EGOCENTRIC ACTION UNDERSTANDING BY LEARNING EMBODIED ATTENTION

A Dissertation
Presented to
The Academic Faculty

By

Miao Liu

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in
Robotics

Georgia Institute of Technology

August 2022

EGOCENTRIC ACTION UNDERSTANDING BY LEARNING EMBODIED ATTENTION

Thesis committee:

Professor James M. Rehg, Advisor
School of Interactive Computing
Georgia Institute of Technology

Professor James Hays
School of Interactive Computing
Georgia Institute of Technology

Professor Diyi Yang
School of Interactive Computing
Georgia Institute of Technology

Professor Jitendra Malik
Department of Electrical Engineering and Computer Science
University of California at Berkeley

Professor Zsolt Kira
School of Interactive Computing
Georgia Institute of Technology

Date approved: June 30th, 2022

ACKNOWLEDGMENTS

While I'm writing this chapter and contemplating the past five years, I'm surprised about how much help I received during this humble journey. I want to express my sincere gratitude to those who have influenced, inspired, and enlightened my research.

First, I want to thank my wonderful advisor– Prof. James M. Rehg. Jim teaches me what it means to do high-quality research. I also have the privilege to see how he hard works, and learn what it takes to succeed in the science world. Because of his support, guidance, and patience, not only have I learned to become a maturer researcher, but I have also learned to become a better person.

I also want to thank the rest of my thesis committee: Prof. Diyi Yang, Prof. Zsolt Kira, Prof. James Hays, and Prof. Jitendra Malik, for the insightful comments and suggestions on my thesis work.

In addition, I want to thank Prof. Yin Li for the continuous guidance on my research and for "babysitting" me from day one; I want to thank our lab alumni Dr. Zhaoyang Lv for the 101 lecture on networking in academia and for helping me connect with researchers from all over the world; I want to thank Prof. Siyu Tang for the enlightenment during my internship at ETH Zürich and Max-Planck Institute, not surprisingly, the visit to her group literally expands the "depth" of my research; I want to thank my mentors and collaborators during my internship at Facebook Reality Lab: Dr. Chao Li, Dr. Lingni Ma, Dr. Kiran Somasundaram, and Prof. Kristen Grauman, for the countless support; I want to thank my lab fellows for the active research discussion and social conversation.

My thanks also go to my family and friends. Their loving and caring make my Ph.D. journey, designed to be challenging, less challenging.

TABLE OF CONTENTS

Acknowledgments iii

List of Tables viii

List of Figures xi

Summary xv

Chapter 1: Introduction 1

1.1 Thesis Statement 2

1.2 Thesis Outline 3

1.2.1 Attention and Actions in Egocentric Video 3

1.2.2 Egocentric Activity Recognition and 3D Localization 3

1.2.3 Human-Object Interaction Anticipation in Egocentric Video 4

1.2.4 Future Hand Segmentation in Egocentric Video 4

1.3 Contributions 5

Chapter 2: Related Work 6

2.1 Egocentric Vision 6

2.2 Action Recognition 8

2.3 Visual Attention 9

2.4 Human-Scene Interaction 10

2.5 Human Body Motion Forecasting 12

Chapter 3: Attention and Actions in Egocentric Video 13

3.1 Introduction 13

3.2 Joint Modeling of Gaze and Actions 16

3.2.1 Model Overview 16

3.2.2 Modeling Gaze with Stochastic Units 17

3.2.3 Training and Inference 18

3.2.4 Approximate Inference 19

3.3 Attention Distillation for Learning Video Representations 19

3.3.1 Model Overview and Key Components 20

3.3.2 Attention Generation 21

3.3.3 Attention Guided Recognition 22

3.3.4 Attention Distillation 22

3.3.5 Our Full Model 23

3.4 Experiment and Results 23

3.4.1 Gaze and Actions 23

3.4.2 Attention Distillation 28

3.5 Conclusion 35

Chapter 4: Egocentric Activity Recognition and 3D Localization 37

4.1 Introduction 37

4.2 Method 39

4.2.1 Joint Modeling with the 3D Environment Representation 40
4.2.2 3D Action Localization 42
4.2.3 Action Recognition with Environment Prior 43
4.2.4 Training and Inference 43
4.3 Experiments and Results 44
4.3.1 Dataset and Benchmarks 45
4.3.2 Action Understanding in Seen Environments 47
4.3.3 Ablation Studies 49
4.3.4 Generalization to Novel Environment 51
4.3.5 Discussion 52
4.4 Conclusion 53

Chapter 5: Human-Object Interaction Anticipation in Egocentric Video 54
5.1 Introduction 54
5.2 Problem Setting and Model Overview 56
5.3 Joint Modeling of Human-Object Interaction 58
5.3.1 Motor Attention Module 58
5.3.2 Interaction Hotspots Module 59
5.3.3 Anticipation Module 60
5.3.4 Training and Inference 60
5.4 Experiment and Results 62
5.4.1 Dataset and Benchmark 62
5.4.2 Action Anticipation Results on EPIC-Kitchens 63

5.4.3 Ablation Study 65

5.4.4 Remarks and Discussion 69

5.5 Conclusion 69

Chapter 6: Future Hand Segmentation in Egocentric Video 71

6.1 Introduction 71

6.2 Method 73

6.2.1 3D Fully Convolutional Network 74

6.2.2 Generative Adversarial Network 75

6.2.3 Full Model of EgoGAN 76

6.3 Experiments 77

6.3.1 Dataset and Metrics 78

6.3.2 Model Ablations and Analysis 79

6.3.3 Comparison to State-of-the-Art Methods 83

6.3.4 Discussion 85

6.4 Conclusion 86

Chapter 7: Conclusion and Future Work 87

7.1 Conclusion 87

7.2 Future Work 88

References 90

LIST OF TABLES

3.1 **Ablation study on backbone networks and probabilistic modeling**. We show F1 scores for gaze estimation and mean class accuracy for action recognition. 24

3.2 **Action Recognition and Gaze Estimation**. For action recognition, we report mean class accuracy at both clip and video level. For gaze estimation, we show F1 scores and their corresponding precision and recall scores. . . . 26

3.3 **Evaluations of attention modules**. We compared 3 different design choices with RGB/flow stream on three datasets. Prob-Atten provides a consistent performance boost on both streams and across datasets. 29

3.4 **Action recognition results on UCF101 and HMDB51 datasets**. We compare the results of our model with previous works. Our model outperforms state-of-the-art methods that use only RGB stream and the same input sequence length by $\sim 1\%$. *For fair comparison, we report results of I3D models that use 24 frames as inputs–the same as our model. 30

3.5 **Action recognition results on on 20BN-V2 dataset [118]**. Our model achieves the best performance among networks that uses RGB frames. Fusing our model with a flow network also outperforms two stream baseline by a significant margin. 31

3.6 **Results of action localization using attention maps on THUMOS'13 localization test set [122]**. We report the best F1 score and its precision and recall. Our motion attention outperforms all baselines that are trained with only action labels. 33

3.7 **Inverting the arrow of time for action recognition**. We train the models on normal samples, yet test them on videos with reversed temporal order. A large performance drop indicates that the model has to rely on motion information for the recognition. 34

4.1 **Comparison with other forms of environment context**. Our Hierarchical Volumetric Representation (HVR) outperforms other methods by a significant margin on both action recognition and 3D action localization. The best results are highlighted with **boldface**, and the second-best results are underlined. 46

4.2 **Ablation study for the 3D representation**. We present the results of our method that adopts different semantic occupancy map resolution M. 49

4.3 **Ablation study for joint modeling of action category and 3D action location**. Our proposed probabilistic joint modeling can consistently benefit the performance on action recognition and 3D action localization. 50

4.4 **Experimental results on unseen environment split**. Our model show the capacity of better generalizing to an unseen environment with known 3D map. The best results are highlighted with **boldface**, and the second-best results are underlined. 51

5.1 **Action anticipation results on Epic-Kitchens**. Ours+Obj model outperforms state-of-the-art by a notable margin. See discussions of Ours+Obj in Chapter 5.4.2. 63

5.2 **Comparison between our methods and previous state-of-the-art results RULSTM.** See Chapter 5.4.2 of our submission for discussion of Ours+Obj. 63

5.3 **Ablation study for action anticipation**. We compare our model with backbone I3D network, and further analyze the role of motor attention prediction, interaction hotspots estimation, and stochastic units in joint modelling. See discussions in Chapter 5.4.3. 65

5.4 **Ablation study for interaction hotspots estimation**. Jointly modeling motor attention with stochastic units can greatly benefit the performance of interaction hotspots estimation. (↑/↓ indicates higher/lower is better) 65

5.5 **Motor attention prediction results on EGTEA**. Our model compares favourably to strong baselines. (↑/↓ indicates higher/lower is better) 67

6.1 **Analysis of variations in our approach**. We conduct detailed ablation studies to validate our model design, and further show the results of variations of our method to demonstrate the benefits of using the GAN for modeling future head motion. *: HeadDir takes future head motions as additional input modalities at inference time, which in fact violates the future anticipation setting (See more discussion in Chapter 6.3.2.). The best results are highlighted with **boldface**. 79

6.2 **Experimental results on generated future head motion**. We calculate the endpoint error (EPE) between the generated head motion and the ground truth head motion. Our method outperforms HeadReg on the EPIC-Kitchens dataset and works on-par with HeadReg on the EGTEA dataset. 82

6.3 **Experimental results using different backbone networks**. Our model achieves consistent performance improvement when using different backbone networks. (See more discussion in Section 6.3.2.) 82

6.4 **Comparison with previous state-of-the-art methods on future image segmentation**. Our results consistently outperform the second-best results (across all methods) by +1.3% on EPIC-Kitchens and +0.7% on EGTEA in average F1 score. *: We re-implement the model to take raw video frames as inputs as our method (See more discussion in Chapter 6.3.3). The best results are highlighted with **boldface**, and the second-best results are <u>underlined</u>. 83

LIST OF FIGURES

3.1 **RGB and flow networks attend to different aspects of an action, yet both are essential for recognition**. (a) Attention maps from RGB and flow streams of I3D [45] by Grad-Cam [97]. (b) Attention maps from our attention distillation model. Our model jointly infers appearance and motion attention from only RGB frames, improves the performance of the RGB stream, and is significantly more efficient than two-stream models. . . 14

3.2 **Proposed joint model of gaze and action**. Our network takes multiple RGB and flow frames as inputs, and outputs a set of parameters defining a distribution of gaze in the middle layers. We then sample a gaze map from this distribution. This map is used to selectively pool visual features at higher layers of the network for action recognition. During training, our model receives action labels and noisy gaze measurement. 17

3.3 **Proposed attention distillation method**. Our model (c) takes multiple RGB frames as inputs and adopts a 3D convolutional network as the backbone. It outputs two attention maps using the attention module (b), based on which the action labels are predicted. The motion map is learned by mimicking the attention from a reference flow network (a). The appearance map is learned to highlight discriminative regions for recognition. These two maps are used to create spatio-temporal feature representations for action recognition. 20

3.4 **Visualization of our gaze estimation and action recognition**. We plot the output gaze heat map and print the predicted action labels and ground-truth labels. Both successful (first row) and failure cases (second row) are presented. 28

3.5 **Visualization of attention maps (Ours vs. Soft-Atten using the same I3D backbone)**. For each video clip, we re-interpolate the attention maps and plot them on the first and last frame. Red regions indicate higher value of attention. Our model produces appearance and motion attention maps that are qualitatively different and index key action regions. 36

4.1 (a) **Motivation**:The activities of daily life take place in a 3D environment, and the semantic and spatial properties of the environment are powerful cues for activity recognition. (b) **Our Proposed Task**: Given an input egocentric video sequence and a 3D volumetric representation of the environment (carrying both semantic and geometric information), our goal is to detect and localize activities, by jointly predicting the action label and location on the 3D map where it occurred. 38

4.2 (a) **Hierarchical Volumetric Representation (HVR)**. We rasterize the semantic 3D environment mesh into two levels of 3D voxels. Each parent voxel corresponds to a possible action location, while the children voxels compose a semantic occupancy map that describes their parent voxel. (b) **Overview of our model**. Our model takes video clips x and the associated 3D environment representation e as inputs. We adopt an I3D backbone network ϕ to extract video features and a 3D convolutional network ψ to extract the global environment features. We then make use of stochastic units to generate sampled action location $\tilde{r}$ for selecting local 3D environment features for action recognition. Note that $\otimes$ represents weighted average pooling, while $\oplus$ denotes concatenation along channel dimension. 40

4.3 **Visualization of predicted 3D action location (projected on top-down view of the reconstructed 3D scene) and action labels (captions above the video frames)**. We present both successful and failure examples. We also show the "zoom-in" spatial region of the action location to help readers to better interpret our action localization results. 52

5.1 **What is the most likely future interaction**? Our model takes advantage of the connection between motor attention and visual perception. In addition to future action label, our model also predicts the interaction hotspots on the last observable frame and hand trajectory (in the order of yellow , green, cyan, and magenta) between the last observable time step to action starting point. Visualizations of hand trajectory are projected to the last observable frame (best viewed in color). 55

5.2 **Proposed model for forecasting egocentric human-object interaction**. A 3D convolutional network $\phi(x)$ is used as our backbone network, with features from its ith convolution block as $\phi_i(x)$ (a). A motor attention module (b) makes use of stochastic units to generate sampled future hand trajectories $\tilde{\mathcal{M}}$ used to guide interaction hotspots estimation in module (c). Module (c) further generates sampled interaction hotspots $\tilde{\mathcal{A}}$ with similar stochastic units as in module (b). Both $\tilde{\mathcal{M}}$ and $\tilde{\mathcal{A}}$ are used to guide action anticipation in anticipation module (d). During testing, our model takes only video clips as inputs, and predicts motor attention, interaction hotspots, and action labels. Note that $\otimes$ represents element-wise multiplication for weighted pooling. 57

5.3 **Visualization of motor attention (left image), interaction hotspots (right image), and action labels (captions above the images) on sample frames from EGTEA (first row) and EPIC-Kitchens (second row)**. Both successful (green label) and failure cases (red label) are shown. Future hands position are predicted at every 8 frames and plotted on the last observable frame with the order of yellow, green, cyan, and magenta. 70

6.1 **Future hand segmentation task**: Given an input egocentric video, our goal is to predict a time series of future hand masks in the anticipation video segment. Δ_1, Δ_2, and Δ_3 represent the short-term, middle-term, and long-term time points in the anticipation segment, respectively. The entanglement between drastic head motion and non-rigid hand movements poses a significant technical barrier in computer vision. Here, we visualize our results on this challenging task (best viewed in color). 72

6.2 **Overview of our proposed EgoGAN model**. Our network takes multiple egocentric video frames as the inputs, and outputs future hand masks at different time steps. It is composed of a *3D Fully Convolutional Network (3DFCN)* and a *Generative Adversarial Network (GAN)*. The Encoder Network ϕ_E in the *3DFCN* extracts video features from the input frames, and is then separated into two branches: (1) encoded feature $\phi_E(x)$ is fed into the *Generator (G)* in for generating fake future head motion m_g, and a *Discriminator (D)* is trained to distinguish the generated future head motion from the real ones; (2) m_g is concatenated to $\phi_E(x)$ and the concatenated tensor are then fed into the Decoder Network ϕ_D in *3DFCN*. Finally, the encoder features are further combined with corresponding decoder features using skip connections for future hand mask prediction. 74

6.3 **Visualization of our results**. From left to right, each column presents the future hand segmentation results of short-term ($t + \Delta_1$), middle-term ($t+\Delta_2$), and long-term($t+\Delta_3$) time steps from the EPIC-Kitchens dataset. Predictions from our method *EgoGAN* and the best baseline *FlowTrans* are presented in each sample. (See more discussion in Chapter 6.3.4) 85

Summary

Videos captured from wearable cameras, known as egocentric videos, create a continuous record of human daily visual experience, and thereby offer a new perspective for human activity understanding. Importantly, egocentric video aligns gaze, embodied movement, and action in the same "first-person" coordinate system. The rich egocentric cues reflect the attended scene context of an action, and thereby provide novel means for reasoning human daily routines.

In my thesis work, I describe my efforts on developing novel computational models that learn the embodied egocentric attention for the automatic analysis of egocentric actions. First, I introduce a probabilistic model for learning gaze and actions in egocentric video and further demonstrate that attention can serve as a robust tool for learning motion-aware video representation. Second, I develop a novel deep model to address the challenging problem of jointly recognizing and localizing actions of a mobile user on a known 3D map from egocentric videos. Third, I present a novel deep latent variable model that makes use of human intentional body movement (motor attention) as a key representation for forecasting human-object interaction in egocentric video. Finally, I propose a novel task of future hand segmentation from egocentric videos, and show how explicitly modeling the future head motion can facilitate future hand movement forecasting.

CHAPTER 1
INTRODUCTION

Video captured from a "first-person" egocentric point of view has emerged to offer a new perspective for human activity understanding. The automatic analysis of egocentric video is often referred to as egocentric vision. Recently, the fast commercialization of wearable cameras and people's growing interest in sharing their daily life on social media have enabled the collection of large-scale egocentric video dataset, and thereby facilitate the research in egocentric vision. Moreover, egocentric activity understanding has become a prevailing research topic, because of its potential application on Augmented Reality (AR) and Robotics. For example, the AR experience can benefit from an accurate understanding of the user's perception, attention, and actions; a mobile robot's efficiency can be improved by imitating how humans explore and exploit the environment from egocentric videos.

Besides the practical applications, egocentric video has several unique properties, which point to exciting research directions. Importantly, egocentric video enables the recording of human daily visual experience while the camera wearer interacts with the world, and thereby couples visual perception with actions. Notably, the camera viewpoint is guided by the head movement of the camera wearer and is thus driven by visual attention. Therefore, the visual signals in the eyes of the "beholder" reveal how the camera wearer attends to the scene context during daily activities. In addition, the egocentric video reflects the camera wearer's presence in the 3D world, which links actions with the surrounding 3D spatial scene context. Furthermore, the intentional body movements provide powerful cues for predicting future actions from egocentric videos.

The motor behaviors incorporated in the egocentric video serve as novel tools for understanding how humans interact with the environment. The primary goal of my thesis work is to model egocentric representations that attend to the meaningful scene context for un-

derstanding egocentric actions. Computational models of attention mechanism have been developed for different tasks by different communities. The Transformer architecture [1] learns to implicitly identify important features of the input data via scaled dot-product attention units. Another body of literature [2, 3] seeks to develop visual attention models that predict gaze behavior as developed in psychology research. The egocentric attention representation investigated in my thesis is not limited to the gaze data. We argue that attention under the egocentric vision paradigm is *embodied*: Apart from the gaze behavior, the sensory-motor behavior in the form of head and hand movements also reflect how human attends to the scene context during daily actions.

Previous works [4, 5, 6] explored egocentric cues such as hand masks, gaze measurements, and object features in designing computational models for recognizing egocentric actions. A weakness of these approaches is that additional egocentric cues are needed for the model to make an inference. In order to address this weakness, we factorize the egocentric attention as a latent representation, so that the model can automatically discover the embodied egocentric cues for action understanding from only video frames. Furthermore, prior egocentric works have primarily considered the attention mechanism as a saliency map defined on the image plane. In my thesis work, I further explore how the attended 3D environment context can facilitate action recognition and how embodied motor attention can characterize the future representation for action anticipation. Going beyond predicting the action category, I also examine the novel problem of predicting the detailed shape of future hand movements during human daily routines by explicitly modeling how visual attention drives intentional head movements.

1.1 Thesis Statement

Embodied egocentric attention provides an effective tool for addressing the tasks of action recognition, localization, and anticipation, and hand motion forecasting.

1.2 Thesis Outline

My thesis work can be organized into four main topics: *Attention and Actions in Egocentric Video* (Chapter 3), *Egocentric Activity Recognition and 3D Localization* (Chapter 4), *Human-Object Interaction Anticipation in Egocentric Video* (Chapter 5), *Future Hand Segmentation in Egocentric Video* (Chapter 6).

1.2.1 Attention and Actions in Egocentric Video

The starting point of my dissertation is to develop a visual attention model for recognizing egocentric actions. In this chapter, I first address the task of joint gaze estimation and action recognition in egocentric video. Our method describes the participant's gaze as a probabilistic variable and models its distribution using stochastic sampling units in a deep network, and uses sampled gaze attention map to guide the aggregation of visual features in action recognition, thereby providing coupling between gaze and action. Furthermore, I demonstrate how to leverage the predicted attention map as a vehicle to distill the learned motion representation from a reference flow network to a student RGB network. Our attention distillation method thereby addresses the challenging problem of learning motion representations from only RGB frames.

This work demonstrates the benefits of using probabilistic units to account for the uncertainty within the supervisory signal of eye gaze data, and thereby significantly improves the performance of the joint model of gaze and actions. Moreover, the attention mechanism provides a robust tool for learning motion-sensitive representation from both exocentric and egocentric videos. More details of this work are presented in Chapter 3.

1.2.2 Egocentric Activity Recognition and 3D Localization

The second problem addressed in my thesis work is how to leverage the attended surrounding 3D scene context to jointly recognize and localize the actions of a mobile user on a

known 3D map from egocentric videos. To this end, I propose a novel two-stream deep probabilistic model. Our model takes a Hierarchical Volumetric Representation (HVR) of the 3D environment and an egocentric video as inputs, infers the 3D action location as a latent variable, and recognizes the action based on the video and 3D contextual cues surrounding its potential locations.

Our method demonstrates strong results on both action recognition and 3D action localization across seen and unseen environments. We believe our work points to an exciting research direction at the intersection of egocentric vision and 3D scene understanding. More details of this work are presented in Chapter 4.

1.2.3 Human-Object Interaction Anticipation in Egocentric Video

Besides action recognition, my thesis work also studies the challenging problem of action anticipation by factorizing hand movements as attentional cues. We observe that the intentional hand movement reveals critical information about the future activity. Motivated by this observation, I adopt intentional hand movement as a feature representation, and propose a novel deep network that jointly models and predicts the egocentric hand motion, interaction hotspots, and future action. Specifically, we consider the future hand motion as the motor attention, and use the predicted motor attention to select the discriminative spatial-temporal visual features for predicting actions and interaction hotspots.

Through extensive experiments, I demonstrate that motor attention can significantly benefit action anticipation performance. Moreover, our method also has the capability of accurately predicting future hand movements and interaction hotspots. More details of this work are presented in Chapter 5.

1.2.4 Future Hand Segmentation in Egocentric Video

The last piece of my thesis work utilizes the future head motion as an embodied representation to address the problem of pixel-wise visual anticipation under the challenging

egocentric setting. I introduce the novel problem of anticipating a time series of future hand masks from egocentric video. To model the stochasticity of future head motions, we propose a novel deep generative model – EgoGAN, which uses a 3D Fully Convolutional Network to learn a spatio-temporal video representation, generates future head motion using Generative Adversarial Network (GAN), and then predicts the future hand masks based on the video representation and the generated future head motion.

Though the head motion can not be used as a conventional attention map for selecting salient visual features, it still reflects how the visual attention drives the scene context change. Our experiments suggest that explicitly modeling the future head motion with GAN can improve the performance of future hand segmentation by a notable margin. More details of this work are presented in Chapter 6.

1.3 Contributions

My dissertation makes the following contribution:

- I propose a novel joint model for learning gaze and actions from egocentric video and show that the attention mechanism can serve as a robust tool for learning a motion-aware video representation. Our work provides valuable insights into attention-based recognition, and a step towards learning spatio-temporal video features.
- I demonstrate the 3D geometric and semantic context of the surrounding environment provides critical information that complements video features for egocentric action understanding.
- I introduce a novel deep latent variable model that makes motor attention a first-class player for forecasting human-object interactions.
- I propose to use a generative adversarial network that explicitly models the underlying distribution of possible future head motion to address the novel task of future hand mask forecasting from egocentric videos.

CHAPTER 2
RELATED WORK

In this chapter, I discuss the relevant literature of my thesis work. These previous works are organized as follows:

- *Egocentric Vision* Section provides a thorough survey of research progress on egocentric action recognition, anticipation, gaze estimation, etc.
- *Action Recognition* Section describes previous works on learning spatial-temporal video representation for action recognition.
- *Visual Attention* Section reviews previous works on developing attention mechanism for visual recognition.
- *Human-Scene Interaction* Section reviews the literature on understanding human activity in the context of environmental cues.
- *Human Body Motion Forecasting* Section discusses previous efforts on designing computation models for anticipating the human body motion.

2.1 Egocentric Vision

The advent of wearable cameras has led to growing interest in egocentric vision (see massive-scale egocentric video dataset [7] and recent surveys in [8, 9]). In this section, I review prior works on egocentric action recognition and anticipation, as well as gaze estimation and hand analysis.

Egocentric action understanding has been the subject of many recent efforts. Earlier works adopt hand crafted features for recognizing egocentric actions. Spriggs et al. [10]

proposed to segment and recognize daily activities using a combination of video and wearable sensor data. Kitani et al. [11] used a global motion descriptor to discover egocentric actions. Fathi et al. [12] presented a joint model of objects, actions and activities. Pirsiavash and Ramanan [13] further advocated for an object-centric representation of FPV activities. Novel deep models were further developed for egocentric action recognition. Ryoo et al. [14] developed a novel pooling method for understanding egocentric videos using deep models. Poleg et al. [15] used temporal convolutions on motion fields for long-term activity recognition. Sudhakaran et al. [16] developed a two-stream LSTM model with an attention mechanism for egocentric action recognition Kazakos et al. [17] proposed to fuse video and audio signals for egocentric action recognition. Wray et al. [18] made use of text descriptions of egocentric actions for zero-shot learning. Another line of work considered the problem of egocentric action anticipation.Shen et al. [19] investigated how different egocentric modalities affect the action anticipation performance. Wei et al. [20] utilized a probabilistic model to infer 3D human attention and intention. Tagi et al. [21] addressed a novel task of predicting the future locations of an observed subject in egocentric videos. Ryoo et al. [22] proposed a novel method to summarize pre-activity observations for robot-centric activity prediction. Soran et al. [23] adopted the Hidden Markov Model to compute the transition probability among sequences of actions. A similar idea was explored in [24]. Furnari et al. [25] considered the task of predicting the next-active objects. Their recent work [26] proposed to factorize the anticipation model into a "Rolling" LSTM that summarizes the past activity and an "Unrolling" LSTM that makes hypotheses of the future activity. Ke et al. [27] proposed a time-conditioned skip connection operation to extract relevant information for action anticipation. Previous efforts also modeled the conversations [28] and reactions [29, 30] in social interactions.

A rich set of literature seeks to understand various egocentric cues. Li et al. [31] estimated egocentric gaze using hand and head cues. They [4] further showed the benefits of gaze-indexed visual features in a comprehensive benchmark. Dessalene et al. [32] fo-

cused on predicting the hand-object interaction region of an action. Fathi et al. [5] utilized hand-eye coordination to design a probabilistic model for gaze estimation. Zhang et al. [33] predicted future gaze by estimating gaze from predicted future frames. Both Singh et al. [34] and Ma et al. [6] explored the use of multi-stream networks to capture egocentric attention. Li et al. [35] addressed the egocentric hand detection task by posing the problem as a model recommendation task. Huang et al. [36] adopted an unsupervised clustering algorithm to learn common grasping modes from egocentric videos. Recently, Cai et al. [37] proposed a Bayesian-based domain adaptation framework for hand segmentation on egocentric video frames. My thesis work shares the same motivation of utilizing egocentric cues to reason about human daily actions. Importantly, our works are the first to model the egocentric cues as latent representations for egocentric action understanding. In Chapter 3, I propose a novel deep latent variable model for joint learning of gaze and actions in the egocentric video. In Chapter 5, I discuss how to utilize motor attention — the intentional body movement, to address the challenging problem of forecasting human-object interactions from egocentric perspective. And in Chapter 6, I further introduce a novel generative adversarial network to address the task of future hand segmentation from egocentric videos.

2.2 Action Recognition

There is a large body of literature on action recognition. Recent review papers [38, 39] provide comprehensive surveys on this topic. Here, we mainly discuss the literature on recognizing trimmed video clips, which is related, but distinctly different from problems of temporal action localization (e.g., ActivityNet [40]), or spatio-temporally action detection (e.g., AVA [41]). Recent efforts focus on developing deep models for action recognition. Simonyan and Zisserman [42] proposed two-stream networks that learn to recognize an action from both optical flow and RGB frames. 3D convolutional networks were further proposed [43, 44, 45] to capture spatio-temporal features beyond a single frame. However, their performance using video frames alone falls far behind their two-stream versions [45].

There are several recent attempts in recovering the motion cues encoded in videos from RGB frames alone. Bilen et al. [46] proposed a dynamic image network that makes use of the parameters of a ranking machine that captures the temporal evolution of the video frames. Ng et al. [47] proposed to jointly predict action labels and flow maps from video frames using multi-task learning. This idea is extended by Fan et al. [48], where they fold the TV-L1 flow estimation [49] into their TVNet. Without using flow, Tran et al. [50] demonstrated that factorized 3D convolutions (2D spatial convolution and 1D temporal convolution) can facilitate the learning of spatio-temporal features. A similar finding was also presented by Xie et al. [51]. Tran et al. [52] further proposed a dense 3D group convolutional network for video classification. Feichtenhofer et al. [53] proposed to combine a low frame rate, high-resolution slow stream with a high frame rate, low-resolution fast stream for action recognition. Yang et al. [54] proposed a temporal pyramid network for learning video representation at different temporal resolutions.

My thesis work shares the similar motivation of learning motion-aware video representation, yet takes a vastly different route. We propose to distill the predicted attention from a flow network to an RGB network. A similar idea was also explored in two previous works [55, 56]. These methods assume that the reference flow model has better recognition performance, and seek to regularize the learning of an RGB stream by asking the RGB network to mimic the features from the flow network. Consequently, those methods might fail where the flow stream under-performs the RGB stream, e.g., on egocentric action recognition datasets [57]. In Chapter 3, I present a novel model that drops the assumption of a strong reference flow network, and thus is more robust for learning motion-sensitive video representation.

2.3 Visual Attention

Attention mechanism has been widely used for visual recognition. Here, I mainly discuss selective visual attention that highlights discriminative regions. This is very different from

the recent efforts on self-attention, i.e., self-similarity [1, 58, 59] adopted in the transformer architecture. Recently, selective attention has been explored in deep models for object recognition [60] and image captioning [61]. Attention enables these models to "fixate" on image regions, where the decision is made based on a sequence of fixations. Several attention mechanisms are proposed for deep models. For example, Sharma et al. [62] integrated soft attention in LSTMs for action recognition. Li et al. [63] further extended the attention-based image captioning model [61] into videos. Specifically, they combined LSTMs with motion-based attention to infer the location of the actions. Girdhar et al. [64] modeled top-down and bottom-up attention using bilinear pooling. Wang et al. [65] proposed a residual architecture for soft attentions.

There have been a few works that showed the benefits of gaze-indexed or hand-indexed visual features for egocentric action recognition [4, 34, 19]. However, these methods require *side information* in addition to the input image at testing time, e.g., hand masks [34, 4], object information [6], or human gaze [19]. More recently, Sudhakaran et al. [66, 16] presented LSTM models with soft attention for FPV action recognition. Furnari and Farinella [26] proposed to combine two LSTMs with attention for FPV action anticipation. However, these methods did not explicitly model the unique egocentric cues. Built on those previous works, my thesis work provide a systematic study of the utility of attention model in action recognition, and further shows that probabilistic attention serves as an ideal tool to distill motion cues from a flow network to an RGB network. I also demonstrate how different forms of egocentric attentional representations can facilitate the understanding of human actions.

2.4 Human-Scene Interaction

Human-Scene constraints have been proven to be effective in reasoning about human body pose [67, 68, 69]. In this section, we focus on the most relevant prior works on visual affordance. Affordance can be helpful for scene understanding [70, 71, 72], human-object in-

teraction recognition [73], and action analysis [74, 75]. Several recent works have focused on estimating visual affordances that are grounded on human-object interaction. Chen et al. [76] proposed to estimate likely object interaction regions by learning the connection between subject and object. Fang et al. [77] proposed to estimate interaction regions by learning from demonstration videos. However, none of these previous works considered future anticipation. Grabner et al. [70] proposed to predict object functionality by hallucinating an actor interacting with the scene. A similar idea was also explored in [78, 79]. Savva et al. [80] predicted action heat maps that highlight the likelihood of an action in the scene by partitioning scanned 3D scenes into disjoint sets of segments and learning a segment dictionary. Gupta et al. [81] presented a human-centric scene representation for predicting the afforded human body poses. Delaitre et al. [71, 82] introduced a statistical descriptor of person-object interactions for object recognition and human body pose prediction. Fang et al. [77] proposed to learn object affordances from demonstrative videos. Nagarajan et al. [83] proposed to use backward attention to approximate the interaction hotspots of future action. Our design of environment prior introduced in Chapter 4 is built on these previous efforts on human-scen interaction.

Far fewer works have considered the environmental factors and the spatial grounding of egocentric activity. Guan et al. [84] and Rhinehart et al. [85] jointly considered trajectory forecasting and egocentric activity anticipation with online inverse reinforcement learning. The most relevant works to my thesis work are recent efforts on learning affordances for egocentric action understanding [86, 74]. Nagarajan et al. [86] introduced a topological map environment representation for long-term activity forecasting and affordance prediction. Rhinehart et al. [74] considered a novel problem of learning "Action Maps" from egocentric videos. However, methods that use ground plane representations of the environment [74] or environmental functionality as the context [86] may lack the specificity provided by 3D proximity. In contrast to these prior efforts, the focus of Chapter 4 of my thesis work is to exploit the geometric and semantic information of the 3D environment

map to address our novel task of joint egocentric action recognition and 3D localization.

2.5 Human Body Motion Forecasting

Here we mainly discuss previous investigations on forecasting the human body motions using generative models. Fragkiadaki et al. [87] proposed to use a recurrent network for predicting and generating the human body poses and dynamics from videos. A similar idea was also explored in [88]. Walker et al. [89] utilized Variational Autoencoders (VAE) for predicting the dense trajectories of video pixels. They further leveraged human body poses as an intermediate feature for generating future video frames with a Generative Adversarial Network (GAN) [90]. Gupta et al. [91] explored a GAN-based model for forecasting human trajectories. Zhang et al. [92, 93] developed a Conditional Variational Autoencoder to generate human body meshes and motions in 3D scenes. Despite the success in forecasting body motion, the use of GANs was largely understudied in egocentric vision. Zhang et al. [33] used a GAN to generate future video frames and further predict future gaze fixation. Though GAN has the capability of addressing the uncertainty of data distribution, using GANs to directly forecast pixels in video [94] remains a challenge, especially when there exists drastic background motion in the egocentric videos [33]. In the final piece of my thesis work, we propose to use the adversarial training mechanism to model the underlying distribution of possible future head motion, which captures the drastic change of scene context for future hand segmentation in the egocentric video.

CHAPTER 3

ATTENTION AND ACTIONS IN EGOCENTRIC VIDEO

3.1 Introduction

Attention mechanism, imitating the human visual system that highlights important foreground regions, has been developed for action recognition models. Focusing on critical regions of video can eliminate the potential distraction of irrelevant background pixels and allows the model to focus on the key elements of the action. Egocentric video provides a new perspective for understanding human visual attention, as gaze, action and video are aligned in the same egocentric coordinate system, and therefore gaze fixation is naturally embodied in the camera wearer's actions. This observation motivates us to propose a joint model that learns the attention map from gaze measurements during the production of actions, and thereby guides action recognition.

A major challenge for the joint modeling task is the uncertainty in gaze measurements. Around 25% [95] of our gaze within daily actions are saccades—rapid gaze jumps during which our vision system receives no inputs [96]. To address this challenge, we characterize gaze as a *latent* distribution of attention in the context of an action, represented as an attention map in egocentric coordinates. Specifically, we model the latent distribution of gaze as stochastic units in a deep network. This representation allows us to sample attention maps. These maps are further used to selectively aggregate visual features in space and time for action recognition. Our model thus both encodes the uncertainty in gaze measurement, and models visual attention in the context of actions.

In addition to selecting important feature representations, the attention module may also provide an attractive vehicle to efficiently and effectively learn representations of both appearance and motion information. Consider the example in Figure 3.1 (a), where we

Figure 3.1: **RGB and flow networks attend to different aspects of an action, yet both are essential for recognition**. (a) Attention maps from RGB and flow streams of I3D [45] by Grad-Cam [97]. (b) Attention maps from our attention distillation model. Our model jointly infers appearance and motion attention from only RGB frames, improves the performance of the RGB stream, and is significantly more efficient than two-stream models.

adopt Grad-Cam [97] to compute attention maps from the appearance (RGB) and motion (flow) streams of a trained I3D model. The attention maps from the RGB and flow streams are qualitatively different. The appearance modality learns to focus on the actor's body and part of the active object (flute), while the motion modality highlights the moving fingers. Intuitively, both appearance attention for highlighting object properties and motion attention for understanding actor moving patterns are needed to recognize the actions.

With optical flow as an additional modality, the two-stream network learns motion-based representation that attends to important moving regions, however, it is computationally expensive. A two-stream model can be 100 times slower than the single RGB stream version [55], largely due to the costly computation of optical flow. While 3D convolutional networks have the capacity to capture the temporal dynamics of video frames [98], their performance still lags behind the two-stream models. Moreover, 3D convolutional networks fail to learn the same motion-based attentional representation as the flow stream model, demonstrating that the flow field provides an indispensable cue. In this context, we address the following research questions: *Does a deep model need explicit flow inputs to learn motion-based attentional representations? How can we bridge the gap between an*

RGB stream network and its two-stream version without incurring the extra computational cost? Several previous works have addressed the challenge of learning a video representation that encodes motion information using a single RGB stream [50, 56, 55, 99, 100]. Our work shares the same motivation, but pursues a very different approach that makes motion attention as the first-class player.

Specifically, we present a novel video representation learning method called *attention distillation*. Our method makes use of an explicit probabilistic attention model, and leverages motion information available at training time to predict the motion-sensitive attention features from a single RGB stream. In addition to their utility in visualizing and understanding learned feature representations, we argue that attention models provide an attractive vehicle for mapping between sensing modalities in a task-sensitive way. Once learned, our model requires only RGB frames as inference inputs, and jointly predicts appearance and motion attention maps for action recognition. We conduct extensive experiments and demonstrate that our attention distillation enables more accurate action recognition across several video datasets, while remaining very efficient.

To summarize, this chapter has the following contributions:

- I propose a novel joint model for learning gaze and actions in egocentric video, and show the proposed model can significantly benefit action recognition performance.
- I present a systematic study of different choices of attention modules for action recognition and show that a general form of probabilistic attention module can better facilitate video representation learning.
- I introduce a novel method for learning motion-based attentional representations from RGB frames. Through extensive experiments, I demonstrate that our approach distills motion knowledge into an RGB network by mimicking the attention map of a reference flow network.

This work was a collaboration with Prof. Yin Li, Dr. Yun Zhang. The work was published

in ECCV 2018 as a poster paper [101] and BMVC 2020 [102] as an oral paper.

In the following sections, I first introduce a novel joint model for learning gaze and actions in egocentric video. I Finally, I present detailed experimental results to show the benefits of our proposed models and then conclude this chapter.

3.2 Joint Modeling of Gaze and Actions

In this section, we present the details of our joint model of gaze and action in egocentric video. Our inspiration comes from the observation that gaze can be characterized by a *latent* distribution of attention in the context of an action, represented as an attention map in egocentric coordinates. This map identifies image regions that are salient to the current action, such as hands, objects, and surfaces. Building on this intuition, we develop a deep latent variable model for joint action recognition and gaze estimation.

3.2.1 Model Overview

We denote an input first person video as $x = (x^1, ..., x^t)$ with its frames x^t indexed by time t. Our goal is to predict the action category y for x. We assume egocentric gaze measurements $g = (g^1, ..., g^t)$ are available during training yet need to be inferred during testing. g^t are measured as a single 2D gaze point at time t defined on the image plane of x^t. We reparameterize g^t as a 2D saliency map $g^t(m, n)$, where the value of the gaze position is set to one and all others are zero.

Figure 3.2 presents an overview of our model. Our model takes a video x as input and outputs the distribution of gaze q as an intermediate result. We then sample the gaze map g from this predicted distribution. g encodes location information for actions and thus can be viewed as a source of action proposals. Finally, we use the attention map to select features from the network hierarchy for recognition.

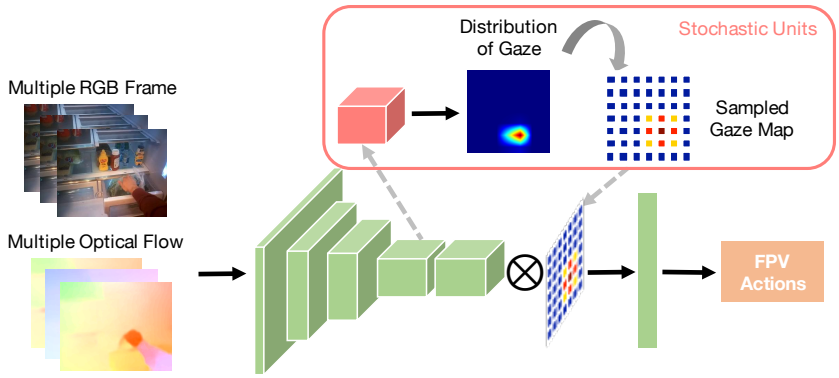

Figure 3.2: **Proposed joint model of gaze and action**. Our network takes multiple RGB and flow frames as inputs, and outputs a set of parameters defining a distribution of gaze in the middle layers. We then sample a gaze map from this distribution. This map is used to selectively pool visual features at higher layers of the network for action recognition. During training, our model receives action labels and noisy gaze measurement.

3.2.2 Modeling Gaze with Stochastic Units

Our main idea is to model $g(m, n)$ as a probabilistic variable to account for its uncertainty. More precisely, we model the conditional probability of $p(y|x)$ by

$$p(y|x) = \int_g p(y|g, x)p(g|x)dg. \tag{3.1}$$

Intuitively, $p(g|x)$ estimates gaze g given the input video x. $p(y|g, x)$ further uses the predicted gaze g to select visual features from input video x to predict the action y. Inspired by [103, 104], we approximate the intractable posterior $p(g|x)$ with a carefully designed $q_\pi(g|x)$. Specifically, we define $q(m, n)$ on a 2D image plane of the same size $M \times N$ as x. q is parameterized by $\pi_{m,n}$, where

$$q(m, n) = q(g_{m,n} = 1|x) = \frac{\pi_{m,n}}{\sum_{m,n} \pi_{m,n}}. \tag{3.2}$$

$\pi = q_\psi(X)$ is the output from a deep neural network q_ψ. $q(g|x)$ thus models the probabilistic distribution of egocentric gaze. Thus, our deep network creates a 2D map of $\pi_{m,n}$. π defines an approximation q_π to the distribution of the latent attention map. We then sample

the gaze map $\tilde{g}$ from q_π for recognition, and use a sampled gaze map $\tilde{g}$ to selectively aggregate visual features $\phi(x)$ defined by network ϕ. In our model, this is simply a weighted average pooling, where the weights are defined by the gaze map $\tilde{g}$. We then send pooled features to the recognition network f. Now we have

$$p(y|g,x) = f(\Sigma_{m,n}\tilde{g}_{m,n}\phi(x)_{m,n}) = softmax\left(W_f^T(\Sigma_{m,n}\tilde{g}_{m,n}\phi(x)_{m,n})\right). \tag{3.3}$$

The sum operation is equivalent to spatially re-weighting individual feature channels. recognition network f has the form of a linear classifier, followed by a softmax function.

3.2.3 Training and Inference

Loss Function. Given our noise model of gaze $p(g|x)$, we now minimize our loss function as the negative of the empirical lower bound, given by

$$-\sum_g \log p(y|g,x) + KL[q(g|x)||p(g|x)]. \tag{3.4}$$

Our objective function thus has two terms: (a) the negative log likelihood term as the cross entropy loss between the predicted and the ground-truth action labels using the sampled gaze maps; and (b) the KL divergence between the predicted distribution $q(g|x)$ and the gaze distribution $p(g|x)$. Note that when the prior distribution of gaze is not available as supervisory signal, we replace $p(g|x)$ with a uniform distribution U.

Reparameterization. Our model is fully differentiable except for the sampling of $\tilde{g}$. To allow end-to-end back propagation, we re-parameterize the discrete distribution $q(m,n)$ using the Gumbel-Softmax approach as in [105, 106]. Specifically, instead of sampling from $q(m,n)$ directly, we sample the gaze map $\tilde{g}$ via

$$\tilde{g}_{m,n} \sim \frac{\exp((\log \pi_{m,n} + G_{m,n})/\tau)}{\sum_{m,n}\exp((\log \pi_{m,n} + G_{m,n})/\tau)}, \tag{3.5}$$

where τ is the temperature that controls the "sharpness" of the distribution. G follows the Gumbel distribution $G = -\log(-\log U)$, where U is the uniform distribution on $[0, 1)$.

3.2.4 Approximate Inference

During testing, we feed an input video x forward through the network to estimate the gaze distribution $q(g|x)$. Ideally, we should sample multiple gaze maps $\tilde{g}$ from q, pass them into our recognition network $f(g, x)$, and average all predictions. This is, however, prohibitively expensive. Since $f(g, x)$ is nonlinear and g has hundreds of dimensions, we will need many samples $\tilde{g}$ to approximate the expectation $E_g[f(g, x)]$, where each sample requires us to recompute $f(\tilde{g}, x)$. We take a shortcut by feeding q_π into f to avoid the sampling. We note that q_π is the expectation of $\tilde{g}$, and thus our approximation is $E_g[f(g, x)] \approx f(E[g], x)$.

This shortcut does provide a good approximation. Recall that our recognition network f is a softmax linear classifier. Thus, f is convex (even with the weight decay on W_f). By Jensen's Inequality, we have $E_g[f(g, x)] \geq f(E[g], x)$. Thus, our approximation $f(E[g], x)$ is indeed a lower bound for the sample averaged estimate of $E_g[f(g, x)]$. Using this deterministic approximation during testing also eliminates the randomness in the results due to sampling. We have empirically verified the effectiveness of our approximation.

3.3 Attention Distillation for Learning Video Representations

The two-stream architecture [42, 45] has proven to be an effective framework to exploit appearance and motion cues for action recognition. However, a two-stream model can be 100 times slower than its single RGB stream version [55]. Several previous works address the challenge of learning video features that encode motion information using a single RGB stream [50, 56, 55, 99, 100]. In addition, learning motion-sensitive representation from egocentric videos is even more challenging due to the drastic head motion. In this section, going beyond modeling the connection between gaze and action, I demonstrate a more general approach that utilizes the attention mechanism for transferring the learned motion

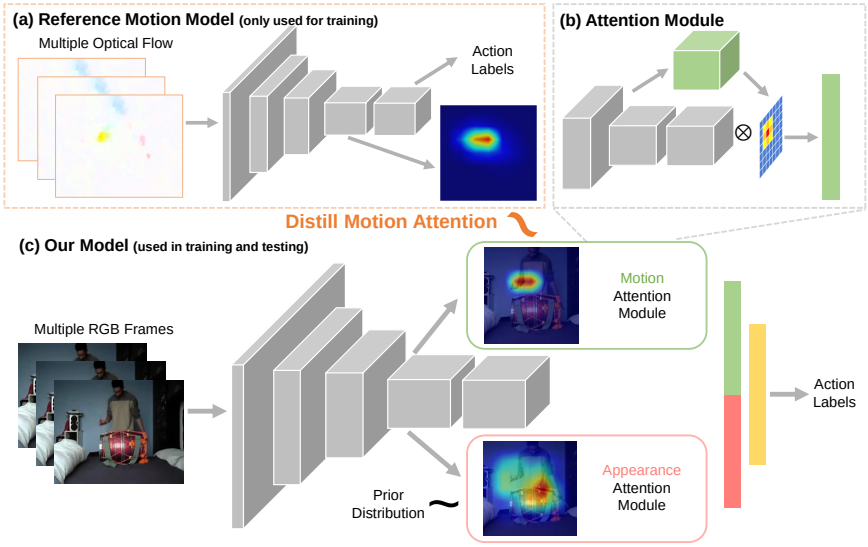

Figure 3.3: **Proposed attention distillation method**. Our model (c) takes multiple RGB frames as inputs and adopts a 3D convolutional network as the backbone. It outputs two attention maps using the attention module (b), based on which the action labels are predicted. The motion map is learned by mimicking the attention from a reference flow network (a). The appearance map is learned to highlight discriminative regions for recognition. These two maps are used to create spatio-temporal feature representations for action recognition.

knowledge from a flow network to an RGB network for both third-person and egocentric action recognition.

3.3.1 Model Overview and Key Components

We denote the input video as $x = \{x^1, x^2, ..., x^T\}$, where x^t is a frame of resolution $H \times W$ with t as the frame number. Given x, our goal is to predict a video-level action label y. We leverage the intermediate output of a 3D convolutional network ϕ to represent x. Figure 3.3 presents an overview of our method. Our model takes multiple video frames x as inputs and learns to predict two attention maps based on $\phi(x)$: $\mathcal{A}^M$ for motion attention and $\mathcal{A}^A$ for appearance attention. Based on these two maps, the model further aggregates visual features that will be passed into the final recognition sub-network. During training, we match $\mathcal{A}^M$ to the attention map $\tilde{\mathcal{A}}^M$ from the reference flow network. For testing, only the input video is required for recognition. In following sections, I detail detail the design of our key components.

3.3.2 Attention Generation

We explore two different approaches for generating an attention map from the features $\phi(x)$, including soft attention [65] and its probabilistic version [101].

Soft Attention. Attention maps can be created by a linear function of $w_a \in R^{C_\phi}$ over the feature map $\phi(x)$,

$$F_{\mathcal{A}}(\phi(x)) = softmax(w_a * \phi(x)), \tag{3.6}$$

where $*$ is the 1x1 convolution on 3D feature grids. Softmax is applied on every time slice to normalize each 2D map.

Probabilistic Soft Attention. An alternative approach is to further model the distribution of linear mapping outputs as discussed in [101], namely

$$\mathcal{A} \sim p(\mathcal{A}) = softmax(w_a * \phi(x)) \tag{3.7}$$

where we model the distribution of A. During training, an attention map can be sampled from $p(\mathcal{A})$ using Gumbel Softmax trick [105, 106]. We follow [101] to regularize the learning by adding additional loss term of

$$\mathcal{L}^R = \sum_t KL\left[\mathcal{A}(t)||U\right], \tag{3.8}$$

where $KL[\cdot]$ is the Kullback-Leibler divergence and U is the 2D uniform distribution ($H_\phi \times W_\phi$). This term matches each time slice of the attention map to the prior distribution. It is derived from variational learning and accounts for (1) the prior of attention maps and (2) additional regularization by spatial dropout [101]. During testing, we directly plug in $p(\mathcal{A})$ (the expected value of $\mathcal{A}$) for approximate inference.

Note that for both approaches, we restrict $F_{\mathcal{A}}$ to a linear mapping without a bias term. In practice, this linear mapping avoids the trivial solution of generating a uniform attention map by setting w to all zeros. This all-zero solution almost never arises during training when using a proper initialization of w.

3.3.3 Attention Guided Recognition

Our recognition module makes use of an attention map $\mathcal{A}$ to select features from $\phi(x)$. Again, we consider two different models for the attention guided recognition.

Attention Pooling. Inspired by [65, 107], we design the function $F_{\mathcal{R}}$ as

$$\tilde{y} = F_{\mathcal{R}}(\phi(x), \mathcal{A}) = softmax\left(W_r^T(\mathcal{A} \otimes \phi(x))\right) \tag{3.9}$$

where $\otimes$ denotes the tilted multiplication $\mathcal{A} \otimes \phi(x) = \sum_{t,h,w} \mathcal{A}(t,h,w)\phi(x)_{t,h,w,c}$. This operation is equivalent to weighted average pooling with the weights shared across all channels.

Residual Connection. Using the attention map to re-weight features helps to filter out background noise, yet may also increase the potential risk of missing important foreground features. This drawback was discussed in [65]. We follow their solution of using a residual connection to the attention map, given by

$$\tilde{y} = F_{\mathcal{R}}(\phi(x), \mathcal{A}) = softmax\left(W_r^T((\mathcal{A} + I) \otimes \phi(x))\right), \tag{3.10}$$

where I is a 3D tensor of all ones. Intuitively, this operation further adds average pooled features to the representation before the linear classifier. By adding the residual term, the features learned by the network are preserved.

3.3.4 Attention Distillation

The key to our approach lies in the use of attention distillation during training. Specifically, we assume that a reference flow network is given as the teacher network. The teacher model also uses an attention mechanism for recognition. Moreover, its motion attention map $\tilde{\mathcal{A}}^M$ is used as additional supervisory signal for training our RGB network. This RGB network is thus the student model that mimics the motion attention map. With probabilistic attention modeling, the imitation of the attention maps is enforced by using the loss

$$\mathcal{L}^{\mathcal{A}} = \sum_t KL\left[\mathcal{A}^M(t) || \tilde{\mathcal{A}}^M(t)\right]. \tag{3.11}$$

This loss minimizes the distance between the attention maps at every time step t. In our implementation, our teacher flow network is trained with the same attention mechanism. Once trained, the weights of the teacher model remain fixed during the learning of the student model. At testing time, only the student model (RGB network) is used for inference.

3.3.5 Our Full Model

Putting everything together, we summarize our full model with probabilistic soft attention and attention distillation. Specifically, our model estimates the two probabilistic attention maps $\mathcal{A}^M \sim F^M_{\mathcal{A}}(\phi(x))$ (motion) and $\mathcal{A}^A \sim F^A_{\mathcal{A}}(\phi(x))$ (appearance). These maps are further used to predict the action labels. This is given by

$$\tilde{y} = F^M_{\mathcal{R}}(\phi(x), \mathcal{A}^M) + F^A_{\mathcal{R}}(\phi(x), \mathcal{A}^A) \tag{3.12}$$

where each $F_{\mathcal{R}}$ follows Equation 3.9. We use equal weighting for $F^M_{\mathcal{R}}$ and $F^A_{\mathcal{R}}$. We found that tuning the weights has negligible effect on the performance in practice.

Loss Function. Our training loss is defined as

$$\mathcal{L} = CE(\tilde{y}, y) + \lambda_1 \sum_t KL\left[\mathcal{A}^M(t) || \tilde{\mathcal{A}}^M(t)\right] + \lambda_2 \sum_t KL\left[\mathcal{A}^A(t) || U\right], \tag{3.13}$$

where CE is the cross entropy loss between the predicted labels $\tilde{y}$ and the ground-truth y. The first KL term (from Equation 3.11) enforces that the motion attention $\mathcal{A}^M$ should mimic the attention map $\tilde{\mathcal{A}}^M$ from the reference flow network. Finally, the second KL term (from Equation 3.8) regularizes the learning of the appearance attention.

3.4 Experiment and Results

3.4.1 Gaze and Actions

Dataset. We use the Extended GTEA Gaze+ egocentric video dataset [57] to evaluate our method. The dataset has both action annotation and gaze tracking data at every frame.

Evaluation Metric. We use standard metrics for both gaze and actions.

• *Gaze*: We consider gaze estimation as binary classification. We evaluate all fixation points

Table 3.1: **Ablation study on backbone networks and probabilistic modeling**. We show F1 scores for gaze estimation and mean class accuracy for action recognition.

(a) **Backbone Network**: We compare RGB, Flow, late fusion and joint training of I3D for action recognition. Joint training works the best.

Networks	Action Acc (Clip)	Action Acc (Video)
I3D RGB	43.69	47.26
I3D Flow	32.08	38.31
I3D Fusion	N/A	48.84
I3D Joint	**46.42**	**49.79**

(b) **Probabilistic Modeling**: We compare our model to its deterministic version (Gaze MLE). We also study the effect of Dropout.

Methods	Gaze F1	Action Acc
I3D Joint	N/A	49.79
Gaze MLE	24.68	51.12
Soft-Atten	10.27	50.30
Ours (Prob.)	**32.97**	**53.30**
Ours w. Dropout	32.66	52.12

and ignore untracked gaze or saccade in action clips. We report the Precision and Recall values and their corresponding F1 score.

• *Action*: We treat action recognition as multi-class classification. We report mean class accuracy at the clip level (24 frames) and at the video level.

Ablation Study. We start with a comprehensive study of our model on the EGTEA Gaze dataset. Our model consists of (1) the backbone network for feature presentation; (2) the probabilistic modeling; and (3) the attention guided action recognition. We separate out these components and test them independently.

Backbone Network: RGB vs. Flow. We evaluate different network architectures on EGTEA dataset for FPV action recognition. Our goal is to understand which network performs the best in the egocentric setting. Concretely, we tested RGB and flow streams of I3D [45], the late fusion of two streams, and the joint training of two streams [108]. The results are summarized in Table 3.1a. Overall, EGTEA dataset is very challenging, even the strongest model has an accuracy below 50%. To help calibrate the performance, we note that the same I3D model achieved 36% on Charades [109, 110], 74% on Kinetics and 99% on UCF [45].

Unlike Kinetics or UCF, where flow stream performs comparably to RGB stream, the performance of I3D flow stream on EGTEA is significantly lower than its RGB counterpart. This is probably because of the frequent motion of the camera in FPV. It is thus more

difficult to capture motion cues. Finally, the joint training of RGB and flow streams performs the best in the experiment. Thus, we choose this network as our backbone for the rest of our experiments.

Modeling: Probabilistic vs. Deterministic. We then test the probabilistic modeling part of our method. We focus on the key question: "What is the benefit of probabilistic modeling of gaze?" To this end, we present a deterministic version of our model that uses maximum likelihood estimation for gaze. We denote this model as *Gaze MLE*. Instead of sampling, this model learns to directly output a gaze map, and apply the map for recognition. During training, the gaze map is supervised by human gaze using a pixel-wise sigmoid cross entropy loss. We keep the model architecture and the training procedure the same as our model. And we disable the loss for gaze when fixation is not available.

We compare our model with Gaze MLE for gaze and actions, and present the results in Table 3.1b. Our probabilistic model outperforms its deterministic version by 2.2% for action recognition and 8.3% for gaze estimation. We attribute this significant gain to the modeling. If the supervisory signal is highly noisy, allowing the network to adapt the stochasticity will facilitate the learning.

Regularization: Sampling vs. Dropout. To further test our probabilistic component, we compare our sampling of gaze map to the dropout of features. As we discussed in Sec 3.4, the sampling procedure in our model can be viewed as a way of "throwing away" features. Thus, we experiment with enabling Dropout directly after the attention pooled feature map in the model. Specifically, we compare two models with the same architecture, yet one trained with Dropout and one without. The results are in Table 3.1b. When Dropout is disabled, the network performs slightly better for action recognition ($+1.2\%$) and gaze estimation ($+0.3\%$). In contrast, removing Dropout from the backbone I3D will slightly decrease the accuracy [111]. We postulated that with regularization from our sampling, further dropping out the features will hurt the performance.

Attention for Action Recognition. Finally, we compare our method to a soft attention

Table 3.2: **Action Recognition and Gaze Estimation**. For action recognition, we report mean class accuracy at both clip and video level. For gaze estimation, we show F1 scores and their corresponding precision and recall scores.

(a) **Action Recognition Results**: Our method outperforms previous methods by at least 3.5%. †: methods use human gaze during testing.

Methods	Clip Acc	Video Acc
EgoIDT+Gaze† [4]	N/A	46.50
I3D+Gaze†	46.77	51.21
EgoConv+I3D [34]	N/A	48.93
Gaze MLE	47.41	51.12
Our Model	**47.71**	**53.30**

(b) **Gaze Estimation Results**: Our model is comparable to the state-of-the-art methods. †: methods jointly model gaze and actions.

Methods	F1	Prec	Recall
EgoGaze [31]	16.63	16.63	16.63*
Simple Gaze	30.10	25.14	37.48
Deep Gaze [33]	**33.51**	**28.04**	41.62
Gaze MLE†	24.68	18.55	36.86
Our Model†	32.97	27.01	**42.31**

model (*Soft-Atten* in Table Table 3.1b) using the same backbone networks. Similar to our model, this method fuses the two streams at the end of the 4th and 5th conv blocks. Soft attention map is produced by 1x1 convolution with Sigmoid activations from the fused features at the 4th conv block. This map is further used to pool the fused features (by weighted averaging) at the 5th conv block for recognition. Thus, this soft attention map receives no supervision of gaze. A similar soft attention mechanism was used in a concurrent work [112].

For action recognition, *Soft-Atten* is worse than gaze supervised models by 0.8-3%, yet outperforms the base I3D model by 0.5%. These results suggest that (1) soft attention helps to improve action recognition even without explicit supervision of gaze; and (2) adding human gaze as supervision provides a significant performance gain. For gaze estimation, *Soft-Atten* is worse (-14%) than any gaze supervised models, as it does not receive supervision of gaze.

Egocentric Action Recognition. We now describe our experiments on egocentric action recognition. Specifically, we consider the following baselines:

• *EgoIDT+Gaze* [4] combines egocentric features with dense trajectory descriptors [113].

• *I3D+Gaze* is inspired by [4, 5], where the ground truth human gaze is used to pool the network features for action recognition.

• *EgoConv+I3D [34]* uses hand mask and head rotation as additional streams of egocentric cues for FPV action recognition.

• *Gaze MLE* is a deterministic version of our model that directly applies the gaze map for recognition without sampling.

Our results for action recognition are shown in Table 3.2a. Our full model reaches the accuracy of **53.30%** and outperforms all baseline methods by a significant margin, including those that use human gaze during testing. Notably, even using human gaze at test time, I3D+Gaze is only slightly better than Gaze MLE (+0.1%). We argue that these results provide a strong evidence to our modeling of uncertainty in gaze measurements. A model must learn to account for this uncertainty to avoid misleading gaze points, which will distract the model to action irrelevant regions.

Egocentric Gaze Estimation. We now present our results for egocentric gaze estimation. We compare our model to the following baseline methods.

• *EgoGaze* [31] makes use of hand crafted egocentric features, such as head motion and hand position, to regress gaze points.

• *Simple Gaze* is a deep model inspired by our previous work [5].

• *Deep Gaze* [33] is the FPV gaze prediction module from [33]

• *Gaze MLE* is the deterministic version of our joint model.

Our gaze estimation results are shown in Table 3.2b. Again, deep models outperform hand crafted features by a large margin. We also observe that models with KL loss are consistently better than those that use cross entropy loss. Moreover, the joint models slightly decrease the gaze estimation performance when compared to gaze-only models.

Visualization of Gaze and Action. we visualize the outputs of gaze estimation and action labels from our model in Figure 3.4. Our gaze outputs often attend to foreground objects that the person is interacting with.

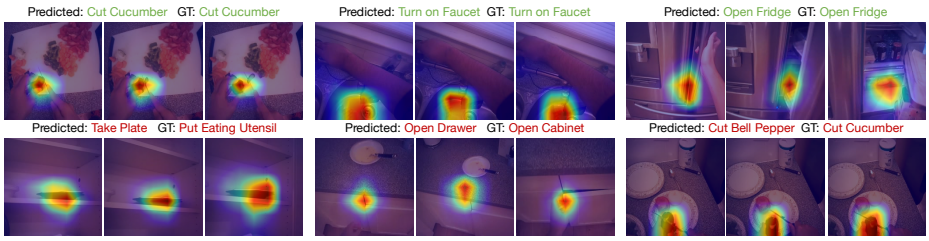

Figure 3.4: **Visualization of our gaze estimation and action recognition**. We plot the output gaze heat map and print the predicted action labels and ground-truth labels. Both successful (first row) and failure cases (second row) are presented.

3.4.2 Attention Distillation

Dataset. We make use of both third-person action recognition datasets UCF101 [114], HMDB51 [115], and egocentric action recognition dataset EGTEA [57] for our experiments. We evaluate mean class accuracy and report the results using the first split of these datasets.

Attention Guided Action Recognition We start from an ablation study of attention-guided action recognition. Specifically, we evaluate different combinations of attention modules and compare their results to those from models without attention. Our experiments show that the proper design of the attention mechanism can consistently improve the performance of action recognition across multiple datasets. We now present our baselines and results.

Baselines. We consider the different combinations of how the model generates attention maps (Soft vs. Probabilistic Attention) and how the attention maps are used for recognition (Attention Pooling vs. Residual Connection). In addition, we also show how the approach to combining motion attention and appearance attention affects the recognition performance. The valid combinations include the following:

- Soft-Atten combines soft attention and attention pooling for recognition similar to [107].
- Soft-Res is the residual attention in [65] that adds residual connection to Soft-Atten.
- Prob-Atten combines probabilistic attention with attention pooling as in [101].

Table 3.3: **Evaluations of attention modules**. We compared 3 different design choices with RGB/flow stream on three datasets. Prob-Atten provides a consistent performance boost on both streams and across datasets.

Method	UCF101	HMDB51	EGTEA
Flow I3D	94.0	73.9	38.3
Flow Soft-Atten	94.7	74.1	39.1
Flow Soft-Res	**95.2**	**74.4**	39.5
Flow Prob-Atten	94.9	74.2	**40.4**
RGB I3D	94.8	70.9	47.3
RGB Soft-Atten	94.7	70.8	48.6
RGB Soft-Res	94.9	70.1	48.6
RGB Prob-Atten	**95.1**	**71.3**	**49.1**

We note that the combination of Prob+Res is invalid, as it violates the probabilistic modeling of attention. In practice, we also found its training to be unstable. Therefore, we report the results of three valid designs for both RGB and flow stream and the vanilla I3D models (our backbone) using the same input sequence length (24 frames) in Table 3.3. Adding attention to the backbone recognition network almost always improves the performance. Importantly, Soft-Res decreases the performance of RGB stream on HMDB51 and Soft-Atten decreases the performance of RGB stream on HMDB51 and UCF101. More interestingly, Prob-Atten is the most robust design choice, despite the lack of human gaze as a supervisory signal as in [101]. Across all of the modalities and datasets, Prob-Atten can consistently improve the recognition accuracy (+$0.3\%/0.4\%/1.8\%$) for the RGB stream and (+$0.9\%/0.5\%/2.1\%$) for the flow stream. The performance boost from the attention module is larger for the flow stream in comparison to the RGB stream. Moreover, attention modules provide more significant boost for egocentric actions (EGTEA). We conjecture that the explicit modeling of attention helps to suppress background objects in first person video.

Impact of Attention Distillation. Table 3.4 compares our results with previous methods on UCF101/HMDB51. We denote our models using Prob-Atten for distillation as *Prob-Distill*. Prob-Distill outperforms all previous state-of-the-art methods of motion representation learning. Specifically, our results are at least 1.2% better than previous state-of-the-

Table 3.4: **Action recognition results on UCF101 and HMDB51 datasets**. We compare the results of our model with previous works. Our model outperforms state-of-the-art methods that use only RGB stream and the same input sequence length by $\sim 1\%$. *For fair comparison, we report results of I3D models that use 24 frames as inputs–the same as our model.

Method	UCF101	HMDB51
Dynamic Image [46]	90.6	61.3
ActionFlowNet [47]	83.9	56.4
TVNet [48]	94.5	71.0
I3D RGB* [45]	94.8	70.9
FeatMatch [116]	94.3	70.7
MARS [55]	94.6	**72.3**
Ours (Prob-Distill)	**95.7**	72.0
Two Stream ResNeXt [55]	95.6	74.0
MARS+Flow ResNeXt [55]	94.9	74.5
Two Stream I3D*	96.7	74.8
Prob-Distill+Flow I3D*	**97.4**	**75.7**

art methods for learning motion-aware video representations from RGB frames, including Dynamic Image [46], ActionFlowNet [47] and TVNet [48]. Our model also outperforms MARS [55], our direct competitor, by 0.9% on UCF101 and performs on-par with MARS on HMDB51 when using a similar sequence length, despite the fact that MARS uses a stronger backbone network. It is worth noting that this performance boost is significant for action recognition. In contrast, with 50 more layers, ResNet101 is only 0.7% better than ResNet50 on HMDB51 [98]. Moreover, Prob-Distill also outperforms another feature distillation method – FeatMatch [116] by a significant margin ($+1.4\%/1.3\%$ on UCF101/HMDB51). These results support our argument that *distilling attention maps is more robust than distilling network features for motion representation learning*. Finally, a late fusion of our model with a reference flow network helps to further boost the performance.

Table 3.5 presents our results on a large scale dataset—20BN-V2. With 1/5 of the temporal receptive field as TRN [117], our model with RGB frames outperforms TRN RGB by $1.1\%/1.5\%$ in top-1/top-5 accuracy. And our method improves the backbone by $2.6\%/3.0\%$ in top-1/top-5 accuracy. Further fusion of our model with a flow network im-

Table 3.5: **Action recognition results on on 20BN-V2 dataset [118]**. Our model achieves the best performance among networks that uses RGB frames. Fusing our model with a flow network also outperforms two stream baseline by a significant margin.

Method	Top-1/Top-5 Acc	Temporal Footprints
TRN RGB [117]	48.8 / 77.6	5 sec
TRN RGB+Flow [117]	**55.5 / 83.1**	5 sec
I3D RGB	47.3 / 76.1	1 sec
I3D Flow	46.7 / 75.9	1 sec
Ours (Prob-Distill)	**49.9 / 79.1**	1 sec
Two Stream I3D	53.7 / 82.5	1 sec
Prob-Distill+Flow I3D	**54.6 / 83.0**	1 sec

proves the results by a large margin (+4.7%), again outperforming the two stream baseline.

Learning from a Weak Flow Network. Crasto et al. [55] pointed out that their model ran into a failure mode when the reference flow network has worse performance than the RGB network. To support our claim that attention distillation can leverage a flow-based teacher network even when the flow network does not provide strong baseline performance, we report the results of our model on the EGTEA Gaze+ dataset. Due to severe ego-motion, flow-based models are less effective than RGB models on this dataset. For instance, I3D Flow is 9% worse than I3D RGB (38.3% vs. 47.3%). Despite a much weaker teacher model, Prob-Distill achieves 49.5%, outperforming the best attention-based I3D models for both RGB (Prob-Atten 49.1%) and Flow (Prob-Atten 40.4%). This indicates that even with a weak teacher model, our proposed method is robust for video representation learning.

Distillation without Forgetting. Feature distillation might "overwrite" the features from RGB stream with the features from flow stream. This is evidenced by the result that fusing MARS with reference flow stream network lags behind the two stream version of the network (MARS + Flow ResNeXt vs. Two Stream ResNeXt in Table 3.4). In contrast, fusing our Prob-distill model with a reference flow model (Prob-Distill + Flow I3D in Table 3.4) further improves the accuracy and outperforms the two-stream I3D model (+0.5% on UCF101, +0.7% on HMDB51 and +0.9% on 20BN-V2). These results indicate that our attention distillation model does not simply copy the feature from the reference flow network, as the distilled RGB model can still preserve meaningful appearance features.

Note that our single-stream (Prob-Distill) results still lag behind the two stream networks when using the same input sequence length (Two Stream I3D*). This gap reveals that our model does not fully capture the concepts of motion that are encoded in the two stream networks. Nonetheless, we believe that our model provides a key step forward for learning motion-aware representations from RGB frames. Note that some most recent works achieved better performance on the benchmark datasets using more advanced network structure [98, 119, 51, 120], additional features [121], or a longer temporal footprint [45, 55]. In this context, our work provides a novel method for learning video representations and a robust strategy for knowledge distillation.

Does the attention help to localize actions? We evaluate our output attention for action localization using THUMOS'13 localization dataset [122]–a subset of UCF101 with bounding box annotations for actions. We present our evaluation metric and discuss our results.

• **Evaluation Metric**. We consider action localization as binary labeling of pixels and report the F1 score from Precision-Recall (PR) curve. Specifically, we first rescale both attention maps and video frames into a fixed resolution (56×56). We then enumerate all thresholds and binarize the attention map. Each threshold defines a point on the PR curve. Given a binary attention map, a positive pixel is considered as a true positive if it is inside the bounding box, or it is within 10-pixel "tolerance zone" of the box. This tolerance is added to compensate for the reduced resolution of the attention map, as in [123]. We report the best F1 score on the curve and its corresponding precision and recall.

• **Results**. We compare attention maps from our model to a set of baseline methods, including a fixed Gaussian distribution (center prior), a latest deep saliency model (DSS [124]), and our Soft-Atten (RGB/Flow). The results are shown in Table 3.6. Our appearance attention beats the baselines of center prior and Soft-Atten (RGB), but is worse than Soft-Atten (flow). Our motion attention achieves the highest score among all methods that only receive action labels as supervision, and only under-performs DSS. We have to emphasis that

Table 3.6: **Results of action localization using attention maps on THUMOS'13 localization test set [122]**. We report the best F1 score and its precision and recall. Our motion attention outperforms all baselines that are trained with only action labels.

Method	Prec	Recall	F1
Gaussian (center prior)	52.6	20.6	29.6
Saliency Map (DSS [124])	**51.2**	47.7	**49.4**
Soft-Atten (RGB)	33.8	40.5	36.9
Soft-Atten (Flow)	39.2	50.0	44.0
Our Appearance	31.5	52.1	39.2
Our Motion	36.3	**62.6**	46.0

directly comparing our results to DSS is unfair. DSS is trained with pixel-level annotations using external data and runs at the original video resolution, while our attention maps are trained using clip-level action labels and down-sampled both spatially (32x) and temporally (8x). These results suggest that our attention maps help to locate the spatial extent of actions.

Does our method learn better motion representation? We further study how the temporal order of the input video frames will affect the recognition performance. We conduct an experiment of classifying reverted videos as in [51, 117]. Specifically, we invert the frame order for all testing videos of UCF101 and HMDB51. We compare their recognition results with those from normal temporal order. If a model truly rely on motion representation for the recognition, this inversion will significantly decrease the recognition performance. We test the vanilla I3D RGB and flow models, as well as our model. And the results are presented in Table Table 3.7. Not surprisingly, I3D flow model has the largest performance drop. In contrast, I3D RGB is barely affected by the reverted arrow of time. Our model has a performance drop that is larger than I3D RGB yet much smaller than I3D flow. This is consistent with our results on action recognition. Our model does not capture the same level of motion information as the flow network.

How is the motion encoded? It is also possible that our model simply copies the motion attention map without encoding motion in the network. To eliminate this hypothesis, we experimented with training an RGB network that directly combines a reference motion at-

Table 3.7: **Inverting the arrow of time for action recognition**. We train the models on normal samples, yet test them on videos with reversed temporal order. A large performance drop indicates that the model has to rely on motion information for the recognition.

Dataset	Method	Mean Class Accuracy		
		Original	Reverted	DeltaΔ
UCF101	I3D RGB	94.8	94.7	0.1
	I3D flow	94.0	89.9	**4.1**
	Ours	95.7	95.1	0.6
HMDB51	I3D RGB	70.9	70.2	0.7
	I3D flow	73.9	66.0	**7.9**
	Ours	72.0	70.6	1.4

tention map and its own appearance attention map for action recognition. The reference motion attention is produced by a flow network during both training and testing. And the rest of this network follows exactly the same architecture as our model. This model has an accuracy of $95.1\%/71.6\%$ on UCF101/HMDB51, under-performing our model by -0.6%/-0.4% on UCF101/HMDB51. These results indicate that the distillation process not only generates motion attention maps, but also learns motion-aware representation.

Visualization of Attention Maps. To better understand our model, we visualize both motion and appearance attention maps from our model. We also compare these maps with attention maps created by our Soft-Atten models from RGB and flow streams in Figure 3.5. Notice that these two attention maps are qualitatively different across all methods. The appearance attention is likely to cover foreground objects or actors, while the motion attention focuses on the moving parts. Moreover, the appearance attention from our model can better localize the foreground regions of actions than those of Soft-Atten from the RGB stream, while the motion attention from our model remains similar to the Soft-Atten from the flow stream. We also find that the attention maps from our model are more "diffused". This is because the regularization by a uniform distribution in Prob-Atten leads to smoother attention maps.

3.5 Conclusion

In this chapter, I first introduce a novel deep model for jointly estimating gaze and recognizing actions in egocentric video. The core innovation is to model the noise in human gaze measurement using stochastic units embedded in a deep neural network. Our model predicts a probabilistic representation of gaze, and uses it to select features for recognition. I then present a novel method that makes use of the attention map as a vehicle to learn a motion-aware video representation. I further present extensive experiments to show the benefits of our approach. Our works thus provide a novel means of developing more expressive deep models using attention mechanisms.

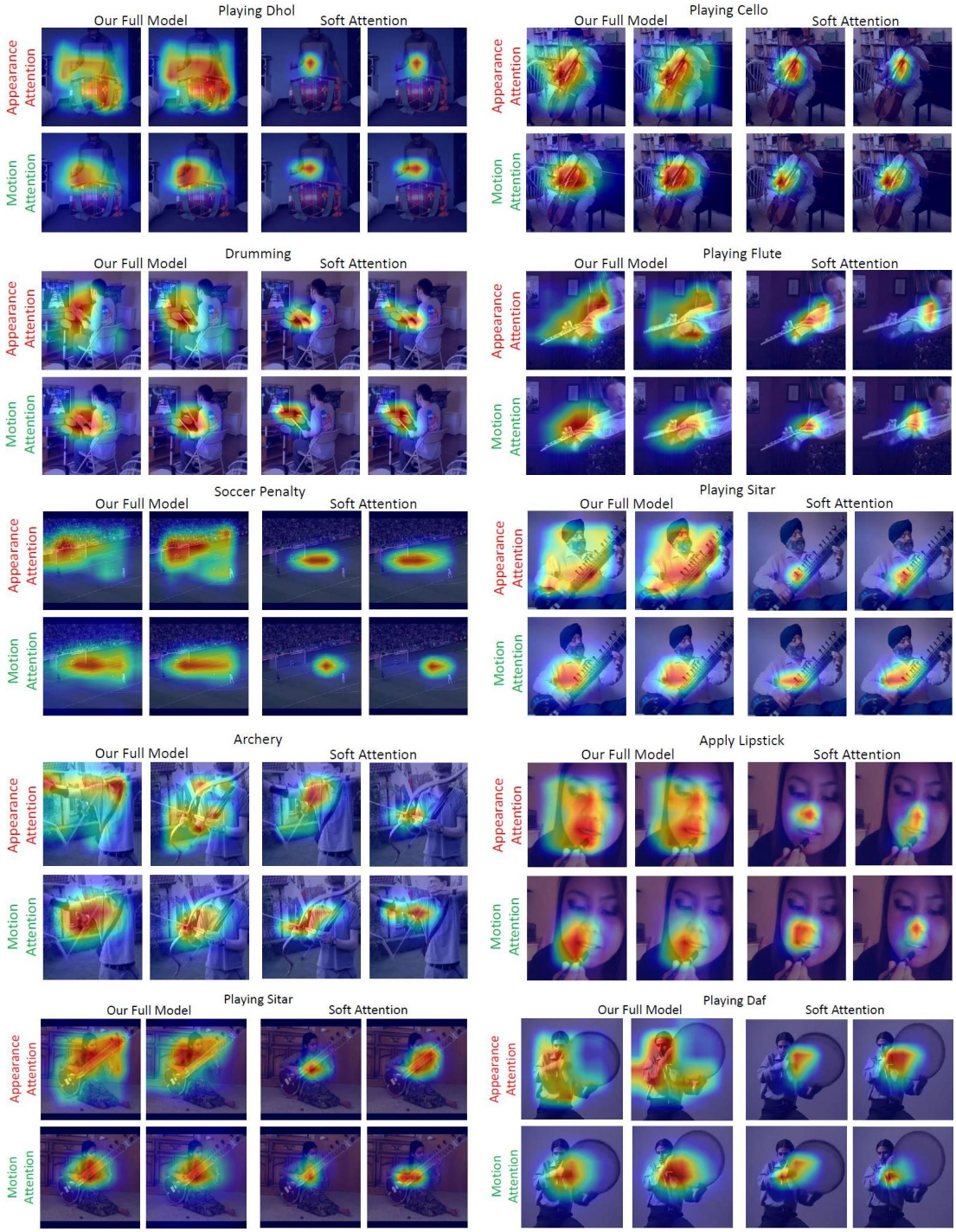

Figure 3.5: **Visualization of attention maps (Ours vs. Soft-Atten using the same I3D backbone)**. For each video clip, we re-interpolate the attention maps and plot them on the first and last frame. Red regions indicate higher value of attention. Our model produces appearance and motion attention maps that are qualitatively different and index key action regions.

CHAPTER 4

EGOCENTRIC ACTIVITY RECOGNITION AND 3D LOCALIZATION

4.1 Introduction

Egocentric videos implicitly and naturally connect the camera wearer's activities to the relevant 3D spatial context, such as the surrounding objects and their 3D layout. While this observation has been true since the beginning of egocentric vision, it is only recently that 3D scene models that can capture this context have become readily available, due to advances in 3D scanners [125] and Augmented Reality (AR) headsets [126]. Figure 4.1 (a) gives an example of a 3D scan of a subject's apartment in which the 3D layout of the furniture and appliances is known *a priori*. How could we design a computational model that leverages such 3D map to reason about the camera wearer's activities and the 3D locations in which they are performed, *e.g.* , drawing a picture while sitting on the sofa?

In this chapter, we introduce the new task of the *joint recognition and 3D localization of egocentric activities given trimmed videos and a coarsely-annotated 3D environment map*. We provide a visual illustration of our problem setting in Fig. Figure 4.1 (b). Two major challenges arise in our task. First, standard architectures for egocentric activity recognition are not designed to incorporate 3D scene context, requiring a new design of action recognition models and associated 3D scene representations. Second, the exact ground truth for the locations of actions in a 3D scene that is the size of an entire apartment is difficult to obtain, due to ambiguities in 2D to 3D registration. As a remedy, we leverage camera registration using structure-from-motion that yields "noisy" locations, which requires the model to address the uncertainty in action locations during training.

To address the challenge of leveraging context in recognition, we develop a Hierarchical Volumetric Representation (HVR) to describe the semantic and geometric information

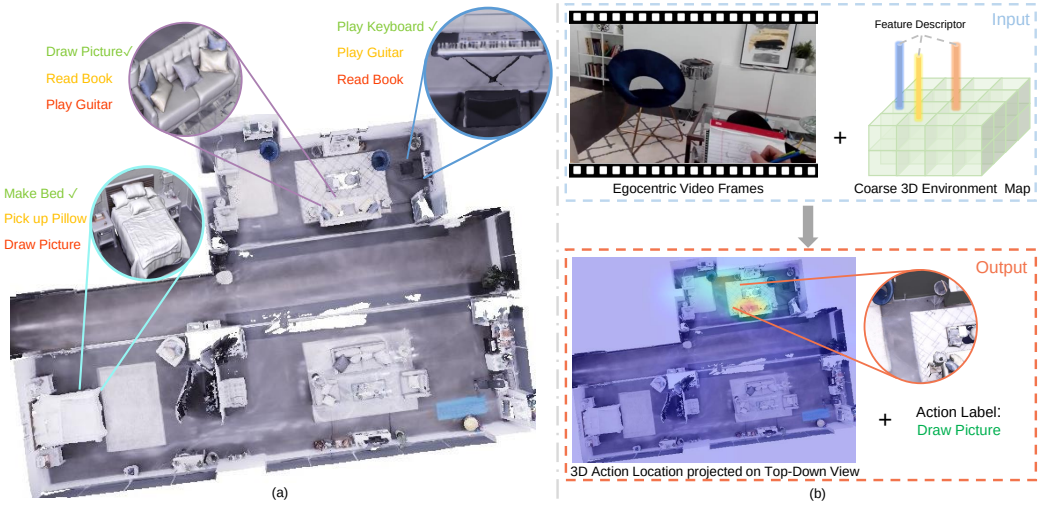

Figure 4.1: (a) **Motivation**:The activities of daily life take place in a 3D environment, and the semantic and spatial properties of the environment are powerful cues for activity recognition. (b) **Our Proposed Task**: Given an input egocentric video sequence and a 3D volumetric representation of the environment (carrying both semantic and geometric information), our goal is to detect and localize activities, by jointly predicting the action label and location on the 3D map where it occurred.

of the 3D environment map (see Fig. Figure 4.2 (a) and Chapter 4.2 for explanation). We further present a novel deep model that takes egocentric videos and our proposed 3D environment HVR as inputs, and outputs the 3D action locations and the activity classes. Our model consists of two branches. The *environment branch* makes use of a 3D convolutional network to extract global environmental features from HVR. Similarly, the *video branch* uses a 3D convolutional network to extract visual features from the input video. The environmental and visual features are further combined to estimate the 3D activity location, supervised by the results of camera registration. Moreover, we tackle the second challenge of noisy localization by using stochastic units to account for uncertainty. The predicted 3D activity location, in the form of a probabilistic distribution, is then used as a 3D attention map to select local environmental features relevant to the action. Finally, these local features are further fused with video features for recognition.

Our method is trained and evaluated on the recent, freely-available Ego4D dataset [7], which contains naturalistic egocentric videos and photo-realistic 3D scene reconstructions

along with 3D static object annotations. We demonstrate strong results on action recognition and 3D action localization. Specifically, our model outperforms a strong baseline of 2D video-based action recognition methods by 4.2% in mean class accuracy, and beats baselines on 3D action localization by 9.3% in F1 score. Furthermore, we demonstrate that our method can generalize to unseen environments not present in the training set yet with known 3D maps and object labels.

To summarize, this chapter has the following contributions:

- I propose a novel task of egocentric action recognition and localization on a 3D environment map.
- I introduce a novel environment representation and a novel computational model to address our proposed task.
- I show that our method achieves consistent performance improvement on both action recognition and 3D localization for both seen and unseen splits.

This work was a collaboration with Dr. Chao Li, Dr. Ling Ma, Dr. Kiran Somasundaram, and Prof. Kristen Grauman. The work was published in ECCV 2022 [127] as a poster paper.

4.2 Method

We denote a trimmed input egocentric video as $x = (x^1, ..., x^t)$ with frames x^t indexed by time t. In addition, we assume a global 3D environment prior e, associated each with input video, is available at both training and inference time. e is environment specific, *e.g.* , the 3D map of an apartment. Our goal is to jointly predict the action category y of x and the action location r on the 3D map. r is parameterized as a 3D saliency map, where the value of $r(w, d, h)$ represents the likelihood of action clip x happening in spatial location w, d, h.[1] r thereby defines a proper probabilistic distribution in 3D space.

[1]For tractability, we associate the entire activity with a specific 3D location and do not model location change over the course of an activity. This is a valid assumption for the activities we address, such as sitting

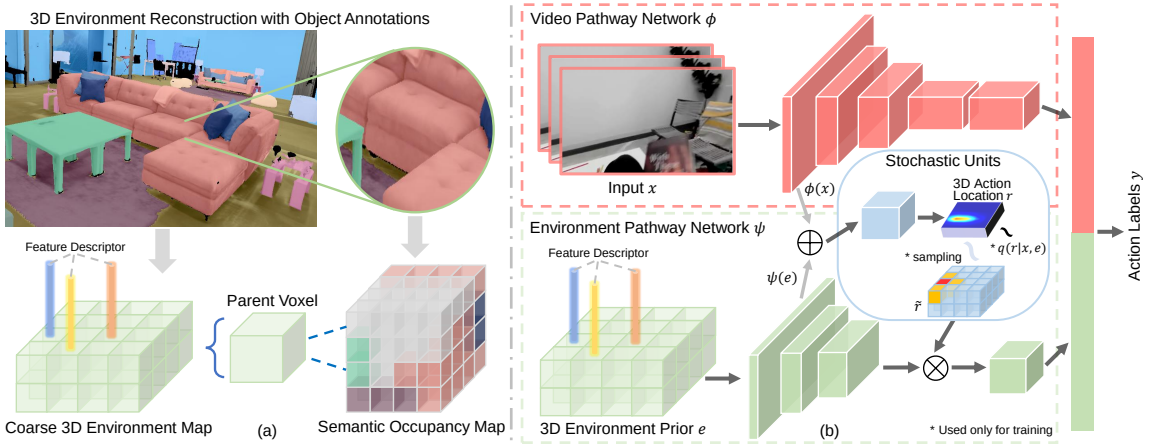

Figure 4.2: (a) **Hierarchical Volumetric Representation (HVR)**. We rasterize the semantic 3D environment mesh into two levels of 3D voxels. Each parent voxel corresponds to a possible action location, while the children voxels compose a semantic occupancy map that describes their parent voxel. (b) **Overview of our model**. Our model takes video clips x and the associated 3D environment representation e as inputs. We adopt an I3D backbone network ϕ to extract video features and a 3D convolutional network ψ to extract the global environment features. We then make use of stochastic units to generate sampled action location $\tilde{r}$ for selecting local 3D environment features for action recognition. Note that $\otimes$ represents weighted average pooling, while $\oplus$ denotes concatenation along channel dimension.

4.2.1 Joint Modeling with the 3D Environment Representation

3D Environment Representation. Existing 3D environment representations are not designed for egocentric action understanding. We seek to design a representation that not only encodes the 3D geometric and semantic information of the 3D environment, but is also effective for 3D action localization and recognition.

Inspired by previous works on human-scene interaction (see discussions in Chapter 2.4), we introduce a Hierarchical Volumetric Representation (HVR) of the 3D environment. We provide an illustration of our method in Figure 4.2(a). We assume the 3D environment reconstruction with object labels is given in advance as a 3D mesh (see Sec.4 for details). We first divide the 3D mesh into $X \times Y \times Z$ parent voxels, that define all possible action locations. We then divide each parent voxel into multiple voxels at a fixed resolution M and

down, playing keyboards, etc. This is different from the active research on SLAM [128]

further assign an object label to each child voxel based on the object annotation. Specifically, the object label of each child voxel is determined by the majority vote of the vertices that lie inside that child voxel. Note that we only consider static objects of the entire environments and treat empty space as a specially-designated "object" category. Therefore, the child voxels compose a semantic occupancy map that encodes both the 3D geometry and semantic meaning of the parent voxel.

We further vectorize the semantic occupancy map and use the resulting vector as a feature descriptor of the parent voxel. The 3D environment representation e can then be represented as a 4D tensor, with dimension $X \times Y \times Z \times (M^3)$. Note that higher resolution M can better approximate the 3D shape of the environment. Our proposed HVR is thus a compact and flexible environment representation that jointly considers the 3D action location candidates, geometric and semantic information of the 3D environment.

Joint Learning of Action Category and Action Location. We present an overview of our model in Figure 3.2(b). Specifically, we adopt a two-pathway network architecture. The video pathway extracts video features with an I3D backbone network $\phi(x)$, while the environment pathway extracts the global 3D environment features with a 3D convolutional network $\psi(e)$. Visual and environmental features are jointly considered for predicting the 3D action location r. We then adopt stochastic units to generate sampled action $\tilde{r}$ for selecting the local environment features relevant to the actions. Local environment features and video features are further fused together for activity recognition.

Our key idea is to utilize the 3D environment representation e for jointly modeling the action label y and 3D action location r of video clip x. We consider the action location r as a probabilistic variable, and model the action label y given input video x and environment representation e using a latent variable model. Therefore, the conditional probability $p(y|x, e)$ is given by:

$$p(y|x, e) = \int_r p(y|r, x, e)p(r|x, e)dr. \tag{4.1}$$

Notably, our proposed joint model has two key components. First, $p(r|x, e)$ models the 3D action location r from video input x and the 3D environment representation e. Second, $p(y|r, x, e)$ utilizes r to select a region of interest (ROI) from the environment representation e, and combines selected environment features with the video features from x for action classification. During training, our model receives the ground truth 3D action location and action label as supervisory signals. At inference time, our model jointly predicts both the 3D action location r and action label y. We now provide additional technical details in modeling $p(r|x, e)$ and $p(y|r, x, e)$.

4.2.2 3D Action Localization

We first introduce our 3D action localization module, defined by the conditional probability $p(r|x, e)$. Given the video pathway features $\phi(x)$ and the environment pathway features $\psi(e)$, we learn a mapping function to predict location r, which is defined on a 3D grid of candidate action locations[2]. The mapping function is composed of 3D convolution operations with parameters w_r and a softmax function. Thus, $p(r|x, e)$ is given by:

$$p(r|x, e) = softmax(w_r^T(\phi(x) \oplus \psi(e))), \tag{4.2}$$

where $\oplus$ denotes concatenation along the channel dimension. Therefore, the resulting action location r is a proper probabilistic distribution normalized in 3D space, and $r(w, d, h)$ can be considered as the expectation of video clip x happening in the spatial location (w, d, h) of the 3D environment.

In practice, we do not have access to the precise ground truth 3D action location and must rely on camera registration results as a proxy. Using a categorical distribution for $p(r|x, e)$ thus models the ambiguity of 2D to 3D registration. We follow [57, 129] to adopt stochastic units in our model. Specifically, we follow the Gumbel-Softmax and reparameterization trick from [105, 106] to adopt the following differentiable sampling mechanism:

[2]The 3D grid is defined globally over the 3D environment scan.

$$\tilde{r}_{w,d,h} \sim \frac{\exp((\log r_{w,d,h} + G_{w,d,h})/\theta)}{\sum_{w,d,h} \exp((\log r_{w,d,h} + G_{w,d,h})/\theta)}, \tag{4.3}$$

where G is a Gumbel Distribution for sampling from a discrete distribution. This Gumbel-Softmax trick produces a "soft" sample that allows the gradients propagation to video pathway network ϕ and environment pathway network ψ. θ is the temperature parameter that controls the shape of the soft sample distribution. We set $\theta = 2$ for our model. Notably, the expectation of sampled 3D action location $E[\tilde{r}]$ can be modeled by the distribution $p(r|x, e)$ using Equation 4.2.

4.2.3 Action Recognition with Environment Prior

Our model further models $p(y|r, x, e)$ with a mapping function $f(\tilde{r}, x, e)$ that jointly considers action location r, video input x and 3D environment representation e for action recognition. Formally, the conditional probability $p(y|r, x, e)$ can be modeled as:

$$p(y|r, x, e) = f(\tilde{r}, x, e) = softmax(w_p^T \Sigma(\phi(x) \oplus (\tilde{r} \otimes \psi(e)))), \tag{4.4}$$

where $\oplus$ denotes concatenation along channel dimension, and $\otimes$ denotes the element-wise multiplication. Specifically, our method uses the sampled action location $\tilde{r}$ for selectively aggregating environment features $\psi(e)$ and combines the aggregated environment features with video features $\phi(x)$ for action recognition. Σ denotes the average pooling operation that maps 3D feature to 2D feature, and w_p denotes the parameters of the linear classifier that maps feature vector to action prediction logits.

4.2.4 Training and Inference

We now present our training and inference schema. At training time, we assume a prior distribution of action location $q(r|x, e)$ is given as a supervisory signal. $q(r|x, e)$ is obtained by registering the egocentric camera into the 3D environment (see more details in Sec.4).

Note that we factorize $p(r|x, e)$ as latent variables, and based on the Evidence Lower Bound (ELBO), the resulting deep latent variable model has the following loss function:

$$\mathcal{L} = -\sum_{r} \log p(y|r, x, e) + KL[p(r|x, e)||q(r|x, e)], \tag{4.5}$$

where the first term is the cross entropy loss for action classification and the second term is the KL-Divergence that matches the predicted 3D action location distribution $p(r|x, e)$ to the prior distribution $q(r|x, e)$. During training, a single 3D action location sample $\tilde{r}$ for each input within the mini-batch will be drawn.

Theoretically, our model should sample $\tilde{r}$ from the same input multiple times and take average of the predictions at inference time. To avoid such dense sampling for high dimensional video input, we choose to directly plug in the deterministic action location r in Equation 4.4. Note that the recognition function f is composed of a linear mapping function and a softmax function, and therefore is convex. Further, $\tilde{r}$ is sampled from the probabilistic distribution of 3D action location r, similar to the formulation in [101, 129], r is thus the expectation of $\tilde{r}$. By Jensen's Inequality, we have:

$$E[f(\tilde{r}, x, e)] \geq f(E[\tilde{r}], x, e) = f(r, x, e). \tag{4.6}$$

That being said, $f(r, x, e)$ provides an empirical lower bound of $E[f(\tilde{r}, x, e)]$, and therefore provides a valid approximation of dense sampling.

4.3 Experiments and Results

We now present our experiments and results, we first describe the dataset and evaluation metrics. We then present our main results on action recognition and 3D action localization, followed by detailed ablation studies to verify our model design. We further show how our model generalizes to a novel environment, and provide visualization and discussion of our results.

4.3.1 Dataset and Benchmarks

Datasets. Note that existing egocentric video datasets (EGTEA [57], and EPIC-Kitchens [130], etc.) did not explicitly capture the 3D environment. We follow [74] to run ORB-SLAM on EGTEA and EPIC-Kitchens. However, less than 30% of frames can be registered, and the quality of the reconstructed point cloud is unsatisfactory. Our empirical finding is that existing visual SfM methods can not address the naturalistic egocentric videos. In contrast, the newly-developed Ego4D [7] dataset has a subset that includes egocentric videos, high-quality 3D environment reconstructions, and 3D static objects annotation.

The subset contains 60 hours of video from 105 different video sequences captured by Vuzix Blade Smart Glasses with a resolution of 1920×1080 at 24Hz. It captures 34 different indoor activities from three real-world living rooms, resulting in 6868 action clips. Similar to [130], we consider both *seen* and *unseen* environment splits. In the seen environment split, each environment is seen in both training and testing sets (5163 instances for training, and 1705 instances for testing). In the unseen split, all sequences from the same environment are either in training or testing (4392 instances for training, and 2476 instances for testing). We will release the both seen and unseen splits on this subset. As discussed in [7], the photo-realistic 3D reconstruction of the environment is obtained from the state-of-the-art dense reconstruction system [131]. Furthermore, the static 3D object meshes are annotated by painting an semantic label over the mesh polygons. 35 object categories plus a background class label are used in annotation. It is worthy noting that the static object annotations can be automated with the state-of-the-art 3D object detection algorithms.

Prior Distribution of 3D Activity Location. To obtain the ground truth of the activity location for each trimmed activity video clip, we first register the egocentric camera in the 3D environment using a RANSAC based feature matching method. Specifically, we first build a base map from the monochrome camera streams for 3D environment reconstruction using Structure from Motion [132, 133]. The pre-built base map is a dense point

Table 4.1: **Comparison with other forms of environment context**. Our Hierarchical Volumetric Representation (HVR) outperforms other methods by a significant margin on both action recognition and 3D action localization. The best results are highlighted with **boldface**, and the second-best results are underlined.

Method	Action Recognition		3D Action Localization		
	Mean Cls Acc	Top-1 Acc	Prec	Recall	F1
I3D-Res50	37.48	55.15	8.14	38.73	13.45
I3D+Obj	37.66	55.11	10.04	35.08	15.61
I3D+2DGround	38.69	55.37	10.88	36.19	16.73
I3D+SemVoxel	39.23	56.07	11.26	**38.77**	17.45
I3D+Affordance	39.95	55.82	11.55	35.35	17.41
Ours(HVR)	**41.64**	**56.94**	**16.71**	35.55	**22.73**

cloud associated with 3D feature points. We then estimate the camera pose of the video frame using active search [134]. Note that registering the 2D egocentric video frames in a 3D environment is fundamentally challenging, due to the drastic head rotation, featureless surfaces, and changing illumination. Therefore, we only consider the key frame camera registration, where enough inliers were matched with RANSAC. As introduced in Chapter 4.2, the action location is defined as a probabilistic distribution in 3D space. Thus, we map the key frame camera location into the index of the 3D action location tensor, with its value representing the likelihood of the given action happening in the corresponding parent voxel. To account for the uncertainty of 2D to 3D camera registration, we further enforce a Guassian distribution to generate the final 3D action location ground truth.

Evaluation Metrics. For all experiments, we evaluate the performance of both action recognition and 3D action localization, following the protocols.

• **Action Recognition**. We follow [130, 57] to report both Mean Class Accuracy and Top-1 Accuracy.

• **3D Action Localization**.We consider 3D action localization as binary classification over the regular 3D grids. Therefore, we report the Precision, Recall, and F1 score on a down-sampled 3D heatmap ($\times 4$ in X, Y direction, and $\times 2$ in Z direction) as in [101].

4.3.2 Action Understanding in Seen Environments

Our method is the first to utilize the 3D environment information for egocentric action recognition and 3D localization. Previous works have considered various environment contexts for other tasks, including 3D object detection, affordance prediction, etc. Therefore, we adapt previous proposed contextual cues into our proposed joint model and design the following strong baselines:

• **I3D-Res50** refers to the backbone network from [110]. We also use the network feature from I3D-Res50 for 3D action localization by adopting the KL loss.

• **I3D+Obj** uses object detection results from a pre-trained object detector [135] as contextual cues as in [26]. This representation is essentially an object-centric feature that describes the attended environment (*i.e.* , where the camera wearer is facing towards), therefore 3D action location can not used for selecting surrounding environment features.

• **I3D+2DGround** projects the object information from the 3D environment to 2D ground plane. A similar representation is also considered in [74]. Note that the predicted 3D action location will also be projected to 2D ground plane to select local environment features.

• **I3D+SemVoxel** is inspired by [136], where we use the semantic probabilistic distribution of all the vertices within each voxel as a feature descriptor. Therefore, the resulting environment representation is a 4D tensor with dimension $X \times Y \times Z \times C$, where X, Y, Z represent the spatial dimension, and C denotes the number of object labels from the 3D environment mesh annotation introduced in Sec.4.

• **I3D+Affordance** follows [86] to use the afforded action distribution as feature descriptor for each voxel. The resulting representation is a 4D tensor with dimension $X \times Y \times Z \times N$, where N denotes the number of action classes.The afforded action distribution is derived from the training set.

Results. Our results on the seen environment split is listed in Table 4.1. Our method outperforms I3D-Res50 baseline by a large margin ($4.2\%/1.8\%$ on Mean Cls Acc/Top1 Acc) on action recognition. We attribute this significant performance gain to explicitly modeling

the 3D environment context. As for 3D action localization, our method outperforms I3D-50 by 9.3% – a relative improvement of **69%**. Notably, predicting the 3D action location based on video sequence alone is erroneous. Our method, on the other hand, explicitly models the 3D environment factor and thus improves the performance of 3D action localization. In subsequent sections, we will show that the performance improvement does not simply come from additional input modalities of 3D environment, but attributes to a careful design of 3D representation and probabilistic joint modeling.

Comparison on environment representation. We now compare HVR with other forms of environment representation. As shown in Table 4.1, I3D+Obj has minor improvement on the over all performance, while I3D+2DGround, I3D+SemVoxel and I3D+Affordance can improve the performance of action recognition and 3D localization by a notable margin. Those results suggest that the environment context (even in 2D space) plays an important role in egocentric action understanding. More importantly, our method outperforms all previous methods by at least 1.7% for action recognition and 5.3% for 3D action localization. *These results suggest that our proposed HVR is superior to a 2D ground plane representation*, and demonstrates that using the semantic occupancy map as the environment descriptor can better facilitate egocentric understanding.

Although our method requires the annotation of static objects, we are the first to show how the 3D scene proximity can facilitate egocentric video understanding. Moreover, baseline methods like I3D+SemVoxel and I3D+2DGround both use the same static objects annotation to describe the environment context as our method. Therefore, it is a fair experiment comparison between our approach and those methods. I3D+Affordance also requires extra inputs, in the form of afforded action distribution across the entire 3D map, which directly links each voxel in the map to the most likely action for that location, a much stronger prior than our HVR representation. Note that the affordance map requires that the observation of action instances densely cover the full 3D scene. Therefore, we argue that I3D+Affordance is less scalable than our model.

Table 4.2: **Ablation study for the 3D representation**. We present the results of our method that adopts different semantic occupancy map resolution M.

Method	Action Recognition		3D Action Localization		
	Mean Cls Acc	Top-1 Acc	Prec	Recall	F1
I3D-Res50	37.48	55.15	8.14	38.73	13.45
I3D+SemVoxel	39.23	56.07	11.26	38.77	17.45
Ours ($M = 2$)	39.04	56.26	12.19	36.82	18.32
Ours ($M = 4$)	**41.64**	**56.94**	**16.71**	35.55	22.73
Ours ($M = 8$)	40.06	56.04	16.13	**39.84**	**22.96**

4.3.3 Ablation Studies

We now present detailed ablation studies of our method on seen split. To begin with, we analyze the role of semantic and geometric information in our hierarchical volumetric representation (HVR). We then present an experiment to verify whether fine-grained environment context is necessary for egocentric action understanding. Furthermore, we show the benefits of probabilistic joint modeling of action and 3D action location.

Semantic Meaning and 3D Geometry. The semantic occupancy map carries both geometric and semantic information of the local environment. To show how each component contributes to the performance boost, we compare Ours with I3D+SemVoxel, where only semantic meaning is considered, in Table 4.2. Ours outperforms I3D+SemVoxel by a notable margin for action recognition and a large margin for 3D localization. These results suggest that semantic occupancy map is more expressive than only semantic information for action understanding, yet it has smaller impact on action recognition than 3D action localization.

Granularity of 3D Information. We further show what level of 3D environment granularity is needed for egocentric action understanding. By the definition of occupancy map, increasing the resolution M of children voxels will approximate the actual 3D shape of the environment. Therefore, we report results of our method with different occupancy map resolution in Table 6.1. Not surprisingly, low occupancy map resolution lags behind Ours for action recognition by $2.6\% \downarrow$ and 3D action localization by $4.4\% \downarrow$, which again shows

Table 4.3: **Ablation study for joint modeling of action category and 3D action location**. Our proposed probabilistic joint modeling can consistently benefit the performance on action recognition and 3D action localization.

	Action Recognition		3D Action Localization		
Method	Mean Cls Acc	Top-1 Acc	Prec	Recall	F1
I3D-Res50	37.48	55.15	8.14	**38.73**	13.45
I3D+GlobalEnv	35.99	54.93	8.82	36.40	14.20
I3D+DetEnv	39.37	55.88	14.11	32.66	19.71
Ours	**41.64**	**56.94**	**16.71**	35.55	**22.73**

the necessity of incorporating the 3D geometric cues. Another interesting observation is that higher resolution can slightly increase the 3D action localization accuracy by 0.2%, yet decreases the performance on action recognition by $1.6\% \downarrow$. These results suggest that fine-grained 3D shape of the environment is not necessary for action recognition. In fact, higher resolution will dramatically increase the feature dimension of the environment representation, and thereby incurs more barriers to the network.

Joint Learning of Action Label and 3D Location. We denote a baseline model that directly fuses global environment features, extracted by the same 3D convolutional network adopted in our method, with video features for action grounding as I3D+GlobalEnv. The results are presented in Table 4.3. Interestingly, I3D+GlobalEnv decreases the performance of I3D-Res50 backbone network by $1.5\% \downarrow / 0.2\% \downarrow$ for action recognition and has marginal improvement for 3D action localization ($+0.8\%$). We speculate that this is because only 3 types of environment representation available for training may lead to overfitting. In contrast, our method makes use of the learned 3D action location to select interesting environment features associated with the action. As the action location varies among different input videos, our method can utilize the 3D environment context without running into the pitfall of overfitting, and therefore outperforms I3D+GlobalEnv by $5.7\%/2.0\%$ for action recognition and 8.5% for 3D action localization.

Probabilistic Modeling of 3D Action Location. As introduced in Sec.4, considerable uncertainty lies in the prior distribution of 3D action location, due to the challenging artifact of 2D to 3D camera registration. To verify that the probabilistic modeling can account for the

Table 4.4: **Experimental results on unseen environment split**. Our model show the capacity of better generalizing to an unseen environment with known 3D map. The best results are highlighted with **boldface**, and the second-best results are underlined.

Method	Action Recognition		3D Action Localization		
	Mean Cls Acc	Top-1 Acc	Prec	Recall	F1
I3D-Res50	29.24	52.22	6.20	45.14	10.90
I3D+2DObject	29.91	53.05	6.31	42.22	10.98
I3D+2DGround	30.06	53.87	6.95	41.27	11.90
I3D+SemVoxel	30.19	53.37	7.03	43.55	12.11
Ours	**31.55**	**55.33**	**7.50**	**44.97**	**12.86**

uncertainty of 3D action location ground truth, we compare our method with a deterministic version of our model, denoted as DetEnv. DetEnv adopts the same inputs and network architecture as our method, except for the differentiable sampling with Gumbel-Softmax Trick. As shown in Table 4.3, Ours outperforms DetEnv by 2.3% for action recognition and 3.0% for 3D action localization. These results demonstrate the benefits of the stochastic units adopted in our method.

Remarks. To summarize, our key finding is that both 3D geometric and semantic contexts convey important information for action recognition and 3D localization. Another important take home is that egocentric understanding only requires a sparse encoding of geometric information. Moreover, without a careful model design, the 3D environment representation has minor improvement on (or even decreases) the overall performance as reported in Table 4.3.

4.3.4 Generalization to Novel Environment

We further present experiment results on the unseen environment split in Table 4.4. Our model outperforms all baselines by a notable margin on both action recognition and 3D action localization. Note that the affordance map requires the observation of action instances on the 3D spatial location and thus cannot be applied on the unseen split. These results suggest that explicitly modeling the 3D environment context can improve the generalization ability to unseen environments with known 3D maps. However, the performance gap

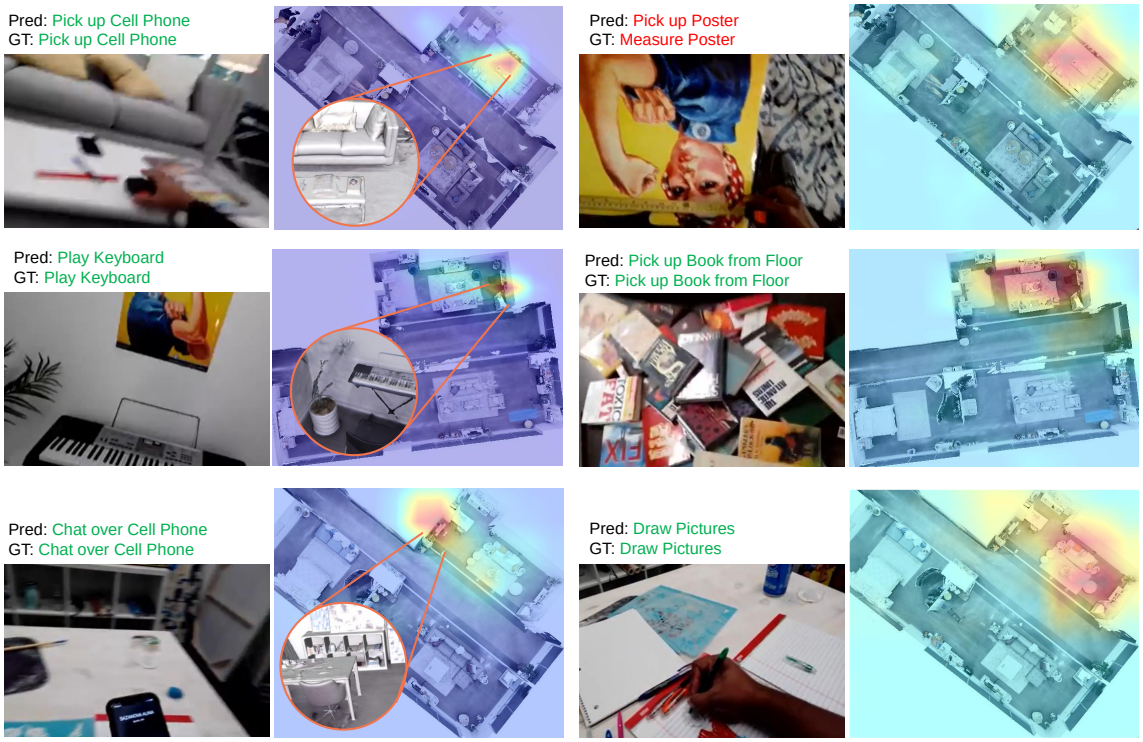

Figure 4.3: **Visualization of predicted 3D action location (projected on top-down view of the reconstructed 3D scene) and action labels (captions above the video frames)**. We present both successful and failure examples. We also show the "zoom-in" spatial region of the action location to help readers to better interpret our action localization results.

is smaller in comparison to the performance boost on seen split. We speculate that this is because we only have two different types of environments for training and therefore the risk of overfitting on unseen split is further exemplified.

4.3.5 Discussion

Visualization of Action Location. We visualize our results on seen environment split. Specifically, we project the 3D saliency map of action location on the top-down view of the 3D environments. As shown in Figure 4.3, our model can effectively localize the coarse action location and thereby select the region of interest from the global environment features for action recognition. By examining the failure cases, we found that the model may run into the failure modes when the video features are not sufficiently discriminative (*i.e.* , when the camera wearer is standing close to a white wall.) Another interesting observa-

tion is that the model may output a “diffused” heatmap, when the foreground active objects take up the majority of the video frames (right column of Figure 4.3). This is because the model receives uniform prior as supervisory signals when the camera registration fails for an action clip. In these cases, our model opts for predicting a diffused heat map of action location to prevent itself from missing important environment features. In doing so, our model might still be able to successfully predict the action labels, despite the failure of camera registration.

4.4 Conclusion

In this chapter, I introduce a deep model that makes use of egocentric videos and a 3D map to address the novel task of joint action recognition and 3D localization. The key insight is that the 3D geometric and semantic context of the surrounding environment provides critical information that complements video features for action understanding. The key innovation of our model is to characterize the 3D action location as a latent variable, which is used to select the surrounding local environment features for action recognition. Our model demonstrated impressive results on seen and unseen environments when evaluated on the newly released Ego4D dataset [7]. We believe our work provides a critical first step towards understanding actions in the context of a 3D environment, and points to exciting future directions in connecting egocentric vision and 3D scene understanding for Augmented Reality and Human-Robot Interaction.

CHAPTER 5

HUMAN-OBJECT INTERACTION ANTICIPATION IN EGOCENTRIC VIDEO

5.1 Introduction

The human ability of "looking into the near future" remains a key challenge for computer vision. Consider the example in Figure 5.1, given a video shortly before the start of an action, we can easily predict what will happen next, e.g., the person will take the canister of salt. Even without seeing any future frames, we can vividly imagine how the person will perform the action, e.g., the trajectory of the hand when reaching for the canister or the location on the canister that will be grasped.

There is convincing evidence that our remarkable ability to forecast other individuals' actions depends critically upon our perception and interpretation of their body motion. The investigation of this anticipatory mechanism dates back to 19th century, when William James argued that future expectations are intrinsically related to purposive body movements [137]. Additional evidence for a link between perceiving and performing actions was provided by the discovery of mirror neurons [138, 139]. The observation of others' actions activates our motor cortex, the same brain regions that are in charge of the planning and control of intentional body motion. This activation can happen even before the onset of the action and is highly correlated with the anticipation accuracy [140]. A compelling explanation from [141] suggests that *motor attention*, i.e., the active prediction of meaningful future body movements, serves as a key representation for anticipation. In this chapter, I introduce our efforts on developing the first computational model for motor attention that can enable more accurate action prediction.

Despite these relevant findings in cognitive neuroscience, the role of intentional body motion in action anticipation is largely ignored by the existing literature [142, 143, 144,

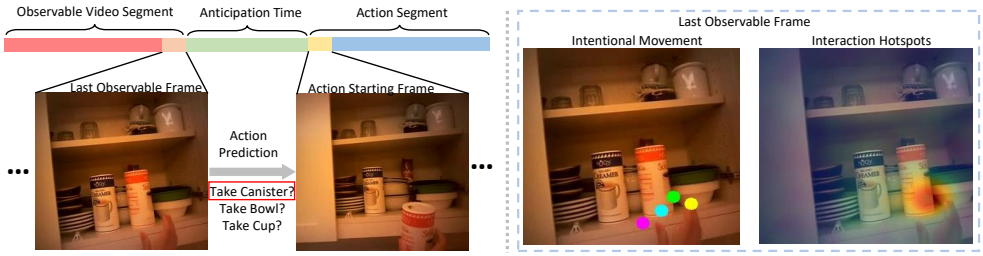

Figure 5.1: **What is the most likely future interaction**? Our model takes advantage of the connection between motor attention and visual perception. In addition to future action label, our model also predicts the interaction hotspots on the last observable frame and hand trajectory (in the order of yellow , green, cyan, and magenta) between the last observable time step to action starting point. Visualizations of hand trajectory are projected to the last observable frame (best viewed in color).

145, 25, 26, 24, 27]. In this work, we focus on the problem of forecasting human-object interactions in First Person Vision (FPV). Interactions consist of a single verb and one or more nouns, with "take bowl" as an example. FPV videos capture complex hand movements during a rich set of interactions, thus providing a powerful vehicle for studying the connection between motor attention and future representation. Several previous works have investigated the problems of FPV activity anticipation [25, 26] and body movement prediction [146, 88, 87, 90]. We believe we are the first to utilize a motor attention model for FPV action anticipation.

Specifically, we propose a novel deep model that predicts "motor attention"—the future trajectory of the hands, as an anticipatory representation of actions. Based on motor attention, our model further localizes the future contact region of the interaction, i.e., interaction hotspots [83] and recognizes the type of future interactions. Importantly, we characterize motor attention and interaction hotspots as probabilistic variables modeled by stochastic units in a deep network. These units naturally deal with the uncertainty of future hand motion and contact region during interaction, and produce attention maps that highlight discriminative spatial-temporal features for action anticipation.

During inference, our model takes video clips shortly before the interaction as inputs, and jointly predicts motor attention, interaction hotspots, and action labels. During training,

our model assumes that these outputs are available as supervisory signals. To evaluate our model, we report results on two major FPV benchmarks: EGTEA Gaze+ and EPIC-Kitchens. In the experiment section, we show our approach outperforms prior state-of-the-art methods by a significant margin. In addition, we present extensive ablation studies to verify the design of our model and evaluate our model for motor attention prediction and interaction hotspots estimation. Our model demonstrates strong results for both tasks.

To summarize, this chapter has the following contributions:

- I propose a novel joint model for predicting motor attention, interaction hotspots, and future action categories.
- I present a systematic ablation study to demonstrate the benefits of motor attention module, interaction hotspots module, and stochastic units adopted in our method.
- Our proposed method achieved the state-of-the-art performance on existing egocentric benchmark datasets.

This work was a collaboration with Prof. Yin Li, and Prof. Siyu Tang. The work was published in ECCV 2020 [129] as an oral paper.

5.2 Problem Setting and Model Overview

We consider the setting of action anticipation from [147]. Denote an input video segment as $x : [\tau_a - \Delta\tau_o, \tau_a]$. x starts at $\tau_a - \Delta\tau_o$ and ends at τ_a with duration $\Delta\tau_o > 0$ as the "observation time". Our goal is to predict the label y of an immediate future interaction starting at $\tau_s = \tau_a + \Delta\tau_a$, where $\Delta\tau_a > 0$ is a fixed interval known as the "anticipation time." Moreover, we seek to estimate future hand trajectories $\mathcal{M}$ within $[\tau_a, \tau_s]$ (projected back to the last observable frame at τ_a), and to localize interaction hotspots $\mathcal{A}$ at τ_a (the last observable frame).

To summarize, our model seeks to anticipate the future action y by jointly predicting the future hand trajectory $\mathcal{M}$ and interaction hotspots $\mathcal{A}$ at the last observable frame. Predicting

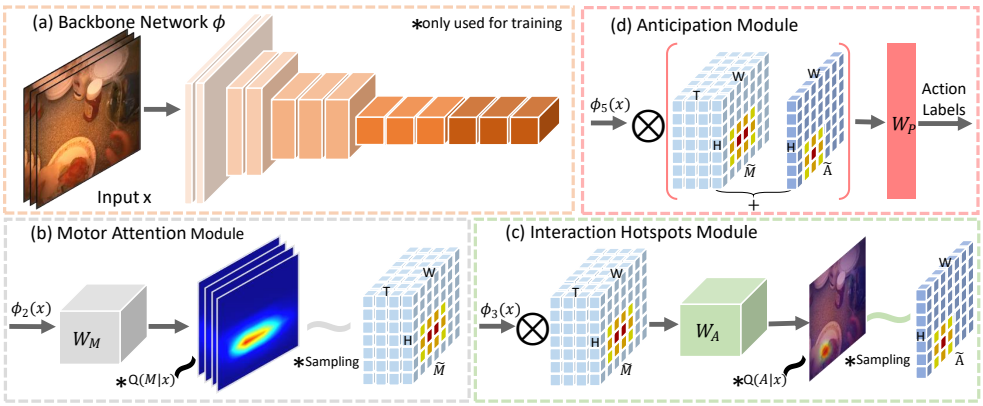

Figure 5.2: **Proposed model for forecasting egocentric human-object interaction**. A 3D convolutional network $\phi(x)$ is used as our backbone network, with features from its ith convolution block as $\phi_i(x)$ (a). A motor attention module (b) makes use of stochastic units to generate sampled future hand trajectories $\tilde{\mathcal{M}}$ used to guide interaction hotspots estimation in module (c). Module (c) further generates sampled interaction hotspots $\tilde{\mathcal{A}}$ with similar stochastic units as in module (b). Both $\tilde{\mathcal{M}}$ and $\tilde{\mathcal{A}}$ are used to guide action anticipation in anticipation module (d). During testing, our model takes only video clips as inputs, and predicts motor attention, interaction hotspots, and action labels. Note that $\otimes$ represents element-wise multiplication for weighted pooling.

the future is fundamentally ambiguous, since the observation of future interaction only represents one of the many possibilities characterized by an underlying distribution. Our key idea is thus to model motor attention and interaction hotspots as probabilistic variables in order to account for their uncertainty. We present an overview of our model in Figure 5.2.

Specifically, we make use of a 3D backbone network $\phi(x)$ for video representation learning. Following the approach in [148, 149], we utilize 5 convolutional blocks, and denote the features from the ith convolution block as $\phi_i(x)$. Based on $\phi(x)$, our motor attention module (b) predicts future hand trajectories as motor attention $\mathcal{M}$ and uses stochastic units to sample from $\mathcal{M}$. The sampled motor attention $\tilde{\mathcal{M}}$ is an indicator of important spatial-temporal features for interaction hotspot estimation. Our interaction hotspot module (c) further produces an interaction hotspot distribution $\mathcal{A}$ and its sample $\tilde{\mathcal{A}}$. Finally, our anticipation module (d) makes use of both $\tilde{\mathcal{M}}$ and $\tilde{\mathcal{A}}$ to aggregate network features, and predicts the future interaction y.

5.3 Joint Modeling of Human-Object Interaction

Formally, we consider motor attention $\mathcal{M}$ and interaction hotspots $\mathcal{A}$ as probabilistic variables, and model the conditional probability of the future action label y given the input video x as a latent variable model, where

$$p(y|x) = \int_{\mathcal{M}} \int_{\mathcal{A}} p(y|\mathcal{A}, \mathcal{M}, x) p(\mathcal{A}|\mathcal{M}, x) p(\mathcal{M}|x) \, d\mathcal{A} \, d\mathcal{M}, \tag{5.1}$$

$p(\mathcal{M}|x)$ first estimates motor attention from video input x. $\mathcal{M}$ is further used to estimate interaction hotspots A ($p(\mathcal{A}|\mathcal{M}, x)$). Given x, $\mathcal{M}$ and $\mathcal{A}$, the action label y is determined by $p(y|\mathcal{A}, \mathcal{M}, x)$. Our model thus consists of three main components.

Motor Attention Module tackles $p(\mathcal{M}|x)$. Given the network features $\phi_2(x)$, our model uses a function F_M to predict motor attention $\mathcal{M}$. $\mathcal{M}$ is represented as a 3D tensor of size $T_m \times H_m \times W_m$. Moreover, $\mathcal{M}$ is normalized within each temporal slice, i.e., $\sum_{w,h} \mathcal{M}(t, w, h) = 1$.

Interaction Hotspots Module targets at $p(\mathcal{A}|\mathcal{M}, x)$. Our model uses a function F_A to estimate the interaction hotspots $\mathcal{A}$ based on the network feature $\phi_3(x)$ and sampled motor attention $\tilde{\mathcal{M}}$. $\mathcal{A}$ is represented as a 2D attention map of size $H_a \times W_a$. A further normalization constrained that $\sum_{w,h} \mathcal{A}(w, h) = 1$.

Anticipation Module makes use of the predicted motor attention and interaction hotspots for action anticipation. Specifically, sampled motor attention $\tilde{\mathcal{M}}$ and sampled interaction hotspots $\tilde{\mathcal{A}}$ are used to aggregate feature $\phi_5(x)$ via weighted pooling. An action anticipation function F_P further maps the aggregated features to future action label y.

5.3.1 Motor Attention Module

Motor Attention Generation. The motor attention prediction function F_M is composed of a linear function with parameter W_M on top of network features $\phi_2(x)$. The linear function is realized by a 3D convolution and a softmax function is used to normalize the attention map. This is given by $\psi = softmax(W_M^T \phi_2(x))$, where the output ψ is a 3D tensor of size

$T_m \times H_m \times W_m$. We further model $p(\mathcal{M}|x)$ by normalizing ψ within each temporal slice:

$$\mathcal{M}_{m,n,t} = \frac{\psi_{m,n,t}}{\sum_{m,n} \psi_{m,n,t}}, \tag{5.2}$$

where $\psi_{m,n,t}$ is the value at location (m, n) and time step t in the 3D tensor of ψ. And $\mathcal{M}$ can be considered as the expectation of $p(\mathcal{M}|x)$.

Stochastic Modeling. Modeling motor attention in the context of forecasting human-object interaction requires a mechanism for addressing the stochastic nature of motor attention in developing the joint model. Here, we propose to use stochastic units to model the uncertainty. The key idea is to sample from the motor attention distribution. We follow the Gumbel-Softmax and reparameterization trick introduced in [105, 106] to design a differentiable sampling mechanism:

$$\tilde{\mathcal{M}}_{m,n,t} \sim \frac{\exp((\log \psi_{m,n,t} + G_{m,n,t})/\theta)}{\sum_{m,n} \exp((\log \psi_{m,n,t} + G_{m,n,t})/\theta)}, \tag{5.3}$$

where G is a Gumbel Distribution used to sample from discrete distribution. This Gumbel-Softmax trick produces a "soft" sampling step that allows the direct back-propagation of gradients to ψ. θ is the temperature parameter that controls the "sharpness" of the distribution. We set $\theta = 2$ for all of our experiments.

5.3.2 Interaction Hotspots Module

The predicted motor attention $\mathcal{M}$ is further used to guide interaction hotspots estimation $p(\mathcal{A}|x)$ by considering the conditional probability

$$p(\mathcal{A}|x) = \int_{\mathcal{M}} p(\mathcal{A}|\mathcal{M}, x) p(\mathcal{M}|x) d\mathcal{M}. \tag{5.4}$$

In practice, $p(\mathcal{A}|x)$ is estimated using sampled motor attention $\tilde{\mathcal{M}}$ based on $p(\mathcal{A}|\tilde{\mathcal{M}}, x)$ and $p(\tilde{\mathcal{M}}|x)$. For each sample $\tilde{\mathcal{M}}$, $p(\mathcal{A}|\tilde{\mathcal{M}}, x)$ is defined by the interaction hotspots estimation function F_A. F_A takes the input of a motor attention map $\tilde{\mathcal{M}}$ and $\phi_3(x)$, and has the form of a linear 2D convolution parameterzied by W_A followed by a softmax function.

$$p(\mathcal{A}|\tilde{\mathcal{M}}, x) = softmax\left(W_A^T(\tilde{\mathcal{M}} \otimes \phi_3(x))\right), \tag{5.5}$$

where $\otimes$ is the Hadamard product (element-wise multiplication). The result $p(\mathcal{A}|\mathcal{M}, x)$ is a 2D map of size $H_a \times W_a$. Intuitively, $\tilde{\mathcal{M}}$ presents a spatial-temporal saliency map to highlight feature representation $\phi_3(x)$. F_A thus normalizes (using softmax) the output of a linear model on the selected features $\tilde{\mathcal{M}} \otimes \phi_3(x)$, and is a convex function. Finally, a similar sampling mechanism as in Eq. Equation 5.3 can be used to sample $\tilde{\mathcal{A}}$ from $p(\mathcal{A}|x)$.

5.3.3 Anticipation Module

We now present the last piece of our model—the action anticipation module. The action anticipation function $p(y|\mathcal{A}, \mathcal{M}, x) = F_P(\mathcal{A}, \mathcal{M}, x)$ is defined as a function of the sampled motor attention map (3D) $\tilde{\mathcal{M}}$, sampled interaction heatmap (2D) $\tilde{\mathcal{A}}$ and the network feature $\phi_5(x)$. This is given by

$$p(y|\tilde{\mathcal{A}}, \tilde{\mathcal{M}}, x) = softmax\left(W_P^T \Sigma\left(\tilde{\mathcal{M}} \otimes \phi_5(x)\right) + W_P^T \Sigma\left(\tilde{\mathcal{A}} \odot \phi_5(x)\right)\right), \tag{5.6}$$

where $\otimes$ is again the Hadamard product. Σ is the global average pooling operation that pools a vector representation from a 2D or 3D feature map. $\odot$ is to use a 2D map ($\tilde{\mathcal{A}}$) to conduct Hadamard product to the last temporal slice of a 3D tensor $\phi_5(x)$. This is because the interaction hotspots $\tilde{\mathcal{A}}$ is only defined on the last observable frame. W_P is a linear function that maps the features into prediction logits. F_P is a combination of linear operations followed by a softmax function, and thus remains a convex function.

5.3.4 Training and Inference

Training our proposed joint model is challenging, as $p(\mathcal{M}|x)$ and $p(\mathcal{A}|\mathcal{M}, x)$ are intractable. Fortunately, variational inference comes to the rescue.

Prior Distribution. During training, we assume that reference distributions of future hand position $Q(\mathcal{M}|x)$ and interaction hotspots $Q(\mathcal{A}|x)$ are known in prior. These distributions are derived from manual annotation.

Variational Learning. Our proposed model seeks to jointly predict motor attention $\mathcal{M}$, interaction hotspots $\mathcal{A}$, and the action label y. Specifically, we inject posterior $p(\mathcal{A}, \mathcal{M}|x)$

into $p(y|x)$ and optimize the resulting latent variable model by maximizing the Evidence Lower Bound (ELBO). However, the prior distribution of $Q(\mathcal{A}, \mathcal{M}|x)$ is not available for training. Hence, we further approximate $p(\mathcal{A}, \mathcal{M}|x)$ by factorizing it into $p(\mathcal{A}|x)$ and $p(\mathcal{M}|x)$. Namely, we assume that $\mathcal{A}$ and $\mathcal{M}$ is conditionally independent given the input x. Thus, we have

$$\begin{aligned}
&KL[p(\mathcal{A}, \mathcal{M}|x)||Q(\mathcal{A}, \mathcal{M}|x)] \\
=&KL[p(\mathcal{A}|\mathcal{M}, x)||Q(\mathcal{A}|\mathcal{M}, x)] + KL[p(\mathcal{M}|x)||Q(\mathcal{M}|x)].
\end{aligned}$$

The ELBO of our proposed joint model can be derived as

$$\begin{aligned}
\log p(y|x) &\geq E_{p(\mathcal{A},\mathcal{M}|x)}[\log p(y|\mathcal{A}, \mathcal{M}, x)] - log(p(\mathcal{A}, \mathcal{M}|x))] \\
&= \sum_{\mathcal{A},\mathcal{M}} \log p(y|\mathcal{A}, \mathcal{M}, x) - KL[p(\mathcal{A}, \mathcal{M}|x)||Q(\mathcal{A}, \mathcal{M}|x)] \\
&= \sum_{\mathcal{A},\mathcal{M}} \log p(y|\mathcal{A}, \mathcal{M}, x) - KL[p(\mathcal{A}|x)||Q(\mathcal{A}|x)] - KL[p(\mathcal{M}|x)||Q(\mathcal{M}|x)].
\end{aligned}$$

Therefore, the loss function $\mathcal{L}$ is given by

$$\mathcal{L} = -\sum_{\mathcal{A},\mathcal{M}} \log p(y|\mathcal{A}, \mathcal{M}, x) + KL[p(\mathcal{A}|x)||Q(\mathcal{A}|x)] + KL[p(\mathcal{M}|x)||Q(\mathcal{M}|x)]. \tag{5.7}$$

The first term in the loss function is the cross entropy loss for action anticipation. The last two terms use KL-Divergence to align the predicted distributions of motor attention $p(\mathcal{M}|x)$ and interaction hotspots $p(\mathcal{A}|x)$ to prior distributions ($Q(\mathcal{M}|x)$ and $Q(\mathcal{A}|x)$).

Approximate Inference. At inference time, our model could have drawn many samples of motor attention $\tilde{\mathcal{M}}$ and interaction hotspots $\tilde{\mathcal{A}}$ for the anticipation. However, the sampling and averaging is computationally expensive. We choose to feed deterministic $\mathcal{M}$ and $\mathcal{A}$ into Eq. Equation 5.5 and Eq. Equation 5.6 at inference time. Note that F_A and F_P are convex, since they are composed of linear mapping function and softmax function. By

Jensen's inequality, we have

$$E[F_A(\tilde{\mathcal{M}}, x)] \geq F_A(E[\tilde{\mathcal{M}}], x) = F_A(\mathcal{M}, x), \tag{5.8}$$

$$E[F_P(\tilde{\mathcal{A}}, \tilde{\mathcal{M}}, x)] \geq F_P(E[\tilde{\mathcal{A}}], E[\tilde{\mathcal{M}}], x) = F_P(\mathcal{A}, \mathcal{M}, x) \tag{5.9}$$

Therefore, such approximation does serves as a shortcut to avoid sampling during testing, by providing a valid lower bound of $E[F_P(\tilde{\mathcal{A}}, \tilde{\mathcal{M}}, x)]$ and $E[F_A(\tilde{\mathcal{M}}, x)]$,

5.4 Experiment and Results

5.4.1 Dataset and Benchmark

Datasets. We make use of two FPV datasets: EGTEA Gaze+ [101, 57] and Epic-Kitchens [147]. We report results on the first split of the EGTEA dataset and follow [26] to split the public training set into training and validation sets with 2513 action classes. We set the anticipation time as 0.5 seconds for EGTEA and 1 second [147] for EPIC-Kitchens.

Evaluation Metrics. Our model is evaluated for action anticipation, and interaction hotspots estimation across EGTEA (using split1) and EPIC-Kitchens (using the train/val split from [26]). Specifically, we consider the following metrics:

• *Action Anticipation*. We report Top1/Mean Class accuracy on EGTEA as in [4] and Top1/Top5 accuracy as on EPIC-Kitchens following [26].

• *Interaction Hotspots Estimation*. We report F1 score as in [101] and KL-Divergence (KLD) as in [83] using a downsampled heatmap (32x) at the last observable frame.

• *Motor Attention Prediction*. We report the average and final displacement errors between the most confident location on a predicted attention map and the ground-truth hand points, similar to previous work on trajectory prediction [150]. Note that the motor attention maps are downsampled by a factor of 32/8 in space/time. Hence, we report displacement errors normalized in spatial and temporal dimension.

Table 5.1: **Action anticipation results on Epic-Kitchens**. Ours+Obj model outperforms state-of-the-art by a notable margin. See discussions of Ours+Obj in Chapter 5.4.2.

	Method	Top1/Top5 Accuracy		
		Verb	Noun	Action
s1	2SCNN [147]	29.76 / 76.03	15.15 / 38.65	4.32 / 15.21
	TSN [147]	31.81 / 76.56	16.22 / 42.15	6.00 / 18.21
	TSN+MCE [151]	27.92 / 73.59	16.09 / 39.32	10.76 / 25.28
	Trans R(2+1)D [24]	30.74 / 76.21	16.47 / 42.72	9.74 / 25.44
	RULSTM [26]	33.04 / **79.55**	22.78 / 50.95	14.39 / 33.73
	Ours	34.99 / 77.05	20.86 / 46.45	14.04 / 31.29
	Ours+Obj	**36.25** / 79.15	**23.83 / 51.98**	**15.42 / 34.29**
s2	2SCNN [147]	25.23 / 68.66	9.97 / 27.38	2.29 / 9.35
	TSN [147]	25.30 / 68.32	10.41 / 29.50	2.39 / 9.63
	TSN+MCE [151]	21.27 / 63.66	9.90 / 25.50	5.57 / 25.28
	Trans R(2+1)D [24]	28.37 / 69.96	12.43 / 32.20	7.24 / 19.29
	RULSTM [26]	27.01 / 69.55	15.19 / 34.38	8.16 / 21.20
	Ours	28.27 / 70.67	14.07 / 34.35	8.64 / 22.91
	Ours+Obj	**29.87 / 71.77**	**16.80 / 38.96**	**9.94 / 23.69**

Table 5.2: **Comparison between our methods and previous state-of-the-art results RULSTM.** See Chapter 5.4.2 of our submission for discussion of Ours+Obj.

Method	**Tasks**	**Training Supervision**	**Testing Inputs**	**End-to-End**
RULSTM [15]	Action Anticipation	Action Labels Object Cls & Boxes	RGB + Object Feat. + Flow	No
Ours	Action Anticipation Visual Affordance Motor Attention Pred	Action Labels Hand & Hotspots	RGB	Yes
Ours+Obj	Action Anticipation Visual Affordance Motor Attention Pred	Action Labels Object Cls & Boxes Hand & Hotspots	RGB + Object Feat.	No

5.4.2 Action Anticipation Results on EPIC-Kitchens

To compete for EPIC-Kitchens anticipation challenge, we used the backbone network CSN152. We trained our model on the public training set and report results using top-1/5 accuracy as in [147]. Table 5.1 compares our results to latest methods on EPIC-Kitchens. Our model outperforms strong baselines (TSN and 2SCNN) reported in [147] by a large margin. Compared to previous best results from RULSTM [26], our model archives +2%/-1.9%/-0.3% for verb/noun/action on seen set, and +1.3%/-1.1%/+0.6% on unseen set of EPIC-Kitchens. Our results are better for verb, worse for noun and comparable or better for actions. Notably, RULSTM requires object boxes & optical flow for training and ob-

ject features & optical flow for testing. In contrast, our method uses hand trajectories and interaction hotspots for training and needs *only RGB frames* for testing.

To further improve the performance, we fuse the object stream from RULSTM with our model (Ours+Obj). Compared to RULSTM, Ours+Obj has a performance gain of $+3.2\%$/$+2.9\%$ for verb, $+1.1\%$/$+1.6\%$ for noun, and $+1.0\%$/$+1.8\%$ for action (seen/unseen). It is worthy pointing out that RULSTM benefits from an extra flow network, while ours+Obj model takes additional supervisory signals of hands and hotspots. Note that our performance boost does not simply come from those extra annotations. In a subsequent ablation study, we have shown that simply training with these extra annotations has minor improvement, when used without our proposed probabilistic deep model.

We note that it is not possible to make a direct apples-to-apples comparison between our model and RULSTM [26], as the two models used vastly different training signals. In Table 5.2, we present the experiment setup of our method and RULSTM. Both RULTM and our model (Ours) use various supervisory signals for training, yet our model only needs RGB frames for inference and is end-to-end trainable. Ours+Obj model does require more training signals in comparison to RULSTM, yet it does not need optical flow for two-stream architecture. We have to point out that, from practical prospective, we care more about the data modality during testing time. Therefore, using more supervisory signals for training does not compromise the contribution of our method. Moreover, our method also address the challenging problem of motor attention prediction and interaction hotspots estimation. In terms of performance, our model is comparable to RULSTM without using any side information for inference. When using additional object stream during inference as in RULSTM, our model outperforms RULSTM by a relative improvement of **7%/22%** on seen/unseen set. More importantly, our model also provides the additional capabilities of predicting future hand trajectories and estimating interaction hotspots.

Table 5.3: **Ablation study for action anticipation**. We compare our model with backbone I3D network, and further analyze the role of motor attention prediction, interaction hotspots estimation, and stochastic units in joint modelling. See discussions in Chapter 5.4.3.

Method	EGTEA			Epic-Kitchens		
	Top1 Accuracy / Mean Cls Accuracy			Top1 Accuracy / Top5 Accuracy		
	Verb	Noun	Action	Verb	Noun	Action
I3D-Res50	48.01/31.25	42.11/30.01	34.82/23.20	30.06/76.86	16.07/41.67	9.60/24.29
JointDet	48.58/32.21	43.95/31.26	35.69/23.59	30.16/**76.86**	16.25/41.71	9.76/24.40
Hotspots Only	47.95/31.94	44.02/32.53	35.50/23.82	30.21/75.93	16.57/42.28	9.66/24.33
Motor Only	**49.35**/32.34	**45.69/33.93**	36.49/25.13	30.63/76.69	17.28/42.56	10.21/25.32
Ours	48.96/**32.48**	45.50/32.73	**36.60/25.30**	**30.65**/76.53	**17.40/42.60**	**10.38/25.48**

Table 5.4: **Ablation study for interaction hotspots estimation**. Jointly modeling motor attention with stochastic units can greatly benefit the performance of interaction hotspots estimation. (↑/↓ indicates higher/lower is better)

Method	EGTEA				Epic-Kitchens			
	Prec ↑	Recall ↑	F1 ↑	KLD ↓	Prec ↑	Recall ↑	F1 ↑	KLD ↓
I3DHeatmap	12.82	37.53	19.11	2.66	17.20	77.39	28.15	3.07
JointDet	16.11	41.82	23.26	1.84	17.32	85.79	28.83	2.21
Ours	**17.43**	**48.81**	**25.69**	**1.62**	**17.86**	**86.59**	**29.60**	**1.99**

5.4.3 Ablation Study

We present ablation studies of our model. We evaluate each component of our model, and then contrast our method to a series of baselines on motor attention prediction and interaction hotspot estimation. For all of our ablation studies, we adopt the lightweight I3D-Res50 [110] as backbone network to reduce computational cost. Our model is evaluated for action anticipation, motor attention prediction and interaction hotspots estimation across EGTEA (using split1) and EPIC-Kitchens (using the train/val split from [26]).

Benefits of Joint Modeling. As a starting point, we compare our model with a backbone I3D-Res50 model. We present the results of action anticipation in Table 5.3. Specifically, our model improves noun and action prediction by $+3.4\%/1.8\%$ on EGTEA and $+1.3\%/0.8\%$ on EPIC-Kitchens. Moreover, we show that our model improves the performance of interaction hotspots estimation. We consider the baseline I3D model that only estimates interaction region with interaction hotspots module as I3DHeatmap. As shown in Table 5.4, our model improves the F1 score by $6.6\%/1.5\%$ on EGTEA/EPIC-Kitchens.

Stochastic Modeling vs. Deterministic Modeling. We further evaluate the benefits of probabilistic modeling of motor attention and interaction hotspots. To this end, we compare our model with a deterministic joint model (*JointDet*). JointDet has the same architecture as our model, except for the stochastic units. As shown in Table 5.3, JointDet slightly improve the I3D baseline for action anticipation ($+0.87\%$ on EGTEA and $+0.16\%$ on EPIC-Kitchens), yet lags behind our probabilistic model. Specifically, our model outperforms JointDet by 0.91% and 0.62% on EGTEA and EPIC-Kitchens. Moreover, in comparison to JointDet, our model has better performance for interaction hotspots estimation ($+2.4\%/+0.8\%$ in F1 scores on EGTEA/EPIC-Kitchens). These results suggest that simply training with extra annotations might fail to capture the uncertainty of visual anticipation. In contrast, our design choice of probabilistic modeling can effectively deal with those uncertainty, therefore helps to improve the performance of joint modeling.

Motor Attention vs. Interaction Hotspots. Furthermore, we evaluate the contributions of motor attention and interaction hotspots for FPV action anticipation. We consider two baseline models in Table 5.4: I3D model equipped with only motor attention module (*Motor Only*), and I3D model equipped with only interaction hotspots module (*Hotspots Only*). Both models underperform the full model across the two datasets, yet the gap between *Motor Only* and the full model is smaller. These results suggest that both components contribute to the performance boost of action anticipation, yet the modeling of motor attention weights more than the modeling of interaction hotspots.

Interaction Hotspots Estimation. We present additional results on interaction hotspots estimation. We compare our results to the following baselines:

•*Center Prior* represents a Gaussian Distribution at the center of the image.

•*Grad-Cam* uses the same I3D network, and produces a saliency map via Grad-Cam [97].

•*EgoGaze* considers possible gaze position as salient region of a given image. This model is trained on eye fixation annotation from EGTEA-Gaze+ [152]. The assumption is that the person is likely to look at the interaction hotspots.

Table 5.5: **Motor attention prediction results on EGTEA**. Our model compares favourably to strong baselines. (↑/↓ indicates higher/lower is better)

Method	Avg. Disp. Error ↓	Final Disp. Error ↓
Kalman Filter	0.32	0.48
GPR	0.29	0.37
LSTM	**0.22**	**0.35**
Ours	0.23	0.36

•*DSS Saliency* predicts salient region during human object interaction. This model is trained on pixel-level saliency annotation from [124].

•*EgoHotspots* is the latest work [83] for estimating interaction hotspots.

Our results are shown in Table 5.4. Our model outperforms the best baselines (EgoGaze and EgoHotspots) by 5.4% on EGTEA and 3.6% on EPIC-Kitchens in F1 scores. These results suggest that our proposed joint model can effectively identify future interaction region. Another observation is that our model performs better on EPIC-Kitchens than EGTEA. This is probably due to the larger number of available training samples.

Motor Attention Prediction. We report our results on motor attention prediction on EGTEA dataset. We consider the following baselines:

•*Kalman Filter* describes the hand trajectory prediction problem with the state-space model, and assumes linear acceleration during the update step.

•*Gaussian Process Regression (GPR)* iteratively predicts the future hand position using Gaussian Process Regression.

•*LSTM* adopts a vanilla LSTM network for trajectory forecasting. We use the implementation from [150].

The results are presented in Table 5.5. Our model outperforms Kalman filter and GPR, yet is slightly worse than LSTM model (+0.01 in both errors). Note that all baseline methods need the coordinate of the first observed hand for prediction. This simplifies trajectory prediction into a less challenging regression problem. In contrast, our model does not need hand coordinates for inference. A model that relies on the observation of hand positions will encounter failure cases when the hand has not been observed, while our model is still

capable of "imagining" the possible hand trajectory. See "Operate Microwave" and "Wash Coffee Cup" in Figure 5.3 for example results from our model.

Visualization of our method. Finally, we visualize the predicted motor attention, interaction hotspots, and action labels from our model in Figure 5.3. The predicted motor attention almost always attends to the predicted objects and corresponding interaction hotspots. Hence, our model can address challenging cases where next-active objects are ambiguous. These results further show that our proposed motor attention module has the remarkable ability of "imagining" possible hand movements even without the presence of hands in the observed video segments. Another interesting observation is that the predicted distribution of interaction hostpots can be sparse in certain circumstances (e.g., "Open Fridge" or "Take Condiment"). This is because of the stochastic patterns of human-object interaction: There might be multiple valid contact regions for interaction, especially when the future active object has a relatively large scale. This again shows the necessity of the stochastic units in our proposed method.

However, the occlusion and absence of active objects might make the anticipation problem extremely challenging even for humans. The failure cases in Figure 5.3 also suggest that the anticipation model can be biased by on-going action. This is because current FPV datasets (especially EPIC-Kitchens) segment a continuous action into several same atomic actions to ensure all action segments have similar temporal dimension. For instance, A video clip of "cutting onions" for 20 seconds is segmented into 7 or 8 shorter clips all having the same "cutting onions" label. This increases the transition probability of staying in current action state, and thereby biases the model. Therefore, the ability of predicting when exactly the action will end is important for more accurate action prediction model. This task is also related to the action localization problem in the literature [41].

5.4.4 Remarks and Discussion

We must also point out that our method has certain limitations, which point to exciting future research directions. For example, our model requires additional annotations for training, which might bring scalability issues when analyzing other datasets. These dense annotations can indeed be approximated using sparsely annotated frames as discussed in Sec. 4.1. We speculate that more advanced hand tracking and object segmentation models can be explored to generating the pseudo ground truth of motor attention and interaction hotspots. Moreover, our model shares a similar conundrum faced by previous work on anticipation. Our model is likely to fail when future active objects are not observed. See "Close Fridge Drawer"and "Put Coffee Maker" in Figure 5.3. We conjecture that these cases requires incorporating logical reasoning into learning based methods—an active research topic in our community.

5.5 Conclusion

In this chapter, I propose a novel deep model that jointly predicts motor attention, interaction hotspots, and future action labels in egocentric vision. Importantly, I demonstrate that motor attention plays an important role in forecasting human-object interactions. Moreover, I show that characterizing motor attention and interaction hotspots as probabilistic variables can account for the stochastic pattern of human intentional movement. I conduct extensive experiments on two major egocentric video datasets (EGTEA Gaze+ and EPIC-Kitchens) and show that our model design can improve the performance of human-object interaction anticipation by a significant margin. We believe that our model provides a solid step towards the challenging problem of visual anticipation.

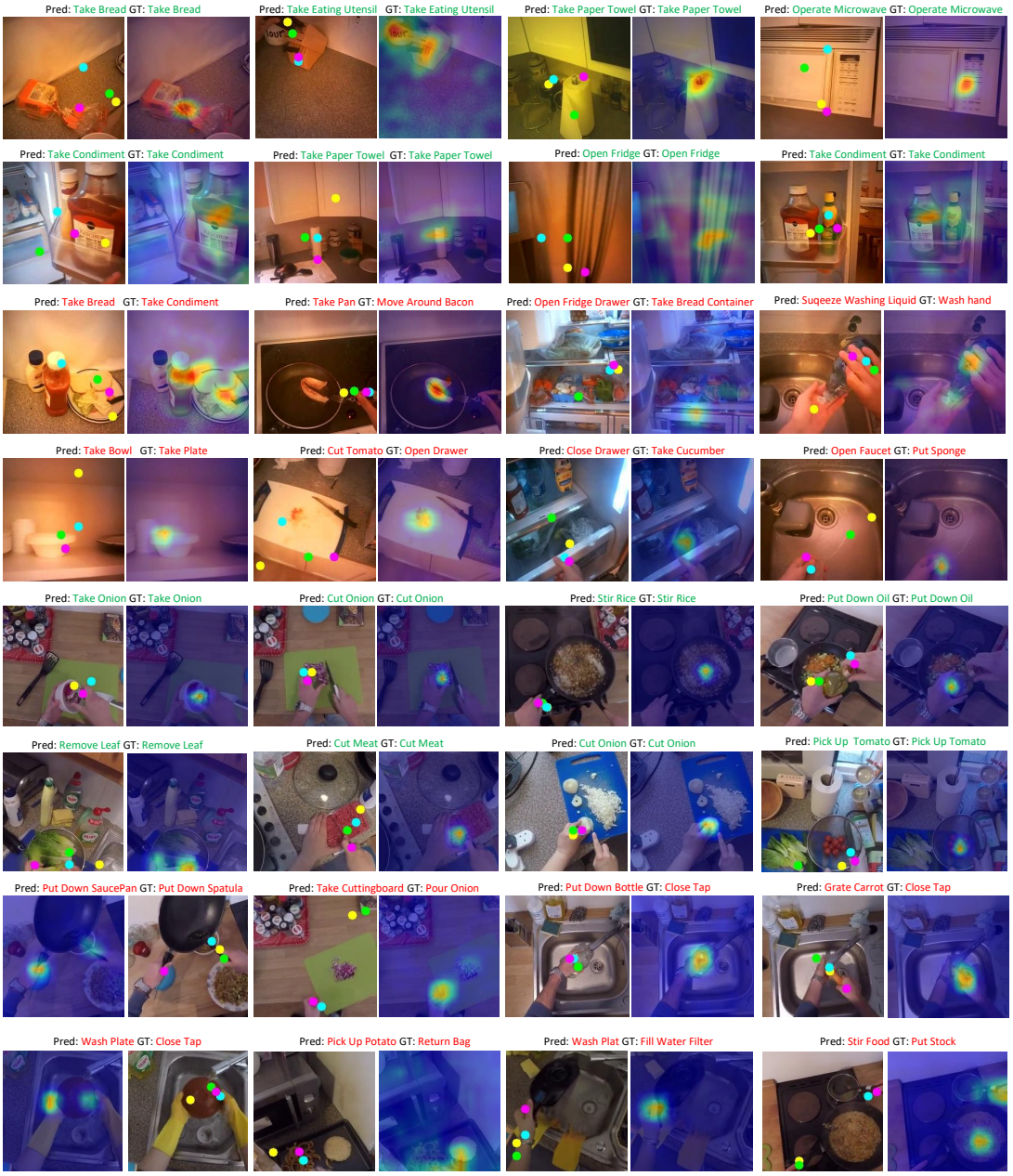

Figure 5.3: **Visualization of motor attention (left image), interaction hotspots (right image), and action labels (captions above the images) on sample frames from EGTEA (first row) and EPIC-Kitchens (second row)**. Both successful (green label) and failure cases (red label) are shown. Future hands position are predicted at every 8 frames and plotted on the last observable frame with the order of yellow, green, cyan, and magenta.

CHAPTER 6

FUTURE HAND SEGMENTATION IN EGOCENTRIC VIDEO

6.1 Introduction

The egocentric vision paradigm provides an ideal vehicle for studying the relationship between visual anticipation and intentional motor behaviors, as head-worn cameras can capture both human visual experience and related sensory-motor signals. In the previous chapter, I demonstrate how such a relationship can be used for egocentric action anticipation. However, the problem of forecasting the detailed shape of hand movements in egocentric video remains unexplored. This is a significant deficit because many everyday motor behaviors cannot be easily categorized into specific action classes and yet play an important role in preparing and executing our routine activities. Such a general prediction capability could enable new applications in AR and robotics, such as monitoring for safety in dangerous environments, or facilitating human-robot collaboration via improved anticipation.

To bridge this gap, in this chapter, I introduce the novel task of forecasting a detailed representation of future hand movements in egocentric video. Specifically, given an egocentric video, we seek to predict the hand masks of future video frames at three time points defined as short-term, middle-term, and long-term future (see Figure 6.1 for a visual illustration of our problem setting). This task is extremely challenging for two reasons: 1) hands are deformable and capable of fast movement, and 2) head and hand motion are entangled in the egocentric video. Addressing these challenges requires the ability to 1) address the inherent uncertainty in anticipating the no-rigid hand movements, and 2) explicitly model the coordination between head and hand [153].

We attack the unique challenges of hand segmentation prediction by introducing a novel deep model – *EgoGAN*. Our model adopts a 3D Fully Convolutional Network (3DFCN) as

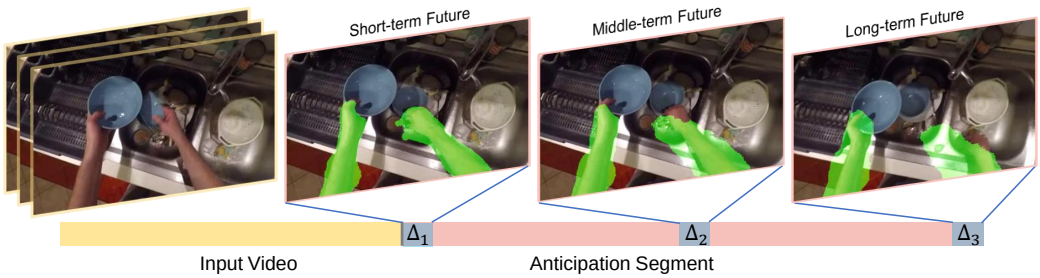

Figure 6.1: **Future hand segmentation task**: Given an input egocentric video, our goal is to predict a time series of future hand masks in the anticipation video segment. Δ_1, Δ_2, and Δ_3 represent the short-term, middle-term, and long-term time points in the anticipation segment, respectively. The entanglement between drastic head motion and non-rigid hand movements poses a significant technical barrier in computer vision. Here, we visualize our results on this challenging task (best viewed in color).

the backbone to learn spatio-temporal video features for pixel-wise visual anticipation. We then utilize a Generative Adversarial Network (GAN) for hand masks anticipation. Instead of using GAN to generate future video frame pixels from egocentric videos as in [33], our key insight is to use the GAN to model an underlying distribution of possible future head motion. The adopted generative adversarial training schema can account for the uncertainty of future hand movements anticipation. In addition, the generated future head motion provides ancillary cues that complement video features for anticipating complex egocentric hand movements. Our end-to-end trainable EgoGAN model uses future hand masks as supervisory signals to train the segmentation network and estimated sparse optical flow maps from head motions (by masking out the hands) to train the Generator and the Discriminator. At inference time, our model predicts a time series of future hand masks based *only* on the egocentric video frame inputs.

To summarize, this chapter has the following contributions:

- I introduce a novel problem of predicting a time series of future hand masks from egocentric videos.
- I propose a novel deep generative model – EgoGAN, that hallucinates future head motions and further predicts future hand masks.

- I present comprehensive experimental results to show the benefits of our method on two benchmark egocentric video datasets: EPIC-Kitchens 55 [147] and EGTEA Gaze+ [57].

This work [154] was a collaboration with Wenqi Jia and was published in ECCV 2022 as a poster paper.

6.2 Method

Given an input egocentric video $x = \{x^1, ..., x^t\}$, where x^t is the video frame indexed by time t, our goal is to predict a time series of future hand masks $h = \{h^{t+\Delta_1}, h^{t+\Delta_2}, h^{t+\Delta_3}\}$. As illustrated in Figure 6.1, we consider hand segmentation as a binary classification problem: the value of $h^i(x, y)$ can be viewed as the probability of spatial position (x, y) being a hand pixel at time step i, where $i \in \{t + \Delta_1, t + \Delta_2, t + \Delta_3\}$. Δ_1, Δ_2, and Δ_3 represent the time steps for short-term, middle-term, and long-term future segmentation, respectively. This three-steps-ahead visual anticipation setting is also used in previous works on future image segmentation [155, 156].

I now present an overview of our *EgoGAN* model in Figure 6.2. We make use of a 3D Fully Convolutional Network (3DFCN) ϕ as the backbone model for future hand segmentation. The 3DFCN is composed of a 3D convolutional encoder ϕ_E and a 3D deconvolutional decoder ϕ_D. We further adopt a Generative Adversarial Network (GAN) for learning future head motions. Specifically, a Generator network (G), composed of 3D convolutional operations, is used to generate future head motion m_g based on the encoded video feature $\phi_E(x)$. A Discriminator Network (D) is trained to distinguish the fake future head motions m_g from real future head motions m_r. Finally, ϕ_D combines m_g and $\phi_E(x)$ for predicting future hand masks. In the following sections, we detail each key component of our model.

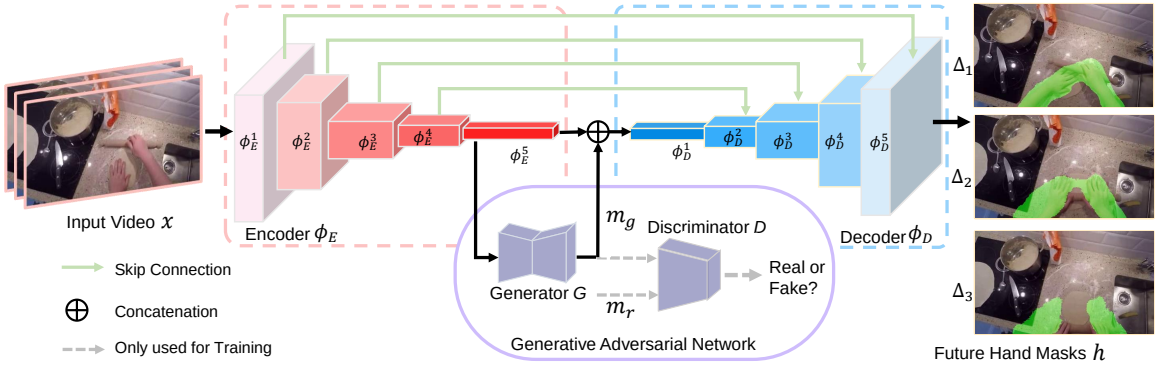

Figure 6.2: **Overview of our proposed EgoGAN model**. Our network takes multiple egocentric video frames as the inputs, and outputs future hand masks at different time steps. It is composed of a *3D Fully Convolutional Network (3DFCN)* and a *Generative Adversarial Network (GAN)*. The Encoder Network ϕ_E in the *3DFCN* extracts video features from the input frames, and is then separated into two branches: (1) encoded feature $\phi_E(x)$ is fed into the *Generator (G)* in for generating fake future head motion m_g, and a *Discriminator (D)* is trained to distinguish the generated future head motion from the real ones; (2) m_g is concatenated to $\phi_E(x)$ and the concatenated tensor are then fed into the Decoder Network ϕ_D in *3DFCN*. Finally, the encoder features are further combined with corresponding decoder features using skip connections for future hand mask prediction.

6.2.1 3D Fully Convolutional Network

We first introduce the 3D Fully Convolutional Network (3DFCN) backbone in our method. We make use of the I3D model [110] as the backbone encoder network ϕ_E for learning spatio-temporal video representations. Following [148, 149], ϕ_E has 5 convolutional blocks, and thereby produces video features at different spatial and temporal resolutions. Following [157], we construct the decoder network ϕ_D symmetric to ϕ_E. Therefore, ϕ_D is also composed of 5 deconvolution layers. We denote the encoder and decoder video features from the ith convolutional block as $\phi_E^i(x)$ and $\phi_D^i(x)$, respectively. (See Fig. Figure 3.2 for the index naming of ϕ_E and ϕ_D.) The features of each decoder layer are combined with the features from the corresponding encoder block with skip connections and are then fed into the next layer. Formally, we have:

$$\phi_D^{i+1}(x) = deconv(\phi_D^i(x) + \phi_E^{6-i}(x)), \tag{6.1}$$

where $i \in \{1, 2, 3, 4\}$. We design our decoder so that ϕ_D^i produces a feature map with the same tensor size as $\phi_E^{6-i}(x)$. The deconvolution operation is implemented with 3D transposed convolution. Note that the last deconvolution layer of ϕ_D produces a tensor with the same size as the input video ($T \times W \times H$). We further apply a 3D convolutional operation with kernel size of $k \times 1 \times 1$ to predict the future hand mask tensor h with size $3 \times W \times H$, where each temporal slice corresponds to the predicted hand masks of the short-term, middle-term, and long-term future video frames.

6.2.2 Generative Adversarial Network

The key to our approach is to use the Generative Adversarial Network (GAN) to hallucinate the future head motions for future hand mask segmentation. Our design choice stems from the observation that head motion causes drastic changes of the active object cues and background scene context captured in the egocentric videos, and this motion is closely related to hand movements. Therefore, we seek to explicitly encode the future head motions cues for hand motion anticipation. Moreover, visual anticipation has intrinsic ambiguity – similar current observations may correspond to different future outcomes. This observation motivates us to use the adversarial training scheme to account for the inherent uncertainty of future representation. In this section, we introduce the egocentric head motion representation. We then describe the design choice and learning objective of the GAN from our method.

Egocentric Head Motion Representation. In the egocentric setting, head motion is implicitly incorporated in the video itself. Thus, we follow [31] to use the sparsely sampled background optical flow to represent the egocentric head motion. As mentioned before, the real future head motion is denoted as m_r, and is only available for training.

Generator Network and Discriminator Network. The generator network (G) takes video feature $\phi_E(x)$ as inputs and generates future head motions $m_g = G(\phi_E(x))$. Following [158, 33, 159, 90], G does not take any noise variables as additional inputs. This is

because the $\phi_E(x)$ is a latent representation that incorporates the noisy signals of visual anticipation. G is composed of multiple 3D convolutional operations and a nonlinearity function, and is trained to produce a realistic m_g that is difficult to distinguish from m_r for an adversarially-trained discriminator network (D). D takes future head motion samples as inputs and determines whether the input sample is real or fake. It is composed of 3D convolutional operations and a sigmoid function for binary classification, and is trained to classify the input sample as either real or generated.

Learning Objective of GAN. We now formally define the objective function of the GAN in our method. The objective function for training the discriminator network is given by:

$$\mathcal{L}_d = \mathcal{L}_{ce}(D(m_r), 1) + \mathcal{L}_{ce}(D(m_g), 0), \tag{6.2}$$

where $\mathcal{L}_{ce}$ is the standard cross-entropy loss for binary classification. The generator loss $\mathcal{L}_g$ can be formulated as:

$$\mathcal{L}_g = \mathcal{L}_{ce}(D(m_g), 1) + \lambda |m_g - m_r|. \tag{6.3}$$

Here, we follow [160] to adopt a traditional L1 distance loss that encourages the generated sample to be visually consistent with the real sample, while λ denotes the weight to balance the two loss terms.

6.2.3 Full Model of EgoGAN

We now summarize the full architecture of our proposed EgoGAN model. The main idea is to explicitly model the underlying distribution of possible future head motion m_g with the GAN, and use m_g as additional cues to facilitate future hand mask segmentation from the video representations of the encoder network. Specifically, the video feature from the last encoder block $\phi_E^5(x)$ and generated future head motions m_g are concatenated and fed

into the first layer of the decoder as inputs. Therefore, we have:

$$\phi_D^1(x) = deconv(\phi_E^5(x) \oplus m_g). \tag{6.4}$$

Hence, the decoder network jointly considers $\phi_E(x)$ and m_g for predicting future hand masks h.

Training and Inference. We adopt the standard adversarial training pipeline in [161], where G and D are trained to play against each other. Therefore, we let the gradients alternatively flow through D, and then G. We then use the binary cross-entropy loss to train the 3DFCN backbone:

$$\mathcal{L}_{seg} = \mathcal{L}_{ce}(\phi_D(\phi_E(x), m_g), \hat{h}), \tag{6.5}$$

where $\hat{h}$ denotes the ground truth of future hand masks. Note that our model does not need the future head motion as additional inputs for making an inference. Instead, our model is capable of generating future head motion and further predicting future hand masks based on only raw video frames. Notably, we freeze the encoder weights during the gradient step on G and D, and freeze the generator weights during the gradient step on the 3DFCN to isolate their training processes from each other.

6.3 Experiments

In this section, we present our experiments and results. We start with a description of the datasets, annotations, and evaluation metrics used in our experiments. We then provide detailed ablation studies to validate our model design, and compare our method to previous state-of-the-art methods on future image segmentation. Furthermore, we provide visualizations and additional discussions of our method.

6.3.1 Dataset and Metrics

Dataset. We make use of two egocentric video benchmark datasets: EPIC-Kitchens 55 [147] and EGTEA Gaze+ [57]. For the EPIC-Kitchens dataset, we set $\delta_{1,2,3} = \{1, 15, 30\}$, which corresponds to a long-term anticipation time of $1.0s$. As for the EGTEA dataset, we set $\Delta_{1,2,3} = \{1, 6, 12\}$, which corresponds to an anticipation time of $0.5s$, because EGTEA has a smaller angle of view in comparison with the EPIC-Kitchens. The same anticipation time setup is also adopted in [129]. To encourage our model to capture the meaningful preparation and planning process of daily actions, we segment the data so that the long-term future frame is chosen right before the beginning of each trimmed action segment annotated in EPIC-Kitchens and EGTEA. We use the train/val split provided by [26] for EPIC-Kitchens 55 and the train/test split1 from EGTEA. We remove the instances where hands are not captured within the anticipation segment, which results in 11,935/2,746 (train/val) samples on EPIC-Kitchens, and 4,042/991 (train/test) samples on EGTEA.

Hand Mask Ground Truth. For the EPIC-Kitchens dataset, we use the domain adaption method introduced in [37] to generate the ground truth hand masks. [37] has empirically verified the quality of generated hand masks. As for the EGTEA dataset, we train a 2D FCN model for frame-level hand segmentation using the provided hand mask annotation. As discussed in [162], the FCN model can generalize well on the entire dataset. We thus use the inference results on the anticipation video frames as the ground truth of future hand masks.

Metrics. As discussed in Chapter 6.2, we consider future hand segmentation as a pixel-wise binary classification problem. Previous future image segmentation works [163, 164] use pixel accuracy and mIoU as evaluation metrics. However, pixel accuracy does not penalize the false-negative prediction of the long-tailed distribution, and mIoU can not properly evaluate the shape of the predicted masks for binary segmentation. Therefore, we follow [101, 129] to report Precision and Recall values together with their corresponding F1 scores.

Table 6.1: **Analysis of variations in our approach**. We conduct detailed ablation studies to validate our model design, and further show the results of variations of our method to demonstrate the benefits of using the GAN for modeling future head motion. *: HeadDir takes future head motions as additional input modalities at inference time, which in fact violates the future anticipation setting (See more discussion in Chapter 6.3.2.). The best results are highlighted with **boldface**.

(a) Experimental Results on EPIC-Kitchens Dataset

Method	EPIC-Kitchens (Precision/ Recall/ F1 Score)		
	short-term	middle-term	long-term
Future Gaze	N/A	N/A	N/A
HeadDir*	70.55/ 71.33/ 70.94	43.15/ 53.66/ 47.83	30.51/ 49.60/ 37.78
3DFCN (w/o GAN, w/o Head)	69.51/ 70.81/ 70.15	42.51/ 51.66/ 46.64	29.88/ 47.46/ 36.67
HeadReg (w/o GAN, w/ Head)	70.46/ 70.25/ 70.36	41.41/ 52.55/ 46.32	29.22/ 48.50/ 36.47
PixelGan (w/ GAN, w/o Head)	69.12/ **71.60**/ 70.34	43.83/ 51.32/ 47.28	30.76/ 47.48/ 37.33
EgoGAN (w/ GAN, w/ Head)	**70.89**/ 71.24/ **71.07**	**43.79/ 53.23/ 48.05**	**31.39/ 48.57/ 38.14**

(b) Experimental Results on EGTEA Gaze+ Dataset

Method	EGTEA (Precision/ Recall/ F1 Score)		
	short-term	middle-term	long-term
Future Gaze	45.17/ 59.94/ 51.51	38.63/ 64.02/ 48.19	35.71/ 63.78/ 45.78
HeadDir*	44.58/ 63.87/ 52.51	41.29/ 60.65/ 49.13	39.36/ 59.02/ 47.23
3DFCN (w/o GAN, w/o Head)	43.62/ **61.69**/ 51.11	40.25/ 58.93/ 47.83	37.83/ 58.32/ 45.89
HeadReg (w/o GAN, w/ Head)	43.54/ 61.03/ 50.82	**41.31**/ 55.24/ 47.27	36.87/ 58.23/ 45.15
PixelGan (w/ GAN, w/o Head)	43.78/ 61.33/ 51.09	38.38/ **63.81**/ 47.93	35.53/ **63.41**/ 45.54
EgoGAN (w/ GAN, w/ Head)	**44.91**/ 61.48/ **51.91**	41.10/ 59.90/ **48.75**	**38.16**/ 59.88/ **46.61**

6.3.2 Model Ablations and Analysis

To validate our model design, we conduct experiments of ablations and variations in our model. Specifically, we investigate how the egocentric head motion cues facilitate future hand segmentation and demonstrate the benefits of using the GAN for modeling future head motion. We also show how modeling future gaze as attentional representation affects the future hand segmentation performance.

Benefits of Encoding Future Head Motions. As a starting point, we compare the model that uses only the 3D Fully Convolutional Network (denoted as *3DFCN*) with the model that directly takes future head motion as an additional input modality (denoted as *HeadDir*). HeadDir shares the same backbone network as 3DFCN, but requires the future head motions for making an inference and therefore violates the future anticipation setting, where the model can not use any information from the anticipation video segment for making an

inference. HeadDir quantifies the performance improvement when the egocentric head motion cues are explicitly encoded into the model in a two-stream structure [42]. The experimental results are summarized in Table 6.1. Compared to 3DFCN, HeadDir achieves a large performance gain on EPIC-Kitchens ($+0.8\%/1.2/1.1\%$ in F1 score for short/middle/long term anticipation), and reaches ($+1.4\%/1.3\%/1.3\%$) on EGTEA.

Our method, on the other hand, outperforms 3DFCN by a large margin on both EPIC-Kitchens($+0.9\%/1.5\%/1.8\%$) and EGTEA ($+0.8\%/0.9\%/0.7\%$). More importantly, our method improves HeadDir by $+0.1\%/0.2\%/0.4\%$ on EPIC-Kitchens. This result suggests that the GAN from our model does not simply learn to predict a future head motion flow map; instead, it models the underlying distribution of possible future head motion and thus improves the future hand anticipation accuracy by addressing the inherent uncertainty of visual forecasting. It is to be observed that our model slightly lags behind HeadDir ($0.6\%/0.4\%/0.6\% \downarrow$) on EGTEA, because EGTEA has fewer samples to train our deep generative model. And we also re-emphasize that our method does not use any additional inputs at inference time as in HeadDir.

The Effect of GAN. To further show the benefits of using the GAN for learning future head motions, we consider a baseline model – *HeadReg*, that uses a regression network to predict future head motions with only L1 distance in Equation 6.3. Note that the regression network is implemented the same way as the generator network from EgoGAN. As shown in Table 6.1, without using adversarial training mechanism in our approach, HeadReg lags behind our model by $0.7\%/1.7\%/1.7\% \downarrow$ and $1.1\%/1.5\%/1.5\% \downarrow$ in F1 score for short/middle/long term anticipation on EPIC-Kitchens and EGTEA, respectively. These results support our claim that the GAN can address the stochastic nature of representation and thereby outperforms HeadReg by a large margin on the future hand segmentation task.

Video Pixel Generation vs. Head Motion Generation. We denote another baseline model that directly uses a GAN for anticipating future hand masks, as *PixelGAN*. This model is composed of the 3DFCN backbone network that generates the future hand masks, and a

discriminator network that classifies whether the given hand masks are real or not. The results are presented in Table 6.1. Importantly, the adversarial training schema in PixelGAN slightly decreases the performance of 3DFCN model on EGTEA, and has minor improvement on EPIC-Kitchens. We speculate that this is because directly using a GAN for predicting future hand masks cannot effectively capture the drastic change of scene context in egocentric video. In contrast, our model uses a GAN to explicitly model the head-hand coordination in the egocentric video thereby is capable of more accurately forecasting egocentric hand masks.

Future Head Motion vs. Future Gaze. Furthermore, we present experimental results on how modeling future gaze fixation affects future hand segmentation. Note that the gaze tracking data is only available for the EGTEA dataset. Specifically, we make use of a GAN to model the probabilistic distribution of future gaze fixation. Instead of concatenating future gaze with encoded video features as in Equation 6.4, we follow [101] to use gaze distribution as a saliency map to select important spatio-temporal video features with element-wise multiplication. As shown in Table 6.1, the resulting future gaze model slightly outperforms the baseline 3DFCN model, yet lags behind our model that uses head motion as the key representation ($0.7\%/0.6\%/0.6\% \downarrow$ in F1 score on EGTEA). Previous work [31] suggested that eye-head-hand coordination is important for egocentric gaze estimation, while our results further show that exploiting the eye-head-hand coordination is also beneficial for pixel-wise egocentric visual anticipation. Moreover, future head motion potentially plays a more important role than future gaze fixation on our fine-grained hand forecasting task.

Results of Generated Future Head Motion. Our model also has the capability of generating future head motions. In Table Table 6.2, we compare our methods with HeadReg – the only baseline model that predicts future head motion. We use the standard endpoint error (EPE) as evaluation metric. On the EPIC-Kitchens dataset, our method outperforms

Table 6.2: **Experimental results on generated future head motion**. We calculate the endpoint error (EPE) between the generated head motion and the ground truth head motion. Our method outperforms HeadReg on the EPIC-Kitchens dataset and works on-par with HeadReg on the EGTEA dataset.

Method	Epic-Kitchens (EPE ↓)	EGTEA (EPE ↓)
HeadReg	10.39	5.27
EgoGAN(Ours)	7.08	5.16

Table 6.3: **Experimental results using different backbone networks**. Our model achieves consistent performance improvement when using different backbone networks. (See more discussion in Section 6.3.2.)

(a) Experimental Results on EPIC-Kitchens Dataset

Method	Backbone	Epic-Kitchens (Precision/ Recall/ F1 Score)		
		short-term	middle-term	long-term
3DFCN	I3DRes50	69.51/ 70.81/ 70.15	42.51/ 51.66/ 46.64	29.88/ 47.46/ 36.67
	I3DRes101	69.48/ 70.96/ 70.21	42.32/ 52.80/ 46.98	29.97/ 48.37/ 37.01
EgoGAN	I3DRes50	70.89/ 71.24/ 71.07	43.79/ 53.23/ 48.05	31.39/ 48.57/ 38.14
	I3DRes101	69.17/ 74.05/ 71.53	44.09/ 53.79/ 48.46	30.79/ 52.60/ 38.85

(b) Experimental Results on EGTEA Gaze+ Dataset

Method	Backbone	EGTEA (Precision/ Recall/ F1 Score)		
		short-term	middle-term	long-term
3DFCN	I3DRes50	43.62/ 61.69/ 51.11	40.25/ 58.93/ 47.83	37.83/ 58.32/ 45.89
	I3DRes101	44.66/ 61.81/ 51.85	40.49/ 59.72/ 48.26	35.70/ 66.18/ 46.38
EgoGAN	I3DRes50	44.91/ 61.48/ 51.91	41.10/ 59.90/ 48.75	38.16/ 59.88/ 46.61
	I3DRes101	45.69/ 60.42/ 52.03	39.40/ 64.27/ 48.85	36.92/ 64.43/ 46.94

HeadReg by a significant margin. The performance improvement of our method is smaller on the EGTEA dataset, due to fewer available training samples. These results suggest that the GAN from our model can generates more realistic future head motion.

Results Using I3D-Res101 Backbones. We further show our method can generalize to different backbone encoder networks. In Table 6.3, we report the future hand segmentation results of both our method and 3DFCN baseline using I3DRes50 and I3DRes101 backbone. Importantly, with 50 more layers, I3D-Res101 backbone, can only improve the model performance by $+0.1\%/0.3\%/0.3\%$ on EPIC-Kitchens and $+0.7\%/0.4\%/0.5\%$ on EGTEA. Our model has a larger performance improvement than switching to dense encoder network. Moreover, the EgoGAN model with I3D-Res101 improves 3DFCN-I3DRes101 by $+0.1\%/0.1\%/0.3\%$ on EGTEA and $+0.5\%/0.4\%/0.7\%$ on EPIC-Kitchens. These results

Table 6.4: **Comparison with previous state-of-the-art methods on future image segmentation**. Our results consistently outperform the second-best results (across all methods) by +1.3% on EPIC-Kitchens and +0.7% on EGTEA in average F1 score. *: We re-implement the model to take raw video frames as inputs as our method (See more discussion in Chapter 6.3.3). The best results are highlighted with **boldface**, and the second-best results are underlined.

(a) Experimental Results on EPIC-Kitchens Dataset

Method	Epic-Kitchens (Precision/ Recall/ F1 Score)		
	short-term	middle-term	long-term
X2X [155]	68.69/ 69.35/ 69.02	40.81/ 50.61/ 45.18	28.14/ 45.76/ 34.85
ConvLSTM [156]	69.02/ 69.44/ 69.22	42.72 /51.78/ 46.82	30.01/ 48.01/ 36.94
FlowTrans [164]	69.38/ 69.70 /69.54	42.90/ 52.02/ 47.02	30.19/ 47.56/ 36.94
EgoGAN (Ours)	**70.89/ 71.24/ 71.07**	**43.79/ 53.23/ 48.05**	**31.39/ 48.57/ 38.14**

(b) Experimental Results on EGTEA Gaze+ Dataset

Method	EGTEA (Precision/ Recall/ F1 Score)		
	short-term	middle-term	long-term
X2X [155]	42.96/ 59.32/ 49.84	38.70/ 59.89/ 47.01	36.55/ 59.67/ 45.33
ConvLSTM [156]	44.55/ 59.43/ 50.93	38.28/ **63.54**/ 47.78	36.58/ 62.04/ 46.03
FlowTrans [164]	44.22/ 61.36/ 51.40	40.38/ 58.62/ 47.82	35.04/ **64.34**/ 45.37
EgoGAN (Ours)	**44.91/ 61.48/ 51.91**	**41.10**/ 59.90/ **48.75**	**38.16**/ 59.88/ **46.61**

further show the robustness of our method. (Note that the performance improvement on EGTEA is relatively small with I3DRes101 backbone, due to the limited training data and dense backbone encoder.)

6.3.3 Comparison to State-of-the-Art Methods

We are the first to address the challenging problem of future hand segmentation from egocentric video. Previous works[1] have considered the related problem of future image segmentation. Therefore, we adapt previous state-of-the-art future image segmentation methods to our problem setting and consider the following strong baselines:

•X2X [155] proposes a recursive method that uses the anticipated mask at time step $t + 1$ as an input to interactively predict the future masks at time step $t + 2$, and $t + 3$.

•FlowTrans [164] jointly predicts the masks and optical flow at time step $t + 1$ and recur-

[1]We note that another branch of prior work addresses the problem of video segmentation [165, 166, 167, 168, 169]. These methods track instances masks over time, and therefore can not be used to address the future segmentation problem where the future video frames are not available as inputs for the tracking model.

sively predicts the future masks with preceding flow and masks.

•ConvLSTM [156] uses a Convolutional LSTM to model the temporal relationships of image features, and use both the sequence of image features and output of the ConvLSTM module for future image segmentation.

It is worth noting that the baseline methods [155, 164, 156] adopt a weaker backbone network than ours. To show that the performance gain of our method does not come from a stronger video feature encoder, we re-implement the above methods with the same I3D-Res50 backbone network as our model. Moreover, both FlowTrans and ConvLSTM assume accurate semantic segmentation of observable video frames is available as inputs, but our model seeks to forecast future hand segmentation using only raw video frames, and thus is a more challenging and practical setting. In addition, accurate semantic segmentation results on egocentric video frames are difficult to obtain, due to the domain gap and lack of training data. Therefore, for a fair comparison, we implement the ConvLSTM and FlowTrans models to take the same input as our method.

The experimental results are summarized in Table 6.4. Among all baseline methods, FlowTrans achieves the best performance for short-term anticipation. However, it is less effective for long-term anticipation, due to the error accumulation of predicted future optical flow. ConvLSTM can better capture the long-term temporal relationship and thereby achieve the best baseline performance for long-term anticipation. Instead of encoding the temporal connection with recursive prediction, we found that the 3D deconvolution is effective for capturing the temporal correlation of anticipation video segments, and in doing so predicts the future hand masks in one shot. More importantly, our method outperforms previous best results (underlined in Table 6.4) by $+1.5\%/1.0\%/1.2\%$ and $+0.5\%/0.9\%/0.6\%$ in F1 score for short/middle/long term hand mask anticipation on EPIC-Kitchens and EGTEA, respectively. These results once again demonstrate the benefits of explicitly modeling future head motion with a GAN.

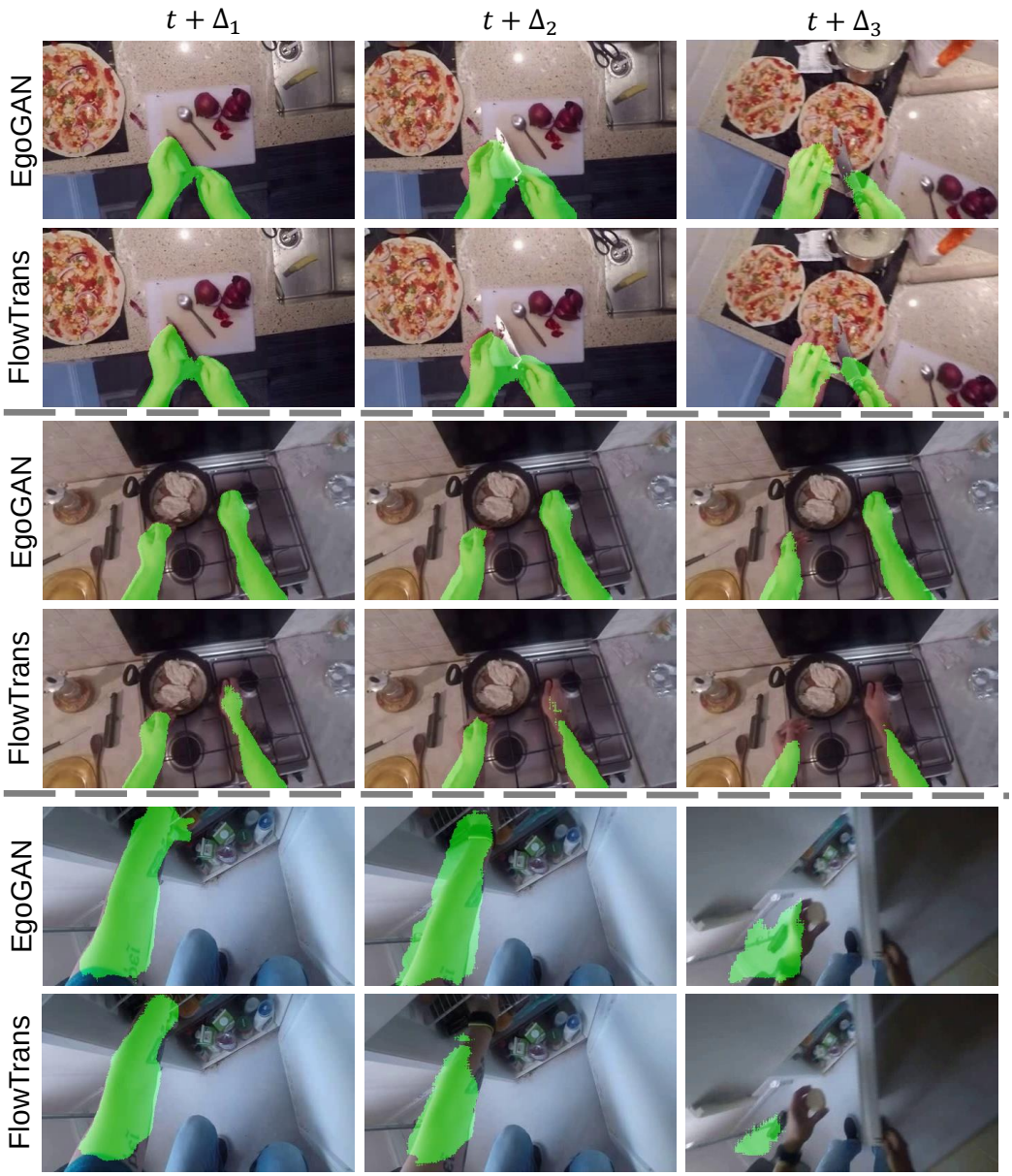

Figure 6.3: **Visualization of our results**. From left to right, each column presents the future hand segmentation results of short-term ($t + \Delta_1$), middle-term ($t + \Delta_2$), and long-term($t + \Delta_3$) time steps from the EPIC-Kitchens dataset. Predictions from our method *EgoGAN* and the best baseline *FlowTrans* are presented in each sample. (See more discussion in Chapter 6.3.4)

6.3.4 Discussion

Visualization. We visualize the results from both our method *EgoGAN* and the best baseline *FlowTrans* on EPIC-Kitchens in Figure 6.3. Even though our proposed problem of

future hand segmentation from egocentric video poses a formidable challenge in computer vision, our method can more accurately predict the hand region of future frames, capture the hand shape and poses compared to FlowTrans. Notably, as the uncertainty increases with the anticipation time, our model may produce blurry predictions, yet can still robustly localize the hand region.

Remarks. To summarize, our quantitative results indicate that future head motion carries important information for future hand movements. We show that explicitly modeling the underlying distribution of possible future hand movements with a GAN enables the model to predict the future hand masks more accurately. Another important takeaway is that our method is more effective than directly using a GAN for predicting future hand masks, as reported in Table 6.1. Furthermore, our visualizations demonstrate that our method can effectively predict future hand masks.

6.4 Conclusion

In this Chapter, we introduce the novel task of predicting a time series of future hand masks from egocentric videos. We present a novel deep generative model EgoGAN to address our proposed problem. The key innovation of our method is to use a GAN module that explicitly models the underlying distribution of possible future head motion for a more accurate prediction of future hand masks. We demonstrate the benefits of our method on two egocentric benchmark datasets, EGTEA Gaze+ and EPIC-Kitchens 55. We believe our work provides an essential step for visual anticipation as well as video pixel generation, and points to new research directions in the egocentric video.

CHAPTER 7
CONCLUSION AND FUTURE WORK

Egocentric vision has emerged as a prevailing research topic due to the advancements in wearable cameras and potential applications in Augmented Reality (AR) and Human-Robot Interaction (HRI). One unique property of egocentric video is the embodiment incorporated in the sequential frames: the visual signals are coupled with the sensory-motor behaviors of the camera wearer. The primary goal of my thesis is to characterize embodied egocentric cues as attention mechanisms for understanding human daily actions, including recognition, anticipation, and localization. In this Chapter, I restate the contribution of my thesis and then discuss several promising future research directions in egocentric vision and artificial social intelligence.

7.1 Conclusion

To summarize, my thesis work makes the following contributions:

- In Chapter 3, I introduce a novel deep model for joint learning of gaze and actions in egocentric video. Through extensive experiments, I demonstrate that our proposed joint model can improve the action recognition performance by a notable margin.
- In Chapter 3, I also show that the attention mechanism provides a novel means for distilling motion information from the optical flow stream to the RGB stream. The experiments suggest that our attention distillation method can facilitate the learning of motion-sensitive video representation.
- My thesis work introduces a novel task of egocentric activity recognition and localization on a 3D map in Chapter 4. By leveraging the attended 3D scene context, the

proposed novel deep model can significantly boost the performance of both action recognition and 3D localization.

- Another contribution of my thesis work is to use the intentional hand movements as a key representation for action anticipation. As discussed in Chapter 5, our method not only achieves state-of-the-art performance on action anticipation, but also has the capability of forecasting future hand movements and interaction hotspots.

- The last piece of my thesis work addresses the novel task of future hand segmentation from egocentric video. I introduce a novel deep generative adversarial network that explicitly models intentional head movements for pixel-wise egocentric visual anticipation, and thereby produces promising results on our proposed task.

7.2 Future Work

In my thesis work, I mainly investigate the problem of understanding human-object interaction using trimmed video sequences and strong supervision. Recent AR glasses (e.g. Aria Glass and Hololens) enable the continuous capture of multiple-sensing modalities from a human-centric perspective. As a result, understanding the untrimmed, continuous, multi-modal egocentric videos is an exciting future research topic. Here, I briefly discuss my future research agenda in this direction.

Open World Egocentric Action Recognition. Most existing literature on action understanding addressed the trimmed video setting, where the meaningful action segment is given for making an inference. However, in real-world scenario, the exact starting and ending time of an action may not be available, especially for the continuous capture with the egocentric camera. Moreover, the video action recognition model does not possess the generalization ability to understand actions that are not yet present in the training set. Therefore, it is critical to develop an online action recognition model that can localize the interesting segments from continuous egocentric video streaming, and understand novel

objects and actions. One future research direction is to explore the open-world egocentric action recognition problem. I plan to develop a never-ending learning video model that can understand the functionality of a novel object by observing how the camera is interacting with the object. This capability may promise a new generation of AR assistants that gradually learns the user's daily routines without any customization.

Towards Computational Theory of Mind. As humans, since infancy, we develop the ability to attribute the mental states to ourselves and others. This remarkable ability is often referred as Theory of Mind (ToM) [170]. Having a Theory of Mind enables us to interpret and anticipate the behavior, emotion, and decision making process of others as well as our own. Can we endow an AI system with the Theory of Mind ability? Egocentric video may serve as an ideal vehicle to study the problem of ToM, as both non-verbal communications (gesture and gaze behavior) and verbal communications (language and audio) are simultaneously captured from a human-centric perspective. Our recent efforts [7] on collecting the egocentric social video dataset already provide a solid step in this direction. As for future work, I plan to explore how the multi-modal transformer architecture can be used for understanding human ToM using this newly captured dataset.

REFERENCES

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *NeurIPS*, 2017.

[2] M. M. Hayhoe, T. McKinney, K. Chajka, and J. B. Pelz, "Predictive eye movements in natural vision," *Experimental brain research*, vol. 217, no. 1, pp. 125–136, 2012.

[3] J. S. Matthis, J. L. Yates, and M. M. Hayhoe, "Gaze and the control of foot placement when walking in natural terrain," *Current Biology*, vol. 28, no. 8, pp. 1224–1233, 2018.

[4] Y. Li, Z. Ye, and J. M. Rehg, "Delving into egocentric actions," in *CVPR*, 2015.

[5] A. Fathi, Y. Li, and J. M. Rehg, "Learning to recognize daily actions using gaze," in *ECCV*, 2012.

[6] M. Ma, H. Fan, and K. M. Kitani, "Going deeper into first-person activity recognition," in *CVPR*, 2016.

[7] K. Grauman, A. Westbury, E. Byrne, Z. Chavis, A. Furnari, R. Girdhar, J. Hamburger, H. Jiang, M. Liu, X. Liu, *et al.*, "Ego4d: Around the world in 3,000 hours of egocentric video," in *CVPR*, 2022.

[8] A. Bandini and J. Zariffa, "Analysis of the hands in egocentric vision: A survey," *IEEE transactions on pattern analysis and machine intelligence*, 2020.

[9] I. Rodin, A. Furnari, D. Mavroeidis, and G. M. Farinella, "Predicting the future from first person (egocentric) vision: A survey," *Computer Vision and Image Understanding*, vol. 211, p. 103 252, 2021.

[10] E. H. Spriggs, F. De La Torre, and M. Hebert, "Temporal segmentation and activity classification from first-person sensing," in *CVPR Workshops*, 2009.

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

[12] A. Fathi, A. Farhadi, and J. M. Rehg, "Understanding egocentric activities," in *ICCV*, 2011.

[13] H. Pirsiavash and D. Ramanan, "Detecting activities of daily living in first-person camera views," in *CVPR*, 2012.

[14] M. S. Ryoo, B. Rothrock, and L. Matthies, “Pooled motion features for first-person videos,” in *CVPR*, 2015.

[15] Y. Poleg, A. Ephrat, S. Peleg, and C. Arora, “Compact CNN for indexing egocentric videos,” in *WACV*, 2016.

[16] S. Sudhakaran, S. Escalera, and O. Lanz, “Lsta: Long short-term attention for egocentric action recognition,” in *CVPR*, 2019.

[17] E. Kazakos, A. Nagrani, A. Zisserman, and D. Damen, “EPIC-Fusion: Audio-visual temporal binding for egocentric action recognition,” in *ICCV*, 2019.

[18] M. Wray, D. Larlus, G. Csurka, and D. Damen, “Fine-grained action retrieval through multiple parts-of-speech embeddings,” in *ICCV*, 2019.

[19] Y. Shen, B. Ni, Z. Li, and N. Zhuang, “Egocentric activity prediction via event modulated attention,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[20] P. Wei, D. Xie, N. Zheng, and S.-C. Zhu, “Inferring human attention by learning latent intentions.,” in *IJCAI*, 2017.

[21] T. Yagi, K. Mangalam, R. Yonetani, and Y. Sato, “Future person localization in first-person videos,” in *CVPR*, 2018.

[22] M. Ryoo, T. J. Fuchs, L. Xia, J. K. Aggarwal, and L. Matthies, “Robot-centric activity prediction from first-person videos: What will they do to me?” In *HRI*, 2015.

[23] B. Soran, A. Farhadi, and L. Shapiro, “Generating notifications for missing actions: Don’t forget to turn the lights off!” In *ICCV*, 2015.

[24] A. Miech, I. Laptev, J. Sivic, H. Wang, L. Torresani, and D. Tran, “Leveraging the present to anticipate the future in videos,” in *CVPRW*, 2019.

[25] A. Furnari, S. Battiato, K. Grauman, and G. M. Farinella, “Next-active-object prediction from egocentric videos,” *VCIP*, 2017.

[26] A. Furnari and G. M. Farinella, “What would you expect? anticipating egocentric actions with rolling-unrolling LSTMs and modality attention.,” in *ICCV*, 2019.

[27] Q. Ke, M. Fritz, and B. Schiele, “Time-conditioned action anticipation in one shot,” in *CVPR*, 2019.

[28] A. Fathi, J. K. Hodgins, and J. M. Rehg, "Social interactions: A first-person perspective," in *CVPR*, 2012.

[29] R. Yonetani, K. M. Kitani, and Y. Sato, "Recognizing micro-actions and reactions from paired egocentric videos," in *CVPR*, 2016.

[30] H. Li, Y. Cai, and W.-S. Zheng, "Deep dual relation modeling for egocentric interaction recognition," in *CVPR*, 2019.

[31] Y. Li, A. Fathi, and J. M. Rehg, "Learning to predict gaze in egocentric video," in *ICCV*, 2013.

[32] E. Dessalene, C. Devaraj, M. Maynord, C. Fermuller, and Y. Aloimonos, "Forecasting action through contact representations from first person video," *TPAMI*, 2021.

[33] M. Zhang, K. Teck Ma, J. Hwee Lim, Q. Zhao, and J. Feng, "Deep future gaze: Gaze anticipation on egocentric videos using adversarial networks," in *CVPR*, 2017.

[34] S. Singh, C. Arora, and C. Jawahar, "First person action recognition using deep learned descriptors," in *CVPR*, 2016.

[35] C. Li and K. M. Kitani, "Model recommendation with virtual probes for egocentric hand detection," in *ICCV*, 2013.

[36] D.-A. Huang, M. Ma, W.-C. Ma, and K. M. Kitani, "How do we use our hands? discovering a diverse set of common grasps," in *CVPR*, 2015.

[37] M. Cai, F. Lu, and Y. Sato, "Generalizing hand segmentation in egocentric videos with uncertainty-guided model adaptation," in *CVPR*, 2020.

[38] J. K. Aggarwal and M. S. Ryoo, "Human activity analysis: A review," *ACM Computing Surveys (CSUR)*, vol. 43, no. 3, pp. 1–43, 2011.

[39] P. Turaga, R. Chellappa, V. S. Subrahmanian, and O. Udrea, "Machine recognition of human activities: A survey," *IEEE Transactions on Circuits and Systems for Video technology*, vol. 18, no. 11, pp. 1473–1488, 2008.

[40] F. Caba Heilbron, V. Escorcia, B. Ghanem, and J. Carlos Niebles, "Activitynet: A large-scale video benchmark for human activity understanding," in *Proceedings of the ieee conference on computer vision and pattern recognition*, 2015, pp. 961–970.

[41] C. Gu, C. Sun, D. A. Ross, C. Vondrick, C. Pantofaru, Y. Li, S. Vijayanarasimhan, G. Toderici, S. Ricco, R. Sukthankar, *et al.*, "Ava: A video dataset of spatio-temporally localized atomic visual actions," in *CVPR*, 2018.

[42] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *NeurIPS*, 2014.

[43] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *ICCV*, 2015.

[44] K. Hara, H. Kataoka, and Y. Satoh, "Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet?" In *CVPR*, 2018.

[45] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *CVPR*, 2017.

[46] H. Bilen, B. Fernando, E. Gavves, and A. Vedaldi, "Action recognition with dynamic image networks," *TPAMI*, 2018.

[47] J. Y.-H. Ng, J. Choi, J. Neumann, and L. S. Davis, "Actionflownet: Learning motion representation for action recognition," in *WACV*, 2018.

[48] L. Fan, W. Huang, S. E. Chuang Gan, B. Gong, and J. Huang, "End-to-end learning of motion representation for video understanding," in *CVPR*, 2018.

[49] J. S. Pérez, E. Meinhardt-Llopis, and G. Facciolo, "TV-L1 optical flow estimation," *IPOL*, 2013.

[50] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A closer look at spatiotemporal convolutions for action recognition," in *CVPR*, 2018.

[51] S. Xie, C. Sun, J. Huang, Z. Tu, and K. Murphy, "Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification," in *ECCV*, 2018.

[52] D. Tran, H. Wang, L. Torresani, and M. Feiszli, "Video classification with channel-separated convolutional networks," in *ICCV*, 2019.

[53] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *ICCV*, 2019.

[54] C. Yang, Y. Xu, J. Shi, B. Dai, and B. Zhou, "Temporal pyramid network for action recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 591–600.

[55] N. Crasto, P. Weinzaepfel, K. Alahari, and C. Schmid, "MARS: Motion-Augmented RGB Stream for Action Recognition," in *CVPR*, 2019.

[56] J. Stroud, D. Ross, C. Sun, J. Deng, and R. Sukthankar, "D3d: Distilled 3d networks for video action recognition," in *WACV*, 2020.

[57] Y. Li, M. Liu, and J. M. Rehg, "In the eye of the beholder: Gaze and actions in first person video," *TPAMI*, 2021.

[58] R. Girdhar, J. Carreira, C. Doersch, and A. Zisserman, "Video action transformer network," in *CVPR*, 2019.

[59] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, "Self-attention generative adversarial networks," in *ICML*, 2019.

[60] V. Mnih, N. Heess, A. Graves, and K. Kavukcuoglu, "Recurrent models of visual attention," in *NeurIPS*, 2014.

[61] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *ICML*, 2015.

[62] S. Sharma, R. Kiros, and R. Salakhutdinov, "Action recognition using visual attention," in *ICLR Workshop*, 2016.

[63] Z. Li, K. Gavrilyuk, E. Gavves, M. Jain, and C. G. Snoek, "Videolstm convolves, attends and flows for action recognition," *CVIU*, 2018.

[64] R. Girdhar and D. Ramanan, "Attentional pooling for action recognition," in *NeurIPS*, 2017.

[65] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, "Residual attention network for image classification," in *CVPR*, 2017.

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

[67] Y. Zhang, M. Hassan, H. Neumann, M. J. Black, and S. Tang, "Generating 3d people in scenes without people," in *CVPR*, 2020.

[68] M. Hassan, V. Choutas, D. Tzionas, and M. J. Black, "Resolving 3D human pose ambiguities with 3D scene constraints," in *ICCV*, 2019.

[69] S. Zhang, Y. Zhang, Q. Ma, M. J. Black, and S. Tang, "PLACE: Proximity learning of articulation and contact in 3D environments," in *3DV*, 2020.

[70] H. Grabner, J. Gall, and L. Van Gool, "What makes a chair a chair?" In *CVPR*, 2011.

[71] V. Delaitre, D. F. Fouhey, I. Laptev, J. Sivic, A. Gupta, and A. A. Efros, "Scene semantics from long-term observation of people," in *ECCV*, 2012.

[72] X. Wang, R. Girdhar, and A. Gupta, "Binge watching: Scaling affordance learning from sitcoms," in *CVPR*, 2017.

[73] S. Thermos, G. T. Papadopoulos, P. Daras, and G. Potamianos, "Deep affordance-grounded sensorimotor object recognition," in *CVPR*, 2017.

[74] N. Rhinehart and K. M. Kitani, "Learning action maps of large environments via first-person vision," in *CVPR*, 2016.

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

[76] C.-Y. Chen and K. Grauman, "Subjects and their objects: Localizing interactees for a person-centric view of importance," *International Journal of Computer Vision*, vol. 126, no. 2-4, pp. 292–313, 2018.

[77] K. Fang, T.-L. Wu, D. Yang, S. Savarese, and J. J. Lim, "Demo2vec: Reasoning object affordances from online videos," in *CVPR*, 2018.

[78] Y. Jiang, H. Koppula, and A. Saxena, "Hallucinated humans as the hidden context for labeling 3d scenes," in *CVPR*, 2013.

[79] Y. Jiang, M. Lim, and A. Saxena, "Learning object arrangements in 3d scenes using human context," in *ICML*, 2012.

[80] M. Savva, A. X. Chang, P. Hanrahan, M. Fisher, and M. Nießner, "Scenegrok: Inferring action maps in 3d environments," *ACM transactions on graphics (TOG)*, vol. 33, no. 6, pp. 1–10, 2014.

[81] A. Gupta, S. Satkin, A. A. Efros, and M. Hebert, "From 3d scene geometry to human workspace," in *CVPR*, 2011.

[82] D. F. Fouhey, V. Delaitre, A. Gupta, A. A. Efros, I. Laptev, and J. Sivic, "People watching: Human actions as a cue for single view geometry," *IJCV*, vol. 110, no. 3, pp. 259–274, 2014.

[83] T. Nagarajan, C. Feichtenhofer, and K. Grauman, "Grounded human-object interaction hotspots from video," in *ICCV*, 2019.

[84] J. Guan, Y. Yuan, K. M. Kitani, and N. Rhinehart, "Generative hybrid representations for activity forecasting with no-regret learning," in *CVPR*, 2020.

[85] N. Rhinehart and K. M. Kitani, "First-person activity forecasting with online inverse reinforcement learning," in *ICCV*, 2017.

[86] T. Nagarajan, Y. Li, C. Feichtenhofer, and K. Grauman, "Ego-topo: Environment affordances from egocentric video," in *CVPR*, 2020.

[87] K. Fragkiadaki, S. Levine, P. Felsen, and J. Malik, "Recurrent network models for human dynamics," in *ICCV*, 2015.

[88] L.-Y. Gui, Y.-X. Wang, X. Liang, and J. M. Moura, "Adversarial geometry-aware human motion prediction," in *ECCV*, 2018.

[89] J. Walker, C. Doersch, A. Gupta, and M. Hebert, "An uncertain future: Forecasting from static images using variational autoencoders," in *ECCV*, 2016.

[90] J. Walker, K. Marino, A. Gupta, and M. Hebert, "The pose knows: Video forecasting by generating pose futures," in *ICCV*, 2017.

[91] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social gan: Socially acceptable trajectories with generative adversarial networks," in *CVPR*, 2018.

[92] Y. Zhang, M. Hassan, H. Neumann, M. J. Black, and S. Tang, "Generating 3d people in scenes without people," in *CVPR*, 2020.

[93] Y. Zhang, M. J. Black, and S. Tang, "We are more than our joints: Predicting how 3d bodies move," in *CVPR*, 2021.

[94] S. Tulyakov, M.-Y. Liu, X. Yang, and J. Kautz, "Mocogan: Decomposing motion and content for video generation," in *CVPR*, 2018.

[95] J. M. Henderson, "Human gaze control during real-world scene perception," *Trends in cognitive sciences*, vol. 7, no. 11, pp. 498–504, 2003.

[96] B. Bridgeman, D. Hendry, and L. Stark, "Failure to detect displacement of the visual world during saccadic eye movements," *Vision research*, vol. 15, no. 6, pp. 719–722, 1975.

[97] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *ICCV*, 2017.

[98] K. Hara, H. Kataoka, and Y. Satoh, "Can spatiotemporal 3D CNNs retrace the history of 2D CNNs and ImageNet," in *CVPR*, 2018.

[99] S. Sun, Z. Kuang, L. Sheng, W. Ouyang, and W. Zhang, "Optical flow guided feature: A fast and robust motion representation for video action recognition," in *CVPR*, 2018.

[100] B. Jiang, M. Wang, W. Gan, W. Wu, and J. Yan, "Stm: Spatiotemporal and motion encoding for action recognition," in *ICCV*, 2019.

[101] Y. Li, M. Liu, and J. M. Rehg, "In the eye of beholder: Joint learning of gaze and actions in first person video," in *ECCV*, 2018.

[102] M. Liu, X. Chen, Y. Zhang, Y. Li, and J. M. Rehg, "Attention distillation for learning video representations," in *BMVC*, 2020.

[103] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *ICLR*, 2014.

[104] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," in *NIPS*, 2015.

[105] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," in *ICLR*, 2017.

[106] C. J. Maddison, A. Mnih, and Y. W. Teh, "The concrete distribution: A continuous relaxation of discrete random variables," in *ICLR*, 2017.

[107] S. Liu, E. Johns, and A. J. Davison, "End-to-end multi-task learning with attention," in *CVPR*, 2019.

[108] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional two-stream network fusion for video action recognition," in *CVPR*, 2016.

[109] G. A. Sigurdsson, G. Varol, X. Wang, A. Farhadi, I. Laptev, and A. Gupta, "Hollywood in homes: Crowdsourcing data collection for activity understanding," in *ECCV*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds., Springer International Publishing, 2016, pp. 842–856.

[110] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *CVPR*, 2018.

[111] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool, "Temporal segment networks: Towards good practices for deep action recognition," in *ECCV*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds., Springer International Publishing, 2016, pp. 20–36.

[112] S. Liu, E. Johns, and A. J. Davison, "End-to-end multi-task learning with attention," *arXiv preprint arXiv:1803.10704*, 2018.

[113] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Action recognition by dense trajectories," in *CVPR*, 2011.

[114] K. Soomro, A. Roshan Zamir, and M. Shah, "UCF101: A dataset of 101 human actions classes from videos in the wild," in *CRCV-TR-12-01*, 2012.

[115] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "HMDB: A large video database for human motion recognition," in *ICCV*, 2011.

[116] S. Zagoruyko and N. Komodakis, "Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer," in *ICLR*, 2017.

[117] B. Zhou, A. Andonian, and A. Torralba, "Temporal relational reasoning in videos," in *ECCV*, 2018.

[118] F. Mahdisoltani, G. Berger, W. Gharbieh, D. Fleet, and R. Memisevic, "Fine-grained video classification and captioning," *arXiv preprint arXiv:1804.09235*, 2018.

[119] J. Wang, A. Cherian, F. Porikli, and S. Gould, "Video representation learning using discriminative pooling," in *CVPR*, 2018.

[120] A. Piergiovanni and M. S. Ryoo, "Representation flow for action recognition," in *CVPR*, 2019.

[121] V. Choutas, P. Weinzaepfel, J. Revaud, and C. Schmid, "Potion: Pose motion representation for action recognition," in *CVPR*, 2018.

[122] H. Idrees, A. R. Zamir, Y.-G. Jiang, A. Gorban, I. Laptev, R. Sukthankar, and M. Shah, "The THUMOS challenge on action recognition for videos "in the wild"," *CVIU*, 2017.

[123] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Is object localization for free?-weakly-supervised learning with convolutional neural networks," in *CVPR*, 2015.

[124] Q. Hou, M.-M. Cheng, X. Hu, A. Borji, Z. Tu, and P. Torr, "Deeply supervised salient object detection with short connections," in *CVPR*, 2017.

[125] M. Z. Sulaiman, M. N. A. Aziz, M. H. A. Bakar, N. A. Halili, and M. A. Azuddin, "Matterport: Virtual tour as a new marketing approach in real estate business during pandemic covid-19," in *Proceedings of the International Conference of Innovation in Media and Visual Design (IMDES 2020). Atlantis Press*, 2020, pp. 221–226.

[126] S. Karthika, P. Praveena, and M. GokilaMani, "Hololens," *International Journal of Computer Science and Mobile Computing*, vol. 6, no. 2, pp. 41–50, 2017.

[127] M. Liu, L. Ma, K. Somasundaram, Y. Li, K. Grauman, J. M. Rehg, and C. Li, "Egocentric activity recognition and localization on a 3d map," *arXiv preprint arXiv:2105.09544*, 2021.

[128] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "Orb-slam: A versatile and accurate monocular slam system," *IEEE transactions on robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[129] M. Liu, S. Tang, Y. Li, and J. M. Rehg, "Forecasting human-object interaction: Joint prediction of motor attention and actions in first person video," in *ECCV*, 2020.

[130] D. Damen, H. Doughty, G. Farinella, S. Fidler, A. Furnari, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price, *et al.*, "The epic-kitchens dataset: Collection, challenges and baselines," *IEEE Computer Architecture Letters*, no. 01, pp. 1–1, 2020.

[131] J. Straub, T. Whelan, L. Ma, Y. Chen, E. Wijmans, S. Green, J. J. Engel, R. Mur-Artal, C. Ren, S. Verma, A. Clarkson, M. Yan, B. Budge, Y. Yan, X. Pan, J. Yon, Y. Zou, K. Leon, N. Carter, J. Briales, T. Gillingham, E. Mueggler, L. Pesqueira, M. Savva, D. Batra, H. M. Strasdat, R. D. Nardi, M. Goesele, S. Lovegrove, and R. Newcombe, "The Replica dataset: A digital replica of indoor spaces," *arXiv preprint arXiv:1906.05797*, 2019.

[132] J.-M. Frahm, P. Fite-Georgel, D. Gallup, T. Johnson, R. Raguram, C. Wu, Y.-H. Jen, E. Dunn, B. Clipp, S. Lazebnik, *et al.*, "Building rome on a cloudless day," in *ECCV*, 2010.

[133] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *CVPR*, 2016.

[134] T. Sattler, B. Leibe, and L. Kobbelt, "Improving image-based localization by active correspondence search," in *ECCV*, 2012.

[135] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, *Detectron2*, https://github.com/facebookresearch/detectron2, 2019.

[136] Y. Zhou and O. Tuzel, "Voxelnet: End-to-end learning for point cloud based 3d object detection," in *CVPR*, 2018.

[137] W. James, F. Burkhardt, F. Bowers, and I. K. Skrupskelis, *The principles of psychology*, 2. Macmillan London, 1890, vol. 1.

[138] G. Di Pellegrino, L. Fadiga, L. Fogassi, V. Gallese, and G. Rizzolatti, "Understanding motor events: A neurophysiological study," *Experimental brain research*, 1992.

[139] R. Hari, N. Forss, S. Avikainen, E. Kirveskari, S. Salenius, and G. Rizzolatti, "Activation of human primary motor cortex during action observation: A neuromagnetic study," *Proceedings of the National Academy of Sciences*, 1998.

[140] S. M. Aglioti, P. Cesari, M. Romani, and C. Urgesi, "Action anticipation and motor resonance in elite basketball players," *Nature neuroscience*, 2008.

[141] M. Rushworth, H. Johansen-Berg, S. M. Göbel, and J. Devlin, "The left parietal and premotor cortices: Motor attention and selection," *Neuroimage*, vol. 20, S89–S100, 2003.

[142] C. Vondrick, H. Pirsiavash, and A. Torralba, "Anticipating visual representations from unlabeled video," in *CVPR*, 2016.

[143] P. Felsen, P. Agrawal, and J. Malik, "What will happen next? forecasting player moves in sports videos," in *ICCV*, 2017.

[144] J. Gao, Z. Yang, and R. Nevatia, "Red: Reinforced encoder-decoder networks for action anticipation," in *BMVC*, 2017.

[145] H. Kataoka, Y. Miyashita, M. Hayashi, K. Iwata, and Y. Satoh, "Recognition of transitional action for short-term action prediction using discriminative temporal cnn feature.," in *BMVC*, 2016.

[146] E. Aksan, M. Kaufmann, and O. Hilliges, "Structured prediction helps 3d human motion modelling," in *ICCV*, 2019.

[147] 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 *ECCV*, 2018.

[148] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *ICLR*, 2015.

[149] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.

[150] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social LSTM: Human trajectory prediction in crowded spaces," in *CVPR*, 2016.

[151] A. Furnari, S. Battiato, and G. Maria Farinella, "Leveraging uncertainty to rethink loss functions and evaluation measures for egocentric action anticipation," in *ECCV Workshops*, L. Leal-Taixé and S. Roth, Eds., Springer International Publishing, 2018, pp. 389–405.

[152] Y. Huang, M. Cai, Z. Li, and Y. Sato, "Predicting gaze in egocentric video by learning task-dependent attention transition," in *ECCV*, 2018.

[153] J. Pelz, M. Hayhoe, and R. Loeber, "The coordination of eye, head, and hand movements in a natural task," *Experimental brain research*, vol. 139, no. 3, pp. 266–277, 2001.

[154] W. Jia, M. Liu, and J. M. Rehg, "Generative adversarial network for future hand segmentation from egocentric video," *arXiv preprint arXiv:2203.11305*, 2022.

[155] P. Luc, N. Neverova, C. Couprie, J. Verbeek, and Y. LeCun, "Predicting deeper into the future of semantic segmentation," in *ICCV*, 2017.

[156] M. Rochan *et al.*, "Future semantic segmentation with convolutional lstm," in *BMVC*, 2018.

[157] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *CVPR*, 2015.

[158] C. Vondrick, H. Pirsiavash, and A. Torralba, "Generating videos with scene dynamics," in *NeurIPS*, 2016.

[159] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *CVPR*, 2017.

[160] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *CVPR*, 2016.

[161] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *NeurIPS*, 2014.

[162] Y. Li, "Learning embodied models of actions from first person video," Ph.D. dissertation, Georgia Institute of Technology, 2017.

[163] H.-k. Chiu, E. Adeli, and J. C. Niebles, "Segmenting the future," *ICRA-L*, 2020.

[164] X. Jin, H. Xiao, X. Shen, J. Yang, Z. Lin, Y. Chen, Z. Jie, J. Feng, and S. Yan, "Predicting scene parsing and motion dynamics in the future," in *NeurIPS*, 2017.

[165] L. Yang, Y. Fan, and N. Xu, "Video instance segmentation," in *ICCV*, 2019.

[166] Y.-H. Tsai, M.-H. Yang, and M. J. Black, "Video segmentation via object flow," in *CVPR*, 2016.

[167] Y.-S. Xu, T.-J. Fu, H.-K. Yang, and C.-Y. Lee, "Dynamic video segmentation network," in *CVPR*, 2018.

[168] S. Chandra, C. Couprie, and I. Kokkinos, "Deep spatio-temporal random fields for efficient video segmentation," in *CVPR*, 2018.

[169] D. Nilsson and C. Sminchisescu, "Semantic video segmentation by gated recurrent flow propagation," in *CVPR*, 2018.

[170] D. Premack and G. Woodruff, "Does the chimpanzee have a theory of mind?" *Behavioral and brain sciences*, vol. 1, no. 4, pp. 515–526, 1978.