in RNN. The decision for the input at the moment *t-1* also affects the decision to be made at time *t*. So, in these networks, inputs produce output by combining current and previous information.

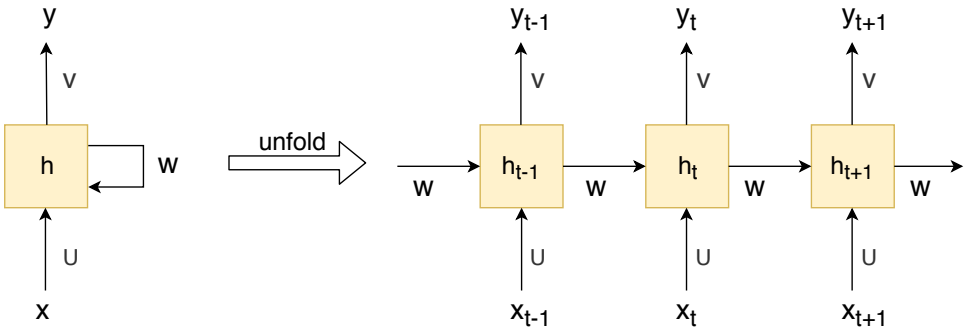


Figure 2.2: Recurrent neural network and ulfolded structure

The mathematical formula used in this model is given in Equation 2.1 and 2.2:

$$h_t = \sigma(U * x_t + W * h_{t-1}) \tag{2.1}$$

$$y_t = \phi(V * h_t) \tag{2.2}$$

where h_t is the hidden state, x_t is the input and y_t is the output at time *t* step. *U, W* and *V* are shared weights in network. The σ and ϕ is the activation function such as sigmoid or tanh. While training step, the weights are updated according to the previous and current information using Back Propagation Through Time (BPTT) algorithms [76, 77]. Backpropagation process is applied until the error between the ground truth and prediction outputs is minimized.

As RNN becomes more complex (deep in time), vanishing gradient problem has occurred since RNN is not able to handle long-term dependencies in practice. This problem has been solved using Long-Short Term Memory (LSTM), a special kind of RNN, which was first designed in 1997 [78].

Long Short Term Memory. Long Short Term Memory (LSTM) is a special type of recurrent neural network which is widely used. However, the difference from the recurrent neural network is that the memory cell in recurrent layers has a simpler structure. In each memory cell of LSTM, there are 3 multiplicative units, also known as gates, which consist of input gate, output gate and forget gate with a special task. LSTM memory cells can write, keep, erase and read information using mathematical operations using these gates. With the help of the gates in memory cell, it is decided

which parts of data are memorized, which part of the data is allowed to write on memory, when information is allowed to read from the memory and which parts of data is forgotten. The structure of LSTM memory cell is denoted in detail in Figure 2.3.

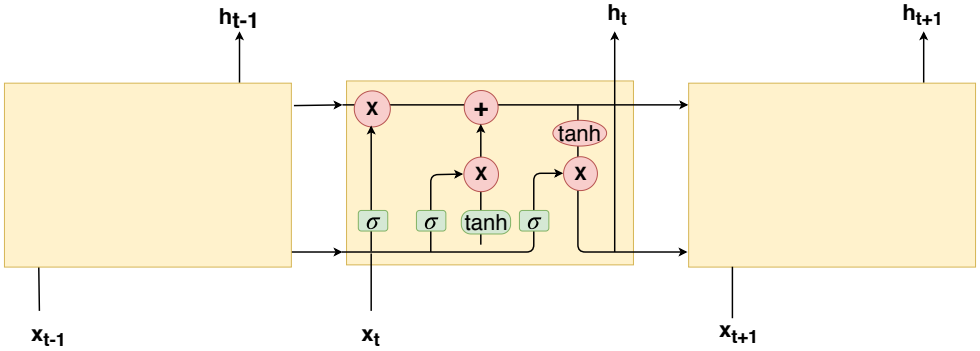


Figure 2.3: Long-Short Term Memory Network cell in detail

The mathematical operations in LSTM memory cell are given in the following Equations (2.3 - 2.7). In each time step of the cell, x_t, h_{t-1} and C_{t-1} are inputs and C_t and h_t are outputs of the network. The first stage in LSTM memory cell is the operation of forget gate. It is decided which part of input data is aroused from memory (see Equation (2.3)):

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f), \tag{2.3}$$

where σ is the sigmoid function, inputs are x_t at t time step and h_{t-1} from t-1 time step and the network parameters are W_f and b_f.

The second stage is the decision of which parts of the data is memorized (see Equation (2.4-2.5)):

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i). \tag{2.4}$$

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c). \tag{2.5}$$

The third stage is the update of current state of C_t using C_{t-1} and i_t (see Equation (2.6)):

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t, \tag{2.6}$$

The last stage of the mathematical operations in LSTM cell is the output gate. After input, forget gates are computed, h_t is decided as output of cell (see Equation (2.7)).

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$
$$h_t = o_t * \tanh(C_t). \tag{2.7}$$

So, LSTM remembers the things for a long time with this memory cell and learns whether it will receive or release the data. Long-term dependency problem is handled with this memory cell structure.

Bidirectional LSTM. Bidirectional LSTM (BiLSTM) [79] is the combination of LSTM structure[78] and bidirectionality from Bidirectional RNN [80]. BiLSTM consists of backward and forward networks and the outputs of these sub-networks in each time step are combined to the same output layer [79]. It is based on the idea that the outputs at any time step may depend on not only previous data but also the future data. In Figure 2.4, it can be seen that BiLSTM is constructed with stacked two LSTM networks, which are forward and backward. Each layer of the BiLSTM feeds backward and forward information of the point of sequence data into the output layer simultaneously at each time step.

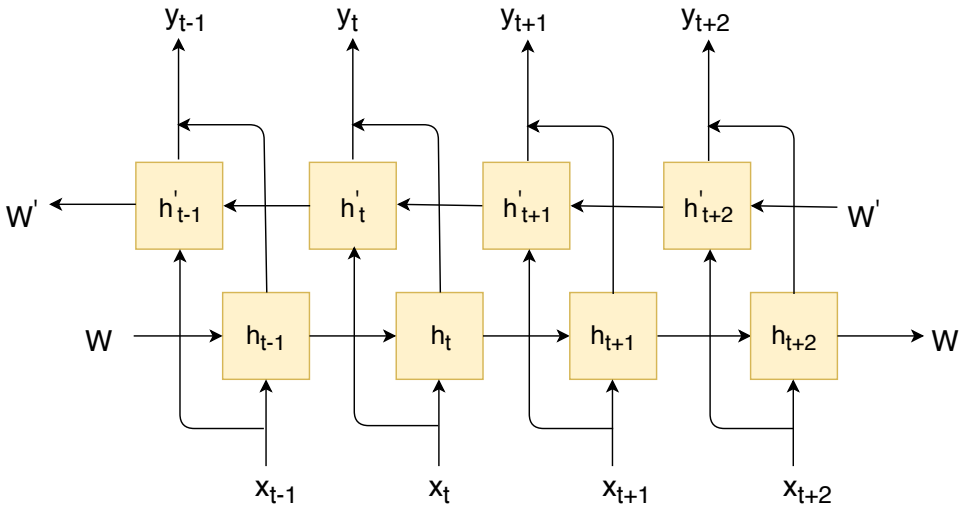


Figure 2.4: BiLSTM structure overview

2.7 YOLO Object Detection

Object detection is the identification of all relevant objects in an image with their bounding boxes and categories. YOLO [81] is a real-time deep learning model to detect objects in images using a regression approach and it is faster than other object detection models [82]. Unlike the prior deep learning models such as R-CNN and its variants [83, 84, 85], YOLO predicts classes and bounding boxes jointly and at real-time.

YOLOv2 architecture consists of just convolutional layers with max pooling. Darknet-19 architecture with 22 convolution layers is used which contains with 3x3 and 1x1 filters. Especially, 1x1 filters are applied on detection layers. Before the last detection layer, the shortcut connection is used in this architecture, as suggested in ResNet [62]. After low-level feature map from the previous layer is reshaped and concatenated with high-level layer feature map in the next layer, the concatenated feature is given to the next layer of the architecture.

Given a single RGB image as input, object detection is performed in the last layer known as a detection layer. Detection layer of the model creates SxSxN tensor as output where the input image is represented with SxS grid spatially. Each grid cell contains predictions of B bounding boxes for each class, each of which consists of 5 components (x, y, w, h, confidence score). x and y are the coordinates of the bounding boxes, w and h represent the width and height. N, depth of output tensor SxSxN, is defined in the following Equation 2.8.

$$N = B\text{x}(5 + C) \tag{2.8}$$

where $B = 5$ is default bounding box number with different scales, C is the class number, 5 is the components (x, y, w, h, confidence score) for each bounding box prediction.

In training of YOLOv2, it uses sum-squared error between the predictions and the ground truth to calculate the loss. Three terms which are confidence loss, classification and localization loss are used for total loss function. *Confidence loss* is the

objectness of the box. If there is no object in the bounding box, then the objectness loss should be reduced. *Localization* loss is an error between the predicted boundary box and the ground truth. The difference between predicted boundary box and ground truth should be minimum. It is also same for *classification* loss. In testing of YOLOv2, a threshold is applied to confidence score to eliminate the unrelated object bounding boxes from all predictions in the image.

The purpose of this YOLO object detection network in this thesis is to define object and hand detection in first-person video frames. It can be said that YOLO is the auxiliary model to find hand and object regions with categories for verb and object models in our proposed structure in Chapter 3.

CHAPTER 3

OUR APPROACH

First-person videos consist of fine-grained action categories with high similarity (e.g., pour seasoning from seasoning container to salad). Due to the fact that the first-person samples are difficult to distinguish from each other, one natural way for fine-grained recognition of these action categories is to decompose actions into verb and object categories and to investigate the co-occurrence of both categories on the same video sample. For instance, *take tomato* action can be identified by the recognition of verb category, *take*, and object category, *tomato*, simultaneously (see Figure 3.1).

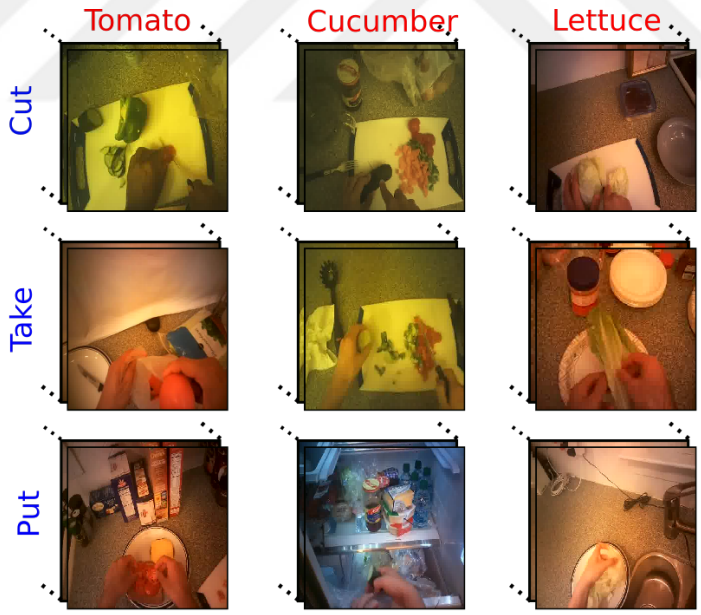


Figure 3.1: Verb-object pairs in first-person videos from EGTEA Gaze+ dataset.

Many studies emphasize the importance of both motion and appearance-based features for recognizing human actions with high success in first-person videos [9].

Both motion and appearance-based features can be modelled in different ways that are modeling with a simple concatenation of these features [59, 60] and modeling with stream-based structure [57, 9, 58].

With these motivations, in this thesis, two complementary steps are proposed to perform action recognition in first-person videos. The first step is determining the verb and object that construct the action. In this step, motion and appearance information is represented by verb and object models respectively and separately. The verb model is constructed by a spatial-temporal model C3D[4] which is a type of 3D convolutional neural network model, while object model is constructed using an object detector YOLOv2 [82], which is the second version of YOLO [81]. The second step is a fusion methodology which is based on the combination of these distinct verb and object models to perform action recognition. A simple count-based fusion methodology along with LSTM-based fusion methodologies are proposed in the second step of our approach.

The chapter continues with the overview of our approach in Section 3.1, verb model in Section 3.2, object model in Section 3.3, and fusion methodologies in Section 3.4 respectively.

3.1 Model Overview

Action recognition model is demonstrated in Figure 3.2. Using verb and object categories, the proposed architecture which has three main streams composed of Verb, Object, and Fusion; (1) Verb stream is a verb classifier that takes successive N clips and returns the verb scores per clip as an output. (2) Object stream is an object detection network taking video frames and returns object proposals with bounding boxes. Finally, (3) Fusion Stream is the action model employing various fusion strategies to recognize action categories taking the outputs of verb and object models.

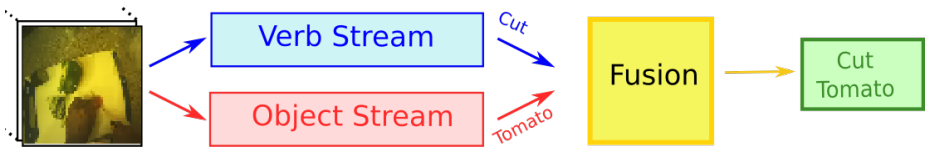


Figure 3.2: Model overview. Compositional modeling utilizes verb and object streams to recognize action in first-person videos.

3.2 Verb Stream as Video Verb Descriptor

We introduce the verb model to recognize the verb category of human action videos. Indeed, we present motion information of video using verb model as video verb descriptor. In this study, verb model is proposed as a composition of two sub-models which are full-scale verb model and hand-scale verb model. While full-scale verb model reveals coarse-grained description by utilizing full-scale frames with hands, objects and scene information, hand-scale model reveals fine-grained description by utilizing zoomed regions around hands.

C3D architecture [4] is employed in both sub-models for verb prediction, since it learns spatio-temporal information on videos during convolution (see Section 2.5). (Tensorflow implementation is used for C3D architecture utilizing the code in [86]) Each verb model, namely full-scale and hand-scale models, generates stream of prediction scores as softmax outputs per video. Particularly, these streams are VxC dimensional verb score matrices. These matrices are further combined using max-pooling over score vectors of clips with hand.

3.2.1 Full-Scale Verb Representation

The purpose of the full-scale model is to describe the verb category in coarser level. Model is based on C3D architecture which takes the video clips as input and outputs the category scores. Unlike the original C3D model [4] which utilizes randomly selected video clips over training videos, our model is trained on ground truth video clips in which the action is performed to increase the accuracy. Ground truth video clips are the clips extracted around the neighborhood of the ground truth frames (ex-

plained in Section 4.1.1). Following the original C3D setting [4], the video clips are resized to 112 x 112 x 3 x 16 before given to the C3D verb model.

Given a video, C successive clips, each of which includes 16 frames, are obtained by sliding 2 frames through the video. Each clip is given to full-scale C3D verb model, and prediction scores per category are taken as output. The model returns VxC dimensional verb score matrix as full-scale verb stream, where V is the number of verb categories and C is the number of clips.

3.2.2 Hand-Scale Verb Representation

Hand information is the most important clue to determine camera wearer's action in first-person videos. The purpose of hand-scale model is to describe verb category in finer level with a hand centric approach. This model utilizes hand regions eliminating background information from the video frames instead of looking at videos in full-scale as in the full-scale verb model. In order to classify verb of video using hand information, hand-scale verb model with two steps is proposed. First, hand detector is used to predict hand regions in frames of videos. Second, verb model takes the hand-volumes cropped around hand regions and predicts verb scores. The details of this model are explained as follows.

In the first step, we propose a hand-detection model to localize hands in video frames. We use the state-of-the art object detector YOLOv2 [87] to detect hands. The YOLO architecture is fine-tuned for one class (hand) using our hand dataset gathered from the EGTEA Gaze+. Our hand dataset and the annotation strategy are explained in Section 4.1.1. Trained hand detector takes video frames as input and returns detected regions as hand proposals with confidence scores.

Proposals having 50% overlap with ground truth hands labelled on ground truth frames are used to extract hand volumes. Particularly, a hand volume is a tube cropped around a hand-region, where the volume contains the hand-region in its mid-frame and lasts 16 frames keeping the spatial location of hand along with all frames. The cropped hand volume is spatially resized into 112 x 112 before it is fed into the hand-scale verb model as input. In order to be consistent with the full-scale verb models,

the hand-scale verb model is trained using C3D architecture [4] as well. While some verb actions such as $open$ and $take$ are performed by one hand, other verb actions such as cut and mix are performed by two hands. In the actions performed by both hands, one of these hands may not move with the same pattern of verb category. We can say that the one hand sometimes may act as an auxiliary hand to perform the main verb category in the video. For example, when we consider cut as a verb in the video, one hand cuts the object while the other hand holds the object. Following this, we extend verb categories with the additional verb category *hold* to learn hand motion more accurately. The hand-scale model is trained with verb categories of full-scale model as well as the verb category *hold*.

Given a video, hand-regions are extracted using the YOLO hand detector by sliding 2 frames through the video. Then, we obtain a hand-volume for each hand-region. Each hand volume is cropped around the hand bounding box. These volumes are fed into C3D model and prediction scores are obtained. It is possible that multiple hand volumes (e.g., cut action) may be detected on the same video frame (clip) since the action may be performed by both hands. In order to represent each video clip by one score vector, we apply max-pooling over prediction scores of multiple volumes. The model returns VxC dimensional verb score matrix as hand-scale verb stream, where V is the number of verb categories and C is the number of clips.

3.3 Object Stream as Video Object Descriptor

In first-person videos, objects manipulated by hand help to recognize fine-grained action categories. We aim to find which objects appeared in the video. YOLOv2 [87] is used to detect objects in video frames. Object detector is fine-tuned with object categories of the given dataset, and the object annotation strategy is explained in Section 4.1.2. Object model takes single frame at a time and proposes possible object bounding boxes with confidence scores. There may be so many objects predictions. In order to eliminate the invalid detections, a threshold (0,4) is applied to the confidence scores for objectness, and max-pooling is applied to pool object scores per frame.

The model returns an object stream as pooled and stacked confidence scores over

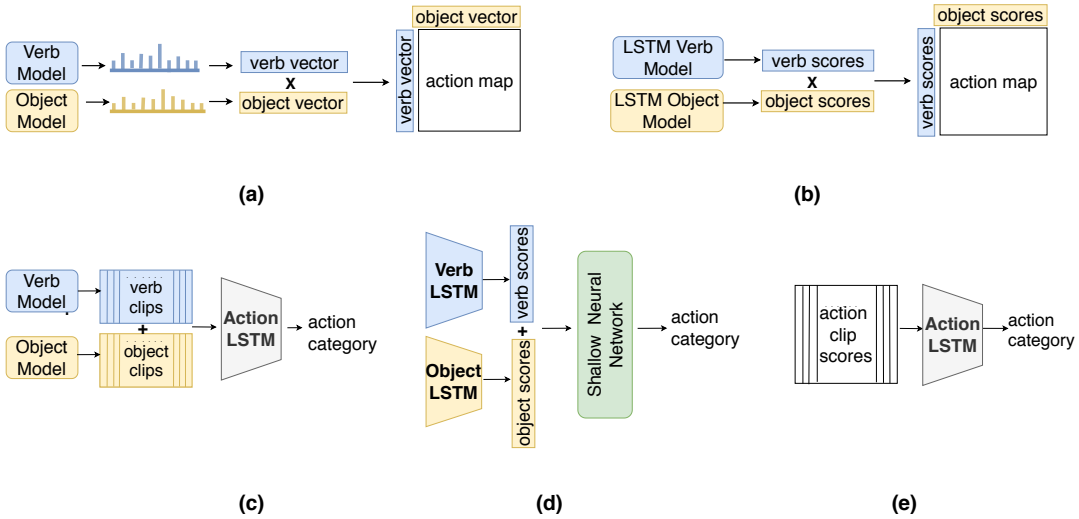


Figure 3.3: Fusion methodologies for action recognition. (a) Methodology 1: Simple Fusion. Verb and object vectors extracted from verb and object streams are multiplied. (b) Methodology 1. After verb and object scores are extracted from LSTM Verb and LSTM Object models, they are multiplied. (c) Methodology 2. Verb and object streams are concatenated using Action LSTM network. (d) Methodology 3. Verb and object streams are concatenated using shallow neural network (e) Methodology 4. Action LSTM utilizes the baseline model outputs for action recognition.

video frames and it corresponds to an OxC dimensional object score matrix, where O is the number of object categories and C is the number of frames.

3.4 Fusion Methodologies

Fusion stream is the last step of our proposed model and is referred to as the action model combining verb and object information extracted from verb and object streams. We employ multiple fusion methodologies with late fusion strategy inspiring from the study in [88]. We can categorize the proposed fusion methodologies in two ways. The first approach proposes a simple fusion strategy consisting of simple vector multiplication without any learning. The second fusion approach employs a set of late fusion strategies with various LSTM learning mechanisms. Figure 3.3 shows the fusion methodologies.

3.4.1 Simple Count Based Fusion Strategy

Using a simple fusion methodology, verb and object information extracted by verb and object models are combined to obtain the action label of the given video. It is simply based on multiplication operation without any learning. The structure of this fusion methodology can be seen in Figure 3.3 (a).

In the verb stream, successive clips for given test video, each of which includes fixed 16 frames, are obtained by sliding 2 frames through the video. First, two VxC verb score matrices where V is the number of verb categories and C is the number of clips is extracted using full-scale and hand-scale C3D verb models (see Section 3.2) and then these matrices are max-pooled. Later, each clip is assigned to a verb category by calculating the category with the maximum score among the prediction scores over CxV matrix. Assigning all clips to a verb category, a histogram showing the distribution of the verb categories over video clips is computed and L1 normalization is applied to eliminate the effect of video length. To sum up, the video is represented as a V dimensional verb-vector v.

In the object stream, in a similar way to verb stream, successive frames are obtained by sliding 2 frames through the video, and objects are detected using the object detection model (see Section 3.3). Each frame of the video is first assigned to an object category by calculating the category with the maximum score using OxC matrix computed by object model. Assigning all frames to an object category, a histogram showing the distribution of the object categories over video frames is extracted from all assigned object categories and L1 normalization is applied to eliminate the effect of video length. To sum up, the video is represented as an O dimensional object-vector o.

Inspired by a recent verb-object study for human-object interactions in still images[89], the VxO dimensional estimation map, E, is calculated to get action categories including all combinations of actions. E is extracted using a simple multiplication of v and o vectors from the verb and object models (see above). In order to evaluate the combinations of {*verb, object*} pairs existing in the dataset, the estimation map, E, is masked by a VxO dimensional binary mask, A, consisting of binary values (see

Equation 3.1). The binary activity mask size is constructed to preserve the verb-object space in our dataset. 1 shows the existence of a verb-object category pair, 0 shows the nonexistence of the verb-object category pair in our dataset (e.g., $cut\text{-}fridge$ pair is not consistent action category with our dataset).

$$\begin{aligned} E &= v \cdot o^T \\ E' &= E \odot A \end{aligned} \tag{3.1}$$

The {*verb, object*} pair with the maximum value over E matrix is assigned as the predicted category of the given video sample. As a result, the video is represented with appearance and motion feature vectors that are completely count-based (histogram based) and are extracted by utilizing verb and object models. This simple fusion methodology utilizes these features and provides the action category without training. The closest study to our simple count-based fusion methodology is the model proposed for human-object interaction in zero-shot setting [89]. However, the study is different from ours in terms of models and dataset that we used.

3.4.2 LSTM Based Fusion Strategies

Since action videos consist of sequential clips, recurrent models can be used to model the temporal relations between labels presented in consecutive clips. Given verb and object prediction vectors per clip, we use long short-term memory (LSTM) models to represent temporally the action of each clip. Such temporal information is complementary to video clips being a part of a longer video sequence, and is critical to smooth the labels of predicted clips. In this section, our aim is to propose fusion methodologies based on recurrent structures to provide not only the fusion of verb and object judgments for action classes but also improve accuracy. In contrast to the first fusion strategy, our proposed fusion strategy in this part is based on two variant model of RNN, namely LSTM and BiLSTM. Available verb and object detection scores are fused using four different ways:

Methodology 1. Individual LSTM networks is trained over verb and object streams of trained samples separately. Given a video, computed prediction scores for verb and

objects, verb-vector v and object-vector o, are combined using a simple multiplication as described in Eq. 3.1 to perform action recognition (see Figure 3.3 (b)).

Methodology 2. This fusion methodology utilizes the LSTM structure to recognize the action category of the video. We concatenate verb and objects prediction scores (streams) as representations of video clips. Later, single action based LSTM model is trained to predict the action category (see Figure 3.3 (c)).

Methodology 3. Similar to *methodology 1*, individual LSTM networks are trained over verb and object streams of train samples separately. Later, fully connected based shallow neural network model is trained to predict action categories over verb and object prediction scores of LSTM outputs. (see Figure 3.3 (d)).

Methodology 4. This fusion methodology utilizes the C3D model trained as an action detector, but not as a verb and object detector. First, a C3D model is trained on action categories on the whole dataset. Having action trained C3D model, the model is used to extract action prediction scores per clip. The prediction along clips are stacked into a single stream and fed into a single action based LSTM model (see Figure 3.3 (e)). Here, clips are extracted with a temporal stride of 2 frames over videos and each clip lasts 16 frames.

CHAPTER 4

EXPERIMENTS

In this chapter, we evaluate our models on action recognition and present the experimental results. We first introduce the dataset and the hand-object annotations we used and extracted in Section 4.1. Later, we present the evaluation criteria used to evaluate action recognition results 4.2. Then, we report the experimental results for (a) evaluations with ablation studies on simple fusion strategy 4.3, and (b) evaluations for LSTM based fusion strategies 4.4. We conclude and compare with recent studies on the same dataset in Section 4.5.

4.1 Dataset and Annotations

We perform our experiments on the EGTEA Gaze+ dataset [66] which includes first-person meal preparation activity videos. The EGTEA Gaze+ dataset is published in 2018 and extended from the GTEA Gaze+. It consists of cooking activities performed by 32 different subjects and includes 86 cooking videos providing 106 action categories in 3 train-test splits. Each train-test split has 8229 train and 2022 test video samples. In this dataset, action categories are made of verb and object pairs. 106 fine-grained action categories consist of 19 verb categories and 53 object categories. While some action categories such as *cut tomato* contain one object category, others such as *pour water-faucet-pot* contain more than one object category.

Since frame based action, hand and object annotations are missing, we annotate ground truth video frames and the hand and object bounding boxes occurring in these frames to use in our supervised model. The annotations of hand, object and action in video frames are demonstrated in Figure 4.1. We present hand, object and action

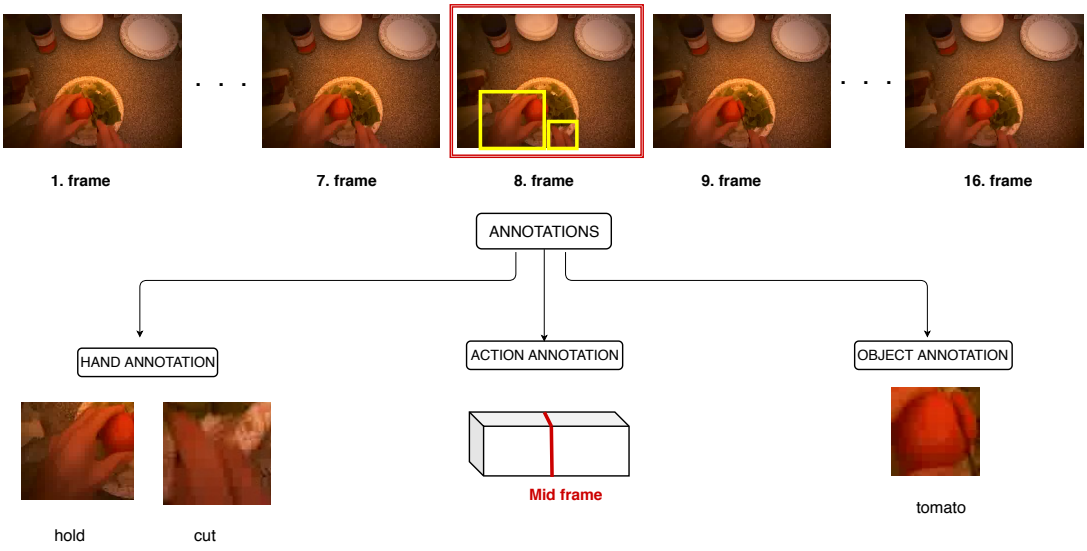


Figure 4.1: Hand, object and action annotations in video clip.

annotation details in Section 4.1.1, Section 4.1.2 and Section 4.1.3 respectively.

4.1.1 Hand Annotations

The EGTEA Gaze+ dataset has sparsely annotated pixel-level hand masks which are publicly available. Contrary to the available sparse annotations in the EGTEA Gaze+, our aim is to create our systematic hand annotations with bounding boxes. In our study, hands having consistent hand motion with the ground truth verb category are annotated in frames of the video samples. The annotated frames with hands correspond to the middle frames of action volumes used in the C3D verb model training (see Section 3.2).

We annotated each video sample that has visible and consistent hand motion over the 3 train-test splits of the dataset. Since *hand* is not included among object categories, *wash hand* action category in *wash* verb and *hand* object is just recognized using additional *wash-hand* verb category (video samples of *wash hand* action category are identified using only verb stream). Moreover, some video samples can not be annotated in a verb category due to the absence and invisibility of hand (video samples of *inspect-read recipe* action category are identified by *recipe* object using only object stream since hands are missing in these samples). As a result, annotations for 20 verb

categories are provided for training of hand models (including *hold* and *wash-hand*, and excluding *inspect-read*). The number of labelled hand samples with verb labels for train split 1, split 2 and split 3 are 8230, 8288 and 8075 respectively.

4.1.2 Object Annotations

The related objects in video clips are considered in our action scenario and the objects interacting with hand are used in our object model. Therefore, the frames with hand annotations are labelled with object locations as well. However, we do not achieve enough and balanced number of object samples, since some objects are not visible when they interact with hand. Therefore, the same object categories are populated with additional annotations from other frames of the videos.

Annotations of *water*, and *seasoning* are skipped since it is difficult to determine the properties (i.e. view, size, shape, etc.) of these object categories. Moreover, the *hand* is not included among object categories, since the hand annotations are used for hand detector which is an auxiliary model in our hand-scale verb model (see Section 3.2.2). As a result, annotations for 50 out of 53 object categories are provided for the object models. The number of labelled object samples for train split1, split2 and split3 are also 7251, 7295 and 7117 respectively.

4.1.3 Action Annotations

Current version of the EGTEA Gaze+ dataset does not include frame level annotations. However, the verb models (see Section 3.2) and baseline action models (see experiments in Section 4.3.3) are trained in a supervised setting and receive labelled video clips. Since annotated hands are on the middle frames of actions, evidently hand annotations are also utilized as action annotations for our proposed models.

4.2 Evaluation Metrics for Recognition

In order to evaluate first-person action recognition task, the classification accuracy metrics are defined as follows:

Overall Accuracy. The accuracy metric measures how accurately our model predicts and is the standard metric to evaluate the performance of our model.

$$\text{accuracy} = \frac{\sum_{i=1}^{C} \text{TP}_\text{i}}{\sum_{i=1}^{C} (\text{TP}_\text{i} + \text{FP}_\text{i})}$$

Mean Class Accuracy. The dataset has unbalanced data with various number of samples in each category. In this case, the category with more instance dominates the overall accuracy and makes it unreliable. In order to evaluate performance of our models, the accuracy is first calculated for each class, and then per-class accuracy numbers are averaged to get final accuracy. The mean accuracy formula is defined as

$$acc_i = \frac{TP_i}{TP_i + FP_i}$$

$$\text{mean accuracy} = \frac{1}{C} \sum_{i=1}^{C} (\text{acc}_\text{i})$$

where the C is the number of action class in dataset, TP is the number of true predicted instances, and FP is the number of false predicted instances. All experimental results are evaluated using mean class accuracy in the following.

4.3 Experiments on Simple Count Based Fusion Methodology

We evaluate the verb, object and action models using count based methodology and report accuracy of each model at video level. The baseline C3D verb model utilizes video clips consisting of 16 frames and YOLO object model utilizes video frames in each time step as input. In order to evaluate a test video, we sequentially test each clip and frame of video at a temporal stride of 2 frames. In the test stage, the set of clips in video at temporal stride 2 frames is called clip sequence. The set of frames in video at a temporal stride of 2 frames is called frame sequence. The final prediction

of each video is extracted using the histogram of categories of each clip and frame of the given video.

4.3.1 Ablation Studies on Verb Model

Verb Models with Scaled ROI. Region of interest (ROI) in different scales get different amount of information from the background. This situation may have crucial effects on accuracy. In order to examine these effects, we evaluate verb recognition performances in different scales for verb models. Individual performances of each scale are evaluated and the experimental results are shown in Table 4.1.

The first verb model is trained in full-scale mode where the whole frame of clips is used in training as explained in Section 3.2.1. For other scales, we crop regions and construct volumes around hand predictions. The region is determined using different enlargement scales. For a scale, horizontal and vertical enlargement is applied in all directions. Hand10 and and20 verb models mean that 10% and 20% of enlargement with respect to width and height of the detected hand regions are applied respectively.

Verb Models with Scaled ROI	Split1	Split2	Split3
C3D (Full Scale)	39.99	41.84	38.69
C3D (Hand20)	37.13	N/A	N/A
C3D (Hand10)	35.91	33.19	34.58

Table 4.1: Mean class accuracy results of verb models with different ROI scales in verb stream. Simple count based methodology is applied in these experiments and some experiments in some splits of dataset are not available.

We observed that full scale verb model outperforms the other hand based verb models since the full region contains the information of hand motion, object, and background. The context provided by each element enhances the ability to recognize action in videos [47]. When the individual verb categories are investigated, hand-scaled verb models outperform the full-scale one in some categories such as *open, put, crack* verb categories.

Combination of Verb Models. It is observed that although the full-scale verb model outperforms hand-scale verb models in terms of mean class accuracy, hand-scale verb models for some verb categories outperform the full-scale verb model. Therefore, we fuse the verb models using two fusion strategies, weighted average, and max-pooling. In the first strategy, softmax values of the clip sequence are averaged using weight parameter at clip level. The weight parameters (α_{full}, α_{hand10}, α_{hand20}) are empirically searched in the [0-1] range and selected by looking at the performances on the test split. Likewise, the max-pooling is applied to the softmax values of video clips. The results are reported in Table 4.2. It has been shown that the combination of softmax values extracted from verb models by weighted average and max-pooling enhances the verb recognition accuracy of the verb model. Due to the fact that full-scale verb model achieves the best accuracy, we keep full-scale verb model and combined with other two in this experiment.

Fusion Strategies	**Models**	**Split1**	**Split2**	**Split3**
Weighted Avg.	C3D (Full-Hand10)	46.99	43.99	43.16
	C3D (Full-Hand20)	45.63	N/A	N/A
	C3D (Full-Hand10-Hand20)	46.38	N/A	N/A
Max-Pooling	C3D (Full-Hand10)	N/A	43.57	42.08
	C3D (Full-Hand20)	44.91	N/A	N/A

Table 4.2: Experimental results of combined verb models based on simple count based methodology. The results are obtained using mean class accuracy metric. In weighted fusion strategy, the verb predictions from different verb models in different scales are averaged using weights, α_{full} for full-scale model, α_{hand10} for hand10 verb model, and α_{hand20} for hand20 verb model. C3D (Full-Hand20) model uses [$\alpha_{full} = 0.5$, $\alpha_{hand20} = 0.5$], C3D (Full-Hand10) model uses [$\alpha_{full} = 0.5$, $\alpha_{hand10} = 0.5$], and C3D (Full-Hand10-Hand20) uses [$\alpha_{full} = 0.4$, $\alpha_{hand10} = 0.3$, $\alpha_{hand20} = 0.3$]. Max-pooling strategy is just reported for Full-Scale and Hand20. The results are taken on split1 of the EGTEA Gaze+ dataset.

According to Table 4.2, the fusion strategy helps the verb model to increase mean class accuracy upto 46.99% when compared to the individual verb models reported in

Table 4.1. Although the accuracy of three scale combination (the combination of full-scale, hand10 and hand20 verb models) using weighted average strategy is higher than others, we continue with the full-scale and hand20 combination using max-pooling to be used in compositional action models in the action experiments. The reason is that max-pooling is a straightforward fusion strategy and easy to apply.

4.3.2 Ablation Studies on Object Model

We evaluate the accuracy of object model presented in Section 3.3. Two different approaches are used for evaluation. First, frame based object detection accuracy is calculated using mAP (Mean Average Precision). Second, the video based main object classification is calculated by using frequency of object appearances in the video frames. The results can be seen in Table 4.3.

In order to evaluate our object detector, YOLO, we provide detection results on split1 and we achieve 55.51% mAP. For *trash*, *mixture*, *condiment* object classes, low accuracies are achieved since these objects are hard to detect. Moreover, some objects are getting mixed up with each other. We observe that action labels of the EGTEA Gaze+ dataset video samples are often confused and some samples of the same class are labelled with different object categories. For instance, $tomato\ container$ object instances are visually similar to $grocery\ bag$ object instances. In another example, $fridge, fridge\ drawer$ and $drawer$ object instances are getting confused and used interchangeably in labelling. The low accuracy in object detection at the frame level seems to be originated from the ground truth labels of the original dataset.

We also evaluate the object model at video level. It is analyzed how often the main object of any action category appears on the video. Object model, YOLO, predicts objects in frames of the video sequence. The most frequently detected object category by YOLO in each video is assigned to that video as the main object category. 63.41%, 63.73%, and 62.24% accuracies are achieved for video object classification using mean class accuracy metric in split1, split2 and split3 respectively.

Object Models	Split1	Split2	Split3
YOLO object classification	63.41	63.73	62.24

Table 4.3: Object models are evaluated at video level with mean class accuracy (mACC) using simple count based methodology.

4.3.3 Experimental Results on Action Recognition

We provide two different experiments for action recognition. While the first experiment is the proposed action model based on simple multiplication, the second experiment is the baseline C3D action model for comparison.

Verb-Object Multiplication Model. The first action recognition model is based on a simple multiplication of verb and object vectors. The verb and object vectors are calculated for each test video using verb and object models. The video samples classified with a verb and an object pair simultaneously are assigned to corresponding action category (see Section 3.4.1 for details). The results are given in Table 4.4.

Baseline Model. The baseline action recognition model is constructed using C3D model having the same architecture with the verb model. The model is trained in supervised setting using annotated action frames (clips) where architecture learns actions rather than verb classes. For each clip of the test video, a softmax output over action labels is retrieved from the C3D model, then the frequently observed action label among clip sequence of the video is evaluated using simple count based histogram method proposed in Section 3.4.1 (instead of applying count based model on verb and object streams separately, we apply the same model on action scores).

Action Recognition Models	Split1	Split2	Split3
Verb-Object Multiplication	33.87	35.86	34.57
Baseline	21.85	N/A	N/A

Table 4.4: The action model based on verb and object multiplication along with the baseline C3D action model.

We observed that action model based on verb-object multiplication outperforms the baseline C3D action model with more than 10% accuracy. Although it is a simple fusion that consists of multiplication of verb and object vectors to recognize the action category without any learning, its performance is higher than baseline C3D action model, shown in Table 4.4.

4.4 Experiments on LSTM based Fusion Methodologies

We evaluate the verb, object and action models using LSTM and BiLSTM fusion methodologies. We make use of recurrent structures to construct verb, object and action models in this part.

4.4.1 Ablation Studies on LSTM Verb Model

Combination of features improves the verb model performance as seen from the verb experiments in Section 4.3. Therefore, we continue with the combination of different scales of verb models where we select the full-scale and a hand-scale verb model to fuse (Note that while hand20 verb model is used in action models of split1, we use hand10 model in action models for other splits since the hand10 and hand20 results in similar performances) and the LSTM structure is used for verb experiments. Combination types such as concatenation and max-pooling are applied on the softmax outputs of the verb models and then these combined features are deployed to BiLSTM and LSTM structures to analyze the performances.

BiLSTM and LSTM verb models are constructed using 2 layers with 50 cell size. We also use 10 as batch size and 0.001 as learning rate of Adam Optimizer. According to experimental results given in Table 4.5, the combination of full-scale and hand-scale verb models helps in verb recognition with 7.46 % improvement over split1 (For more detail, see confusion matrices of verb models in Figure 4.2). It is observed that the concatenation outperforms the max-pooling fusion method in verb models. It is clearly seen that BiLSTM verb models combining of softmax values outperform the simple count based fusion strategy given in Section 4.3. Since the verb model with BiLSTM structure (50.43%) gives higher accuracy than LSTM structure (48.13%) in

LSTM Verb Models	Split1	Split2	Split3
C3D + BiLSTM (full scale)	46.49	N/A	N/A
C3D + BiLSTM (full scale + hand, concatenation)	53.95	49.96	N/A
C3D + BiLSTM (full scale + hand, max-pooling)	**50.43**	**50.04**	**50.25**
C3D + LSTM (full scale + hand, max-pooling)	48.13	N/A	N/A

Table 4.5: Various combination of softmax values of verb models are employed by BiLSTM based and LSTM verb networks.

Table 4.5, we continue with the other verb models based on BiLSTM structure.

4.4.2 Ablation Studies on LSTM Object Model

In this part, we evaluate the object model presented in Section 3.3 using BiLSTM and LSTM. Object category probabilities of each frame in frame sequence, which are extracted by object model, are deployed to BiLSTM object model. Therefore, the video based main object classification is also performed using BiLSTM. The structure is constructed using 1 layer with 100 cell size. We use 30 as batch size and 0.001 as learning rate of Adam Optimizer. BiLSTM object model achieve 70.59% accuracy in split1 as can be seen in Table 4.6 (For more detail, see confusion matrix of in Figure 4.3). It is observed that the recurrent object model outperforms the object model which is in the simple count-based ablation study. It has been seen that BiLSTM object model also improves the accuracy compared to LSTM object model over split1.

LSTM Object Models	Split1	Split2	Split3
BiLSTM object model	70.59	71.34	67.73
LSTM object model	68.73	N/A	N/A

Table 4.6: Evaluation of LSTM based object models in the available splits of EGTEA Gaze+ dataset.

4.4.3 Experimental Results on Action Recognition

We make use of recurrent verb and object models and provide action models based on different fusion methodologies. The results of these experiments are presented in Table 4.7.

Action Recognition Models with LSTM based Methodology	Split1	Split2	Split3
Methodology 1 - BiLSTM Multiplication Model	45.29	N/A	N/A
Methodology 2 - BiLSTM Verb Object Combination Model, concatenation	45.18	N/A	N/A
Methodology 2 - LSTM Verb Object Combination Model, concatenation	44.36	N/A	N/A
Methodology 2 - BiLSTM Verb Object Combination Model, max-pooling	**46.49**	**41.20**	**40.82**
Methodology 2 - LSTM Verb Object Combination Model, max-pooling	44.10	N/A	N/A
Methodology 3 - BiLSTM Fusion using Simple CNN	45.45	N/A	N/A
Methodology 4 - BiLSTM Baseline Model	23.31	N/A	N/A

Table 4.7: LSTM based action models are provided with different approaches. Methodology 1 is the implementation of simple fusion based on verb and object vector multiplication, methodology 2 are LSTM action model which utilizes the different combination type of verb and object probabilistic values extracted from verb and object streams. Methodology 3 is the other experiment with simple CNN which consist of 2 fully connected layers. Finally, experiment in methodology 4 provides also LSTM based action model utilizes baseline C3D action model.

Methodology 1 - LSTM Multiplication Model. In order to identify action category of video, verb-object multiplication is simply applied. The verb and object vectors are extracted from LSTM verb model (C3D + BiLSTM (full scale + hand, max-pooling) given in Table 4.2) in the verb stream and LSTM object model (BiLSTM object model) in the object stream respectively. Given a test video, we simply multiply the LSTM verb vector and LSTM object vector. The verb-object pair, which has the maximum value on the matrix obtained by multiplication, is selected as the predicted action category of the video. This experiment is applied over split1, and 45.29% accuracy is achieved as can be seen in Table 4.7.

Methodology 2 - LSTM Verb Object Combination Model. Another experiment is performed utilizing BiLSTM and LSTM structures as an action recognition model. Unlike, methodology 1, the learning based model employs the combination of verb

and object streams. The verb probabilistic values which are concatenated with different scales and then are concatenated with object softmax values. To sum up, the model employs the verb and object knowledge for recognition. The networks are simply constructed using 1 layer, cell size of 100. It is trained and tested using parameters of 20 as batch size and 0.001 as learning rate. With this combined features, the action recognition accuracy is reached to 45.18% using BiLSTM structure over split1 in Table 4.7.

Similar to the combination with concatenation, we evaluate LSTM and BiLSTM structures by using a different combination of verb and object softmax values as input. In this experiment, max-pooling is applied to the verb softmax values from full-scale and hand-scale Verb Models. The max-pooled verb values are concatenated with object softmax values, and finally, combined values are employed by the model. The same architecture is also used for this experiment. The experimental results are given in Table 4.7. According to the table, 46.49%, 41.20% and 40.82% accuracies of BiLSTM are achieved for action recognition over split1, split2 and split3 respectively (For more detail, see confusion matrix of action model in Figure 4.4).

For both combination types, experimental results show that BiLSTM structure improves the accuracy compared to LSTM structure in Table 4.7. For this reason, we continue with BiLSTM structure for other experiments.

Methodology 3 - LSTM Fusion using Shallow Neural Network. In this experiment, we evaluate action recognition with shallow neural network which consists of 2 fully connected layers which have 512, 256 neurons respectively. In this small network, dropout with a rate of 0.5 is added to each fully connected layer. Finally, it ends with softmax layer. We use Adam Optimizer with learning rate of 0.001 to train. The action model utilizes concatenated softmax values of BiLSTM verb and BiLSTM object models at a clip level. As given in Table 4.7, 45.45% accuracy is obtained over split1.

Methodology 4 - LSTM Baseline Model. LSTM based action model is also experimented utilizing baseline C3D action model. Clip sequence, which is extracted from the softmax layer of baseline model is utilized by BiLSTM action model. BiLSTM architecture is constructed using 1 layer with 100 cell size. For this experiment,

parameters which are 0,001 learning rate, batch size of 30 are utilized. In this experiment, accuracy reached to 23.31% in Table 4.7.

4.5 Comparison with Other Studies and Discussion

In this section, we compare our model performance and model structure to the others in literature that reported results in EGTEA Gaze+ dataset. To the best our knowledge, recent action recognition models [90, 12, 59] in literature are also performed in EGTEA Gaze+ dataset. Compared to our model, different approaches with these models are presented as follows.

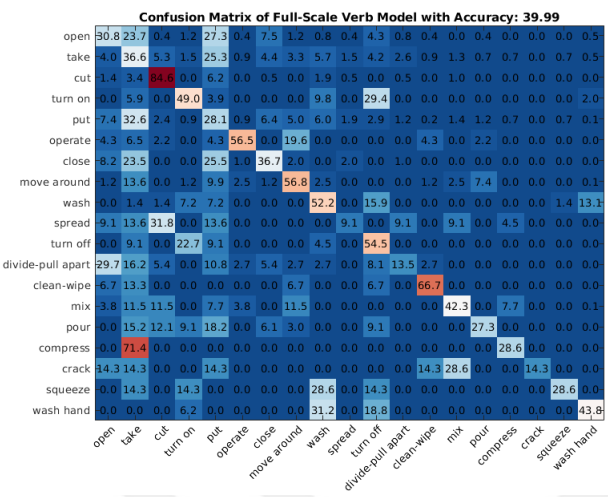
Sudhakaran et al. [90] propose an object-centric deep learning model for first-person action recognition. Similar to our approach, LSTM based structure is utilized to encode the spatio-temporal information in this study. ConvLSTM is proposed [91] model with ResNet34 [62] structure as backbone. ResNet34 network is trained using RGB video images and gives spatial attention map. Spatial attention map provides attention to the region that contains the objects. After training ResNet34, Convolutional LSTM network with ResNet34 backbone network is trained together. It outperforms our model (our model achieves 51.93% overall accuracy over split1) in EGTEA Gaze+ dataset with 60.76% performance for EGTEA Gaze+ dataset. While our model utilizes strong supervision of object and hand bounding boxes to identify the action in video, this model utilize weakly supervision with action class labels.

Another study, which is completely different structure from ours, to recognize the first-person action in EGTEA Gaze+ dataset is presented by Yin Li et al. [12]. The proposed deep learning model based on Regions with CNN (R-CNN) [83] learns the first-person action and gaze information jointly. Their network takes the RGB frames and optical flows of videos as input and gives the attention map, named as gaze map, in stochastic units in the middle layer of the network. Using this attention map in the next layers of the model, the action category is defined and the accuracy of the model is achieved to 53.3% over split1.

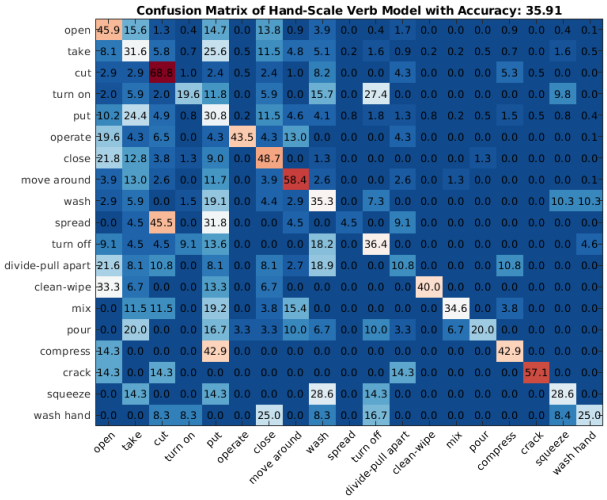
Hahn et al. [59] propose action recognition model which is different from other studies with using textual information from the recipe of the video as well as visual in-

formation. The proposed model consists of three steps which are action proposal, object recognition, and recipe alignment steps. To get the action classes, they use a recipe-video alignment technique based on natural language processing utilizing the object categories obtained from the object recognition step. Object recognition step was constructed using gaze information of dataset. They evaluate their studies with 49.05%, which is so close to our model performance, in EGTEA Gaze+.

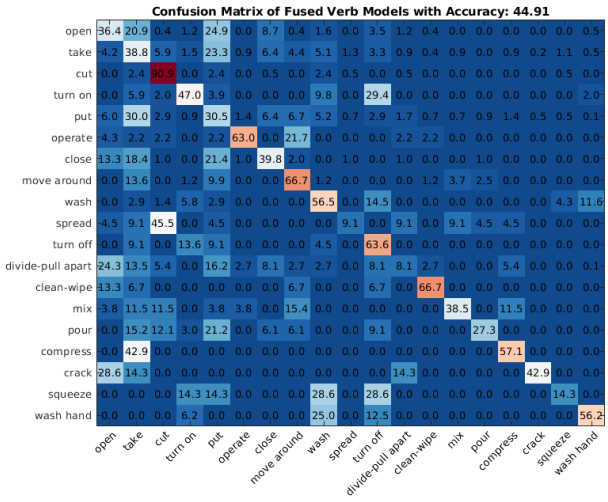
In contrast to our approach, particularly the last two works utilize recipe and gaze information for recognition. To reach gaze information, the camera user should wear an eye-tracking device, and these devices may make the user feel uncomfortable. Moreover, detailed pre-processing is required for recipe and gaze information to be used in action recognition. Although our model does not use gaze and recipe information which are available in the dataset, our performance is especially close to the last study.



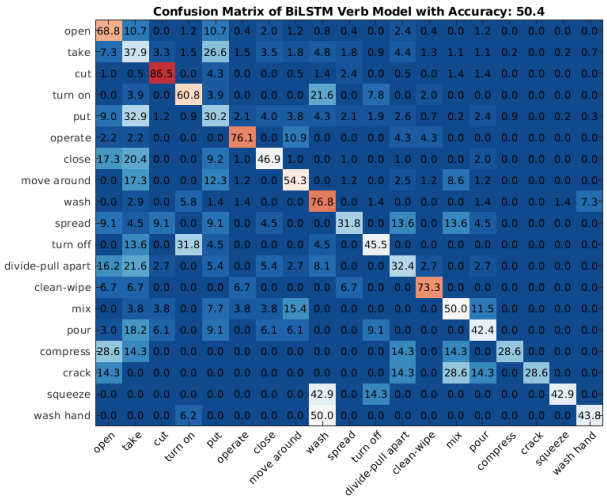
(a) Full-Scale Verb Model



(b) Hand-Scale Verb Model



(c) Fusion of Verb Models



(d) LSTM Verb Model

Figure 4.2: Confusion Matrices of Verb Models over split1. (a) Full Scale Verb Model is constructed over split1 with full vision. (b) Hand scale verb models are constructed over split1 using 10% enlargement respectively. (c) Combined verb models using max-pooling over full-scale and hand-scale model. (d) LSTM based verb model utilizes combined full and hand scale verb softmax values from the verb stream.

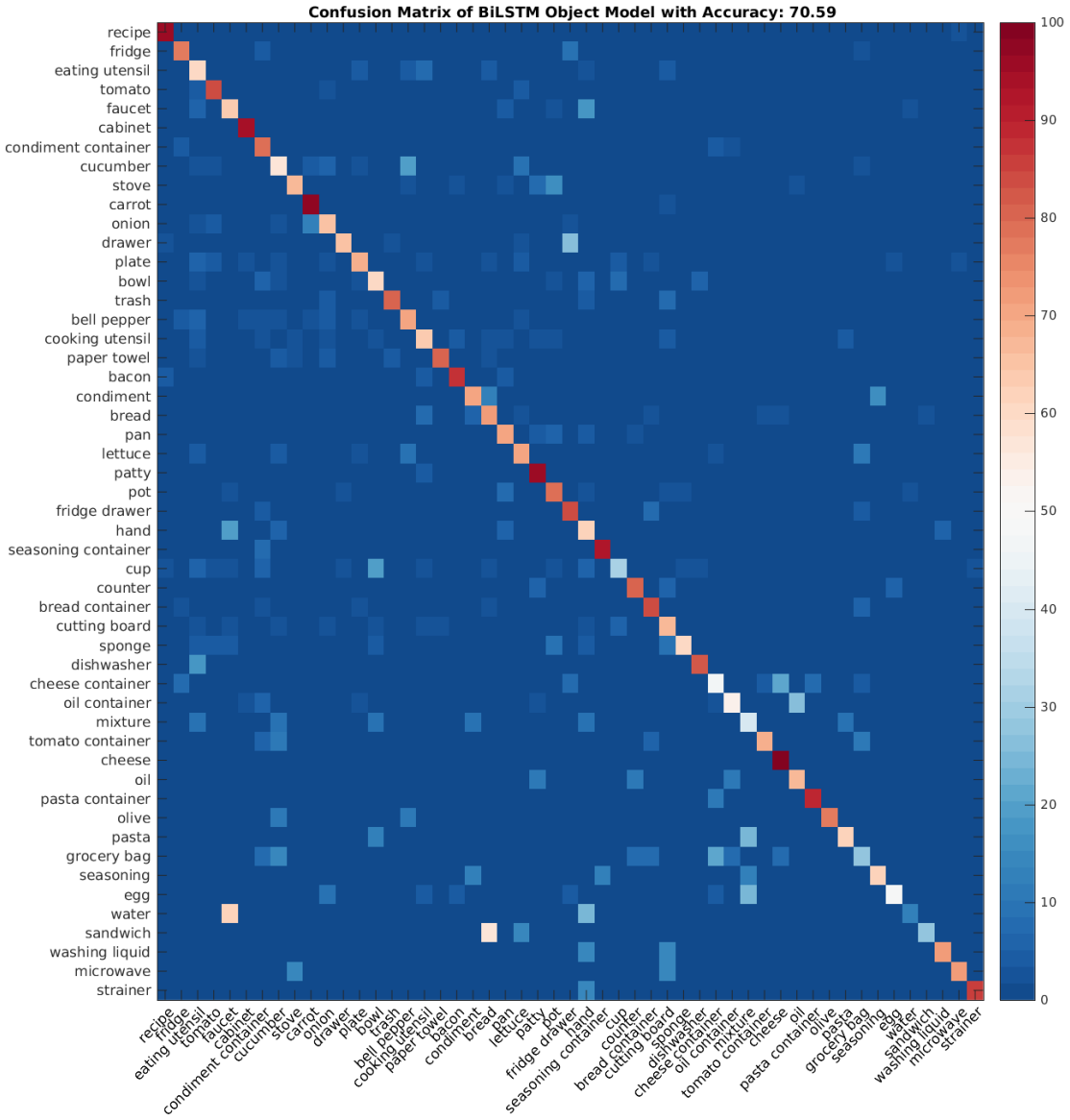


Figure 4.3: Confusion matrix of BiLSTM object model over split1. The 51 out of 53 object classes (excluding trash container and pasta which are not main object in action) are taken into consideration for the object model.

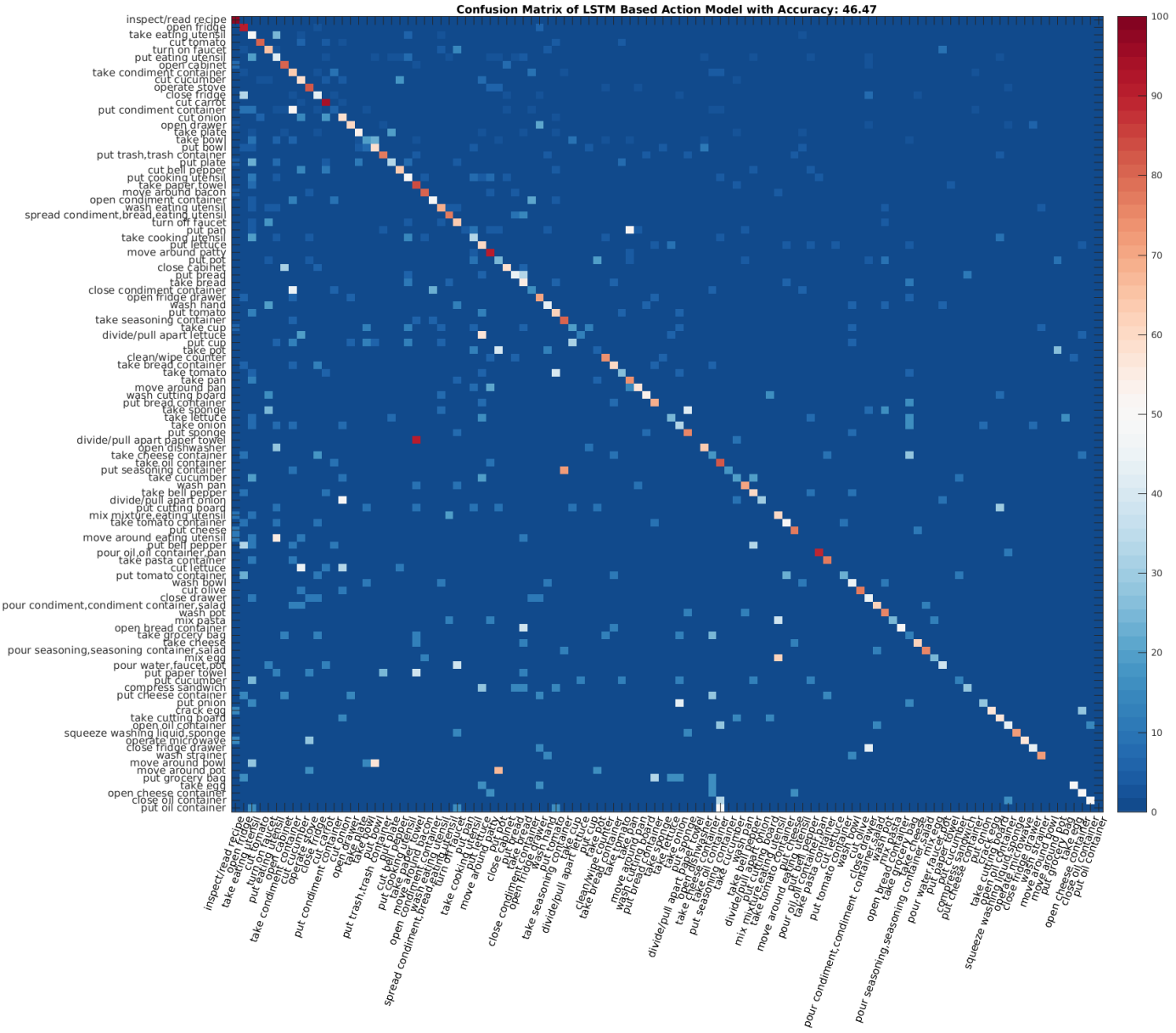


Figure 4.4: Confusion matrix of LSTM based action model. The compositional model which utilizes both verb model (max-pooled full-scale and hand-scale verb models) and the object model is evaluated in split 1.

CHAPTER 5

CONCLUSION

5.1 Summary and Discussion

Action recognition in first-person videos such as *take tomato*, *mix salad*, is more challenging than action recognition which involves the basic action pattern such as *take*, *mix*. Action videos in first-person view videos contain large number of interactions with high similarities, and actions are expressed using human-object interactions corresponding to fine-grained action categories.

Due to human-object interactions, first-person action categories are presented by verb and object pairs. For instance, $take\ tomato$ action can be defined by a combination of $tomato$ as an object and $take$ as a verb occurring in the same video clip. Considering the aforementioned problem definition of fine-grained action recognition in first-person videos, we introduce compositional model including two complementary steps to perform action recognition. The first step is based on construction of verb and object models which are decomposition of actions. Particularly, verb video model as the action representation and object video model as the interaction is represented as two separate pathways. The second step is the fusion stage to identify action category, where distinct verb and object models are combined to give their action judgments. We proposed and applied two different fusion approaches: (1) count-based fusion model with a simple multiplication step and (2) LSTM-based fusion model with a recurrent step collecting verb and object label judgments along a temporal video sequence. We evaluate detection performances for verb, object and action models and we present extensive experimental evaluation for action recognition over fusion approaches on the EGTEA Gaze+ dataset.

In summary,

- Experimental results show that decomposing actions model into verb and object models significantly improves the performance compared to the baseline action model for a large number of action classes.
- Despite its simplicity, count-based fusion model results in acceptable recognition rates and it outperforms the baseline action model without learning. In addition to its simplicity and applicability without the need of learning strategy, it provides detection of previously unseen verb-object pairs.
- Experimental results show that the recurrent networks improve action recognition performance along with verb and object model performances.
- Our compositional model consists of verb and object streams. In this work, we assume that the object is appearing in the field of view to recognize the action. As a disadvantage, our model will fail if the objects do not appear in the field of view and in the first-person videos without human-object interaction.

5.2 Future Work

Our architecture is constructed using various sub-models which are verb model, object model, fusion model to get action category. Different extensions and changes can be applied to improve action recognition as future work.

- In our model, verbs modelling is employed using C3D network structure for spatial-temporal analysis. Verb models can be improved using dense-trajectories that are known as the complementary feature for C3D. This could further improve the recognition accuracies.
- The EGTEA Gaze+ dataset that is the meal preparation dataset also provides the recipe for each meal. The framework can be further improved with an additional text stream.
- Gaze location is an important clue for first-person video analysis. Gaze provides knowledge of the region of interest where the action is performed and the

object is manipulated by hand. This rich information can also be utilized as an additional stream for our framework.

- In order to evidence the power of our proposed model and consistency, our proposed model can be tested on other first-person datasets consists of human-object interaction.

5.3 Related Publication

Z. Gokce, S. Pehlivan "Human Action Recognition in First Person Videos using Verb-Object Pairs", IEEE 27. Signal Processing and Communications Applications Conference (SIU), 2019.

REFERENCES

[1] Y. Li, A. Fathi, and J. M. Rehg, "Learning to predict gaze in egocentric video," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3216–3223, 2013.

[2] A. Fathi, Y. Li, and J. M. Rehg, "Learning to recognize daily actions using gaze," in *European Conference on Computer Vision*, pp. 314–327, Springer, 2012.

[3] A. Fathi and J. M. Rehg, "Modeling actions through state changes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2579–2586, 2013.

[4] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, pp. 4489–4497, 2015.

[5] R. Megret, D. Szolgay, J. Benois-Pineau, P. Joly, J. Pinquier, J.-F. Dartigues, and C. Helmer, "Wearable video monitoring of people with age dementia: Video indexing at the service of helthcare," in *2008 International workshop on content-based multimedia indexing*, pp. 101–108, IEEE, 2008.

[6] J. Lee and M. S. Ryoo, "Learning robot activities from first-person human videos using convolutional future regression," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1–2, 2017.

[7] N. Das, E. Ohn-Bar, and M. M. Trivedi, "On performance evaluation of driver hand detection algorithms: Challenges, dataset, and metrics," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, pp. 2953–2958, IEEE, 2015.

[8] H. Rhodin, C. Richardt, D. Casas, E. Insafutdinov, M. Shafiei, H.-P. Seidel, B. Schiele, and C. Theobalt, "Egocap: egocentric marker-less motion capture

with two fisheye cameras," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 6, p. 162, 2016.

[9] M. Ma, H. Fan, and K. M. Kitani, "Going deeper into first-person activity recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1894–1903, 2016.

[10] S. Saif, S. Tehseen, and S. Kausar, "A survey of the techniques for the identification and classification of human actions from visual data," *Sensors*, vol. 18, no. 11, p. 3979, 2018.

[11] S. Bambach, "A survey on recent advances of computer vision algorithms for egocentric video," *arXiv preprint arXiv:1501.02825*, 2015.

[12] Y. Li, M. Liu, and J. M. Rehg, "In the eye of beholder: Joint learning of gaze and actions in first person video," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 619–635, 2018.

[13] F. Zhu, L. Shao, J. Xie, and Y. Fang, "From handcrafted to learned representations for human action recognition: A survey," *Image and Vision Computing*, vol. 55, pp. 42–52, 2016.

[14] A. G. del Molino, C. Tan, J.-H. Lim, and A.-H. Tan, "Summarization of egocentric videos: A comprehensive survey," *IEEE Transactions on Human-Machine Systems*, vol. 47, no. 1, pp. 65–76, 2016.

[15] G. Cheng, Y. Wan, A. N. Saudagar, K. Namuduri, and B. P. Buckles, "Advances in human action recognition: A survey," *arXiv preprint arXiv:1501.05964*, 2015.

[16] D. Weinland, R. Ronfard, and E. Boyer, "Free viewpoint action recognition using motion history volumes," *Computer vision and image understanding*, vol. 104, no. 2-3, pp. 249–257, 2006.

[17] R. Souvenir and J. Babbs, "Learning the viewpoint manifold for action recognition," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–7, IEEE, 2008.

[18] L. Xia, C.-C. Chen, and J. K. Aggarwal, "View invariant human action recognition using histograms of 3d joints," in *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp. 20–27, IEEE, 2012.

[19] S. Zhang, Z. Wei, J. Nie, L. Huang, S. Wang, and Z. Li, "A review on human activity recognition using vision-based method," *Journal of healthcare engineering*, vol. 2017, 2017.

[20] C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland, "Pfinder: Real-time tracking of the human body," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 19, no. 7, pp. 780–785, 1997.

[21] D. Koller, J. Weber, T. Huang, J. Malik, G. Ogasawara, B. Rao, and S. Russell, "Towards robust automatic traffic scene analysis in real-time," in *Proceedings of 12th International Conference on Pattern Recognition*, vol. 1, pp. 126–131, IEEE, 1994.

[22] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149)*, vol. 2, pp. 246–252, IEEE, 1999.

[23] A. F. Bobick and J. W. Davis, "The recognition of human movement using temporal templates," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 3, pp. 257–267, 2001.

[24] X. Lu, Q. Liu, and S. Oe, "Recognizing non-rigid human actions using joints tracking in space-time," in *International Conference on Information Technology: Coding and Computing, 2004. Proceedings. ITCC 2004.*, vol. 1, pp. 620–624, IEEE, 2004.

[25] B. D. Lucas, T. Kanade, *et al.*, "An iterative image registration technique with an application to stereo vision," 1981.

[26] D. Tran and A. Sorokin, "Human activity recognition with metric learning," in *European conference on computer vision*, pp. 548–561, Springer, 2008.

[27] I. Laptev, "On space-time interest points," *International journal of computer vision*, vol. 64, no. 2-3, pp. 107–123, 2005.

[28] C. G. Harris, M. Stephens, *et al.*, "A combined corner and edge detector.," in *Alvey vision conference*, vol. 15, pp. 10–5244, Citeseer, 1988.

[29] I. Laptev, B. Caputo, *et al.*, "Recognizing human actions: a local svm approach," in *null*, pp. 32–36, IEEE, 2004.

[30] J. C. Niebles, H. Wang, and L. Fei-Fei, "Unsupervised learning of human action categories using spatial-temporal words," *International journal of computer vision*, vol. 79, no. 3, pp. 299–318, 2008.

[31] H. Wang, A. Kläser, C. Schmid, and L. Cheng-Lin, "Action recognition by dense trajectories," in *CVPR 2011-IEEE Conference on Computer Vision & Pattern Recognition*, pp. 3169–3176, IEEE, 2011.

[32] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

[33] R. Chaudhry, A. Ravichandran, G. Hager, and R. Vidal, "Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1932–1939, IEEE, 2009.

[34] N. Dalal, B. Triggs, and C. Schmid, "Human detection using oriented histograms of flow and appearance," in *European conference on computer vision*, pp. 428–441, Springer, 2006.

[35] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105, 2012.

[36] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255, Ieee, 2009.

[37] S. J. Berlin and M. John, "Human interaction recognition through deep learning

network," in *2016 IEEE International Carnahan Conference on Security Technology (ICCST)*, pp. 1–4, IEEE, 2016.

[38] L. Mo, F. Li, Y. Zhu, and A. Huang, "Human physical activity recognition based on computer vision with deep learning model," in *2016 IEEE International Instrumentation and Measurement Technology Conference Proceedings*, pp. 1–6, IEEE, 2016.

[39] S. Ji, W. Xu, M. Yang, and K. Yu, "3d convolutional neural networks for human action recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 1, pp. 221–231, 2013.

[40] J. Zhang, L. Chen, and J. Tian, "3d convolutional neural network for action recognition," in *CCF Chinese Conference on Computer Vision*, pp. 600–607, Springer, 2017.

[41] A. Grushin, D. D. Monner, J. A. Reggia, and A. Mishra, "Robust human action recognition via long short-term memory," in *The 2013 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, IEEE, 2013.

[42] A. Montes, A. Salvador, S. Pascual, and X. Giro-i Nieto, "Temporal activity detection in untrimmed videos with recurrent neural networks," *arXiv preprint arXiv:1608.08128*, 2016.

[43] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Advances in neural information processing systems*, pp. 568–576, 2014.

[44] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International journal of computer vision*, vol. 42, no. 3, pp. 145–175, 2001.

[45] D. G. Lowe, "Object recognition from local scale-invariant features," in *International Conference on Computer Vision*, 1999.

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

[47] A. Fathi, A. Farhadi, and J. M. Rehg, "Understanding egocentric activities," in *2011 International Conference on Computer Vision*, pp. 407–414, IEEE, 2011.

[48] E. H. Spriggs, F. De La Torre, and M. Hebert, "Temporal segmentation and activity classification from first-person sensing," in *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp. 17–24, IEEE, 2009.

[49] F. De la Torre, A. Bargeil, X. Martin, and J. Hodgins, "Guide to the cmu multimodal activity (cmu-mmac) database," *Technical Report CMU-RI-TR-08-22, Robotics Institute, Carnegie Mellon University*, 2008.

[50] H. Pirsiavash and D. Ramanan, "Detecting activities of daily living in first-person camera views," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2847–2854, IEEE, 2012.

[51] R. E. Schapire, "Explaining adaboost," in *Empirical inference*, pp. 37–52, Springer, 2013.

[52] A. Cartas, P. Radeva, and M. Dimiccoli, "Contextually driven first-person action recognition from videos,"

[53] J. Pont-Tuset, P. Arbelaez, J. T. Barron, F. Marques, and J. Malik, "Multiscale combinatorial grouping for image segmentation and object proposal generation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 1, pp. 128–140, 2017.

[54] G. Gkioxari, R. B. Girshick, and J. Malik, "Contextual action recognition with r*cnn," *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1080–1088, 2015.

[55] K. M. Kitani, T. Okabe, Y. Sato, and A. Sugimoto, "Fast unsupervised ego-action learning for first-person sports videos," in *CVPR 2011*, pp. 3241–3248, IEEE, 2011.

[56] Y. Li, Z. Ye, and J. M. Rehg, "Delving into egocentric actions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 287–295, 2015.

[57] K. Simonyan and A. Zisserman, "Advances in neural information processing systems," 2014.

[58] Y. Tang, Y. Tian, J. Lu, J. Feng, and J. Zhou, "Action recognition in rgb-d egocentric videos," in *2017 IEEE International Conference on Image Processing (ICIP)*, pp. 3410–3414, IEEE, 2017.

[59] M. Hahn, N. Ruiz, J.-B. Alayrac, I. Laptev, and J. M. Rehg, "Learning to localize and align fine-grained actions to sparse instructions," *arXiv preprint arXiv:1809.08381*, 2018.

[60] G. Kapidis, R. Poppe, E. van Dam, L. P. Noldus, and R. C. Veltkamp, "Egocentric hand track and object-based human action recognition," *arXiv preprint arXiv:1905.00742*, 2019.

[61] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," *arXiv preprint arXiv:1405.3531*, 2014.

[62] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[63] M.-C. De Marneffe, B. MacCartney, C. D. Manning, *et al.*, "Generating typed dependency parses from phrase structure parses.," in *Lrec*, vol. 6, pp. 449–454, 2006.

[64] D. Damen, H. Doughty, G. Maria 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)*, pp. 720–736, 2018.

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

[66] "Georgia tech egocentric activity datasets." `http://www.cbi.gatech.edu/fpv/`. Accessed: 2019-06-14.

[67] D. Damen, T. Leelasawassuk, O. Haines, A. Calway, and W. W. Mayol-Cuevas, "You-do, i-learn: Discovering task relevant objects and their modes of interaction from multi-user egocentric video.," in *BMVC*, vol. 2, p. 3, 2014.

[68] D. Damen, H. Doughty, G. M. Farinella, S. Fidler, A. Furnari, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price, and M. Wray, "Scaling egocentric vision: The epic-kitchens dataset," in *European Conference on Computer Vision (ECCV)*, 2018.

[69] Y. J. Lee, J. Ghosh, and K. Grauman, "Discovering important people and objects for egocentric video summarization," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1346–1353, IEEE, 2012.

[70] Z. Lu and K. Grauman, "Story-driven summarization for egocentric video," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2714–2721, 2013.

[71] G. A. Sigurdsson, A. Gupta, C. Schmid, A. Farhadi, and K. Alahari, "Charades-ego: A large-scale dataset of paired third and first person videos," *arXiv preprint arXiv:1804.09626*, 2018.

[72] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, *et al.*, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[73] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[74] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, 2015.

[75] S. Ji, W. Xu, M. Yang, and K. Yu, "3d convolutional neural networks for human action recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 1, pp. 221–231, 2012.

[76] M. C. Mozer, "A focused backpropagation algorithm for temporal," *Backpropagation: Theory, architectures, and applications*, vol. 137, 1995.

[77] A. Robinson and F. Fallside, *The utility driven dynamic error propagation network*. University of Cambridge Department of Engineering, 1987.

[78] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[79] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional lstm and other neural network architectures," *Neural Networks*, vol. 18, no. 5-6, pp. 602–610, 2005.

[80] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.

[81] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

[82] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7263–7271, 2017.

[83] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580–587, 2014.

[84] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, pp. 1440–1448, 2015.

[85] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, pp. 91–99, 2015.

[86] "C3d-tensorflow implementation." `https://github.com/hx173149/C3D-tensorflow`. Accessed: 2019-06-14.

[87] J. Redmon and A. Farhadi, "Yolo9000: Better, faster, stronger," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6517–6525, 2017.

[88] Y. C. Zhang, Y. Li, and J. M. Rehg, "First-person action decomposition and zero-shot learning," in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 121–129, IEEE, 2017.

[89] L. Shen, S. Yeung, J. Hoffman, G. Mori, and L. Fei-Fei, "Scaling human-object interaction recognition through zero-shot learning," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1568–1576, IEEE, 2018.

[90] S. Sudhakaran and O. Lanz, "Attention is all we need: Nailing down object-centric attention for egocentric activity recognition," *arXiv preprint arXiv:1807.11794*, 2018.

[91] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," in *Advances in neural information processing systems*, pp. 802–810, 2015.