INDRAPRASTHA INSTITUTE *of*
INFORMATION TECHNOLOGY **DELHI**

Analysing Long Egocentric Videos

Submitted in fulfillment of the requirements for the degree of Doctor of Philosophy in Department of Computer Science and Engineering

PRAVIN NAGAR

pravinn@iiitd.ac.in

(PhD15016)

Indraprastha Institute of Information Technology

Supervisor:
Prof. Chetan Arora (IIT Delhi)

December 19, 2022

Certificate

This is to certify that the thesis titled **Analysing Long Egocentric Videos** being submitted by **Pravin Nagar** to the Indraprastha Institute of Information Technology-Delhi, for the award of the degree of **Doctor of Philosophy**, is an original research work carried out by him under my supervision. In my opinion, the thesis has reached the standards fulfilling the requirements of the regulations relating to the degree.

The works contained in this thesis have not been submitted in part or full to any other university or institute to award any degree/diploma. However, the video summarization work was done in collaboration with Anuj Rathore (IIIT Hyderabad) and resulted in two publications published in ACMMM and PAMI. This thesis includes the work titled "Generating Personalized Summaries of Day Long Egocentric Video" published in PAMI.

November, 2021

Prof. Chetan Arora

Adjunct Faculty,

Department of Computer Science,

Indraprastha Institute of Information Technology Delhi,

Okhla Industrial Estate, Phase III, New Delhi, Delhi 110020.

"Take up one idea. Make that one idea your life — think of it, dream of it, live on that idea. Let the brain, muscles, nerves, every part of your body, be full of that idea, and just leave every other idea alone. This is the way to success."

– Swami Vivekananda

Acknowledgements

I would like to express my heartiest gratitude to my supervisor, Prof. Chetan Arora, for his continuous support, guidance, and faith in me. I am immensely thankful for his patience and motivation in the tight spots of my journey. I am very grateful to him for sharing his immense knowledge and envision to shape my dissertation. Our long conversations encompassing technical and/or non-technical discussions have greatly inspired me and helped hone my research skills.

I would like to express my sincere gratitude to Prof. C.V. Jawahar (IIIT Hyderabad) for his collaboration that enriched my professional and personal development. I would also like to acknowledge Prof. Anubha Gupta for her guidance in the early days of my Ph.D. I also express my gratitude to the early-stage researchers who collaborated with me. They not only made conducting research fun and exciting but also inspired me in many ways: Mansi Khemka, Anuj Rathore, Sagar Verma, Divam Gupta, and Pulkit Kumar.

I would like to thank my Ph.D. monitoring committee members, Dr. A. V. Subramanyam and Dr. Saket Anand, for their encouragement and critical comments throughout my Ph.D. journey. I would also like to thank Dr. Parag Singla for his valuable feedback on my Ph.D. comprehensive exam.

I am also grateful to Visvesvaraya Ph.D. fellowship from the Government of India and DST, Government of India, under project id T-138, who funded this research. This work not have been possible without their financial support.

Special gratitude goes to my parents, Mrs. Laxmi Nagar and Mr. M. S. Nagar, for nurturing me into the person I am today and for the moral and emotional support all the way through. Special thanks to my beloved younger brother Rahul Nagar for supporting my journey in every possible way and taking care of my parents in my absence.

Like every Ph.D. student, my journey was also full of ups and downs. I am very thankful to all my Ph.D. colleagues with whom, at times, I might have disagreements, but in the end, without them, this roller coaster journey would have been challenging: Anupriya, Dhriti, Ishant, Sharat, Shubham, Surabhi, and Ridhi.

Of all, the last two years of my Ph.D. journey were the most difficult. My mental health was severely affected during sudden and necessary lockdowns imposed during the Covid-19 pandemic. Each variant was surprisingly different from the earlier one and raised a unique set of challenges. I was worried about my family staying 550 miles away from me, especially my father, a frontline worker. However, I would like to thank IIIT-D management for their policies, including allowing Ph.D. researchers to stay on the campus. This lockdown journey was impossible without all my friends: Sharat and Ridhi

(who stayed on campus with me), and Anupriya, Shubham, and Dhriti (who went back to their homes). I enjoyed the early morning lockdown workout with Sharat and Ridhi. I will never forget the long group calls (motivating each other) and the online games we played to boost our morale. The best part was my lockdown birthday having three beautiful homemade cakes by Anupriya, Ridhi, and Sharat! I am fortunate to be a part of this beautiful family, including the foodies and a variety of 'chefs.' The time during differing lockdowns flew by quickly as I just needed to focus on two things: prepare for my next meeting and the next new dish to prepare. Due to the surge in covid cases, it was challenging to focus on the research. However, I was so scared during the second wave when my whole family (including two maternal uncles and an aunt) contracted the delta variant, and I lost both maternal uncles. This was the most challenging time of my life, and I am very thankful for the support of my advisor and my friends during this time. Special thanks to Arshad, for his support to my family during the second wave. The one positive side of this pandemic was that it united my family, and I got a chance to spend a lot of time with my family during the lockdowns.

I also thank Ph.D. administrative Priti Patel and IT staff Bhawani for helping with all the administrative queries and IT-related queries, respectively. Last but not least, IIIT-Delhi for the beautiful campus to nurture me as a researcher.

Pravin Nagar

Publications

1. **Pravin Nagar**, and Chetan Arora, Recovering Activity Patterns from Weeks-long Lifelogs, Communicated to CVPR 2023.

2. **Pravin Nagar**, Anuj Rathore, C. V. Jawahar, and Chetan Arora, Generating Personalized Summaries of Day Long Egocentric Videos, in IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 2021.

3. **Pravin Nagar**, Mansi Khemka, and Chetan Arora, Concept Drift Detection for Multivariate Data Streams and Temporal Segmentation of Daylong Egocentric Videos, in Proceedings of the 28th ACM International Conference on Multimedia (ACMMM), 2020.

4. Anuj Rathore*, **Pravin Nagar***, Chetan Arora, and C. V. Jawahar, Generating 1 Minute Summaries of Day Long Egocentric Videos, in Proceedings of the 27th ACM International Conference on Multimedia (ACMMM), 2019.
(* denotes equally contribution)

5. Sagar Verma, **Pravin Nagar**, Divam Gupta, and Chetan Arora, Making third person techniques recognize first-person actions in egocentric videos, in 25th IEEE International Conference on Image Processing (ICIP), 2018.

6. Pulkit Kumar, **Pravin Nagar**, Chetan Arora, and Anubha Gupta, U-SegNet: fully convolutional neural network based automated brain tissue segmentation tool, in 25th IEEE International Conference on Image Processing (ICIP), 2018.

Abstract

Egocentric videos are recorded in a hands-free, always-on, under enhanced privacy-sensitive scenario and are often collected from day to weeks. For efficient consumption, such videos require robust video analysis techniques that can deal with *extremely long* sequences in an *unsupervised* setting. This dissertation explores a novel research area by developing video analysis tasks for extremely long and sequential data (ranging from a day to weeks long) in a self-supervised/unsupervised setting. In this dissertation, we address the three key video analysis problems, namely temporal segmentation, summarization, and recovering activity patterns, specifically designed to deal with the issues of scalability, privacy, and unlabeled data.

There are a plethora of works in the literature for third person video analysis. However, third person videos are often recorded from point-and-shoot cameras, thus generating small video samples (up to a few minutes). In this dissertation, we work on Disney (up to 8 hrs video sequence), UT Egocentric (UTE) (up to 5 hrs), and EgoRoutine (up to 20 days of photo-stream lifelogs) datasets that are recorded in a real-life setting. Therefore, third person video analysis techniques do not typically scale for long sequences. For example, the simplest task of temporal segmentation becomes challenging for extremely long sequence data as the length of events ranges from a few seconds to hours long. Similarly, for video summarization, we usually consider the whole video sequence to select the appropriate frames/sub-shots for generating a compact yet comprehensive summary. In activity pattern recovery, we need to model the underlying distribution of activity patterns for the whole data (weeks long lifelog), and the task becomes cumbersome when the distributions are highly skewed. In all these instances, the complexity of the task increases multifold and requires a different level of comprehension for modeling the extremely long video sequences. We further demonstrate that state-of-the-art (`SOTA`) approaches based on Recurrent Neural Networks (RNNs), Long Short-Term Memory Networks (LSTMs), Graph Convolutional Networks (GCNs), or Transformers networks fail miserably to handle massively long sequences. This dissertation proposes scalable solutions to analyze extremely long egocentric videos, typically ranging from a day to weeks.

The long and unconstrained nature of egocentric videos makes it imperative to use temporal segmentation as an important pre-processing step for many higher-level video analysis tasks. In the first work, we present a novel unsupervised temporal segmentation technique especially suited for extremely long egocentric videos. We formulate the problem as detecting concept drift in a time-varying, non i.i.d. sequence of frames. Statistically bounded thresholds are calculated to detect concept drift between two temporally adjacent multivariate data segments with different underlying distributions while

establishing guarantees on false positives.

The egocentric videos are extremely long and highly redundant in nature, and these videos are difficult to watch from beginning to end. Hence, require summarization tools for their efficient consumption. The second work presents a novel unsupervised deep reinforcement learning framework to generate video summaries from day long egocentric videos. We also incorporate user choices using interactive feedback for including or excluding a particular type of content in the generated summaries.

Lifelogging applications for egocentric videos require analyzing a huge volume of data often captured over weeks to months for a particular subject and contain long-term dependencies. High-level video analysis tasks over lifelogs include recognizing daily living activity (ADL), routine discovery, event detection, anomaly detection, etc. We observe that the `Transformer`-based `SOTA` architectures still fail for extremely long video sequences. Our analysis reveals that the key ingredient missing is the inability of the architecture to exploit strong spatio-temporal visual cues inherent in video data. To capture such cues within a transformer architecture, we propose a novel architecture named **S**emantic **A**ttention **T**rans**Former** (`SATFormer`), which factorizes the self-attention matrix into a semantically meaningful subspace. We use `SATFormer` within a novel self-supervised training pipeline developed specifically for the task of recovering activity patterns in extremely long (weeks-long) egocentric lifelogs. In the proposed pipeline, we alternatively learn feature embedding from the proposed `SATFormer` using the pseudo-label assigned to each frame and learn the pseudo-labels from the clustering done using feature embedding from `SATFormer`.

Overall, this dissertation is a significant feat addressing the broader issues of *scalability*, *privacy*, and *unlabeled data* and establishing `SOTA` performance for the respective tasks. The proposed works are pioneers in handling massively long (up to 60k time steps) sequence video data in an unsupervised setting.

Table of Contents

[1]This work was done in collaboration with Anuj Rathore (IIIT Hyderabad) and resulted in two publications published in ACMMM and PAMI. This chapter includes the work titled "Generating Personalized Summaries of Day Long Egocentric Video" published in PAMI.

*

List of Tables

Table **Page**

List of Figures

Figure **Page**

Chapter 1

Introduction

1.1 Egocentric Videos

Egocentric videos or first-person videos (FPV) are captured by wearable devices and approximate the visual field of the camera wearer. Consequently, these videos consider the camera wearer as a central reference point and provide the unique perspective of engagement of the wearer to the realistic environment. Fig. 1.1 depicts the comparison between egocentric and traditional videos or third-person videos (TPV).

(a)

(b)

Figure 1.1: Figures (a) and (b) compare a frame sequence generated by a small video snippet of egocentric and third-person videos, respectively. The figure shows that both the subjects perform similar activities (interacting with the dishes) in the kitchen; however, the visuals differ considerably.

The wearable devices used to capture such videos are typically worn on the head or chest of the wearer. Fig. 1.2 depicts a few popular wearable cameras.

1.2 First-person vs. Third-person Videos

Egocentric videos are often captured in a hands-free, always-on manner leading to extremely long and highly redundant video sequences comprising a large variety of un-

Figure 1.2: The figure demonstrates a few wearable devices with their installation on the wearer. GoPro, Pivothead, and SenseCam are head/hat-mounted, glass-mounted, and chest-mounted devices, respectively.

constrained environments compared to their third-person counterpart. By virtue of the specialized placement of the camera, these videos are shaky and lack subject pose information (refer Fig. 1.1). Furthermore, these videos are often captured in enhanced privacy constraints, ruling out the creation of large annotated datasets, and supervised models. Due to these unique characteristics, the third-person techniques do not scale for first-person videos. Table 1.1 compares the two modalities.

	Annotations	Extremely Long	Redundant	Subject Pose Information	Shaky	Privacy Constraints
FPV	✗	✓	✓	✗	✓	✓
TPV	✓	✓	✗	✓	✗	✗

Table 1.1: Comparison between first-person videos (FPV) and third-person videos (TPV).

1.3 Egocentric Vision

Egocentric vision (or first-person vision) is a subfield of computer vision that analyzes egocentric images and videos. Early research in egocentric vision focused on health behavior analysis such as measuring sedentary behavior and nutrition-related behaviors [43]. After that, the computer vision community delved deeper into egocentric vision

due to rapid advancement in wearable devices and the increasing number of potential applications [17]. In 2012, Kanade and Hebert [78] developed the first egocentric vision system to understanding the user's environment and activities. This system focused on the following key characteristics of the egocentric videos: a localization module that estimates the surrounding (match the current image with a large database), a recognition module able to identify important objects and people, and an activity recognition module that recognizes the current activity of the user. Inspired by this work, many egocentric video analysis works focused on hand-related action/activity recognition and social interaction analysis. Wearable devices are typically used in an 'always on manner' that leads to abundant data. Temporal segmentation [33, 124] and summarization [90] are the key problems addressed to handle such long sequential data. Furthermore, activity forecasting, routine discovery, and preserving privacy in the egocentric video are also explored extensively. More recently, egocentric videos have been used to understand human and animal cognition, human-human and human-robot interaction, and augmented reality interfaces.

1.4 Motivation

Due to rapid technological advancements in the last decade, we are witnessing a widespread creation of multimedia data, including image, text, audio, and video. For example, statistics (of the year 2018) show that Instagram users have uploaded over 20 billion photos, Twitter users sent approximately 500 million tweets every day, and YouTube users uploaded over 300 hrs of videos every minute (i.e., more than 8 trillion images per year assuming the frame rate is 15 fps) [72, 128]. It is evident that video constitutes the major share of available multimedia data out of all the modalities. The computer vision community seeks to provide efficient and effective video analysis frameworks to understand, index, and retrieve such gigantic data to handle real-world challenges.

Recently, video data generated by point and shoot cameras have increased exponentially due to mobile technology, low-cost storage, and social media platforms. These videos are triggered by user interest and are typically very short (ranges up to a few minutes). However, three categories of videos are extremely long and constitute the major share of available video data, namely surveillance video, sport videos/movies, and lifelogs.

Furthermore, the surveillance videos contribute the majority among all kinds of long videos captured. In video surveillance, the camera is fixed to a specific location, and the signal is transmitted to a limited set of monitors. Due to always-on recording, these videos are very long and boring. Nowadays, computer vision techniques are extensively used for automated behavioral analysis of the huge volume of data generated from surveillance videos. In behavioral analysis, the abnormal behavior of people, vehicles, machines, and the environment is identified by observing the data collected by surveillance cameras. For example, a vehicle violating the speed limit is abnormal behavior,

and slipping/falling of a pedestrian on the road is abnormal behavior. However, from an algorithmic perspective, the analysis of surveillance videos is straightforward and can be done by subtracting static background and choosing frames with significant foreground objects.

On the other end, the majority of the sports/movies video analysis comprises predefined events/criteria such as leading characters and action scenes for movies [23, 64], and specific events in a sports video [21, 46, 121, 161]. For example, for soccer video analysis, the predefined events could be a goal, the movement of the ball, the position and/or movement of players, etc [27, 146].

The expeditious progress in technology has made wearable cameras [60, 123, 138] affordable and popular, and apart from recreational purposes, these wearable cameras are increasingly being used in law enforcement, geriatric care (for old people), and lifelogging applications. The egocentric videos possess unique characteristics (refer to Table 1.1) because of the specialized position of the camera and peculiar recording style (moving camera). Due to these unique characteristics, third-person video analysis techniques fail for egocentric video analysis. For example, the `SOTA` third-person video analysis approaches use advanced frameworks such as TCNs [40, 41, 89, 110], LSTMs [18, 96, 109, 110, 145, 188, 199], GCNs [5, 175, 190], and Transformer networks [58, 139] that are not scalable for massively long sequences and/or are not applicable for the unlabelled data. Furthermore, the third-person action/activity recognition works rely on subject pose information and hence are not applicable for egocentric videos [75, 178]. Similarly, traditional `SOTA` approaches for egocentric video analysis rely on predefined objects and people present, are specific to known environments (e.g., daily life, cooking video) and fail for the unseen environments common in egocentric videos [91, 104]. Hence, we require a new set of video analysis techniques specifically designed to address the above challenges and efficiently consume the massive volume of data resulting from the egocentric videos.

1.5 Aims & Research Questions

Motivated by the challenges, this dissertation aims to design novel scalable and unsupervised video analysis frameworks for massively long (up to 60K time steps) multivariate sequences, suitable for high-level video analysis tasks viz temporal segmentation, summarization, and activity recognition.

We articulate the following research questions for the three fundamental video analysis tasks:

1. Temporarily segment day long egocentric videos where the length of events is very dynamic (ranging from a few seconds to hours).

2. Summarize day long egocentric videos while maintaining the representativeness of the whole video. Personalize summaries by incorporating user feedback (in the form of video exemplars).

3. Recover activity patterns from one's weeks long lifelog.

1.6 Thesis Contributions

Overall, this dissertation is a significant feat to address the broader issues of scalability, privacy, and unlabeled data and establishes the SOTA. Suggested solutions represent the pioneering efforts for several video analysis tasks on massively long (up to 60k time steps) sequences in an unsupervised fashion. The proposed frameworks demonstrate novel deep learning and theory-based solutions for egocentric videos/photostreams analysis and show practical applicability in the real-life domain. All the works are deeply inspired by the recent advancement of deep learning, such as *LSTM*, *Reinforcement Learning*, *Transformer Networks*, etc., to pursue scalable solutions. Furthermore, We also demonstrate a statistical framework that deals with scalability in multivariate streaming data.

The dissertation explores three fundamental video analysis tasks: temporal video segmentation, summarization, and recovering activity patterns from a massively long multivariate sequence. Each task requires a different level of semantic understanding of massively long egocentric video sequences. To the best of our knowledge, we are the first to work on the Disney (up to 8 hrs video sequence) [51], and UTE datasets (up to 5 hrs) [90, 104] for temporal segmentation and summarization and EgoRoutine dataset (up to 20 days long photostream sequence) [160] for activity patterns recovery. A brief introduction and contributions of the problems addressed are as follows:

1.6.1 Temporal Segmentation of Day Long Egocentric Videos

The long and unconstrained nature of egocentric videos makes it imperative to use temporal segmentation as an important pre-processing step for many higher-level inference tasks. Activities of the wearer in an egocentric video typically span over hours and are often separated by slow, gradual changes. Furthermore, the change of camera viewpoint due to the wearer's head motion causes frequent and extreme but spurious scene changes. This work presents a novel statistical unsupervised temporal segmentation technique especially suited for day long egocentric videos. We formulate the problem as detecting concept drift in a time-varying, non i.i.d. sequence of frames. Statistically bounded thresholds are calculated to detect concept drift between two temporally adjacent multivariate data segments with different underlying distributions while establishing guarantees on false positives.

Contributions:

1. To the best of our knowledge, we are the first to formulate the problem of temporal segmentation of extremely long egocentric videos.

2. We use a multivariate generalization of Hoeffding's bound to compute distribution invariant segmentation threshold for multivariate time series arising out of a given frame sequence.

3. Our technique gives significantly improved f-score of 59.44%, on HUJI dataset [126], in comparison to current state of the art of 45.70% by [33].

1.6.2 Summarization and Personalized Summarization of Day Long Egocentric Videos

Egocentric videos (specially in lifeloging) comprise repetitive and long uninteresting portions. For efficient consumption and indexing, automatic summarization is imperative. Video summarization aims to create a compact and comprehensive synopsis by selecting the most informative parts of the original video [9]. The problem is a well-studied area in computer vision and broadly divided into two styles: *keyframes* and *video skims*. The proposed work focuses on *video skims* based summarization, where the summary is generated by the collection of video segments extracted from the original video sequence. While recording, the camera wearer often moves in a variety of scenes and performs various daily activities. The characteristics rule out techniques relying on the detection of important pre-specified events or objects. Further, obtaining annotated samples for summarization is hard for egocentric videos, often captured in an enhanced privacy scenario. Therefore, this work proposes a novel sliding window-based unsupervised deep reinforcement learning (RL) technique to summarize egocentric videos spanning 4 to 8 hrs. While generating visually diverse summaries, it is observed that the summarization criteria are inherently personal. Specifically, in the day long lifelogs, the same user may want to explore the summary focusing on the different types of events like social interaction, having food, walking, etc. Hence, we propose interactive summarization to personalize summaries by interactively collecting user feedback on-the-fly.

Contributions:

1. To the best of our knowledge, this work is the first work to summarise arbitrary long input videos and can be trained to generate summaries of various lengths. We demonstrate it by generating 1, 5, 10, and 15 minutes summaries of day long egocentric videos from several benchmark datasets [51, 90, 104, 125, 127].

2. Our approach can focus on various user-specified saliency criteria for the summary, such as distinctiveness, indicativeness, and object, or motion saliency.

3. We also propose an interactive summarization framework that can personalize summaries based on the length, content as well as interactive feedback from the user.

4. We achieve state-of-the-art performance on benchmark egocentric video datasets. We report Relaxed F-score [33] of 29.60 against 19.21 from the `SOTA` [199]. We also report BLEU score of 11.55 from our approach in comparison to 10.64 by the `SOTA` on the Disney dataset [51].

5. Though our focus is on egocentric videos, our technique can summarize hand-held videos as well. We obtain F-score of 46.40 and 58.3 on SumMe [61] and TVSum [153] datasets respectively, against the `SOTA` scores of 41.4 and 57.6 respectively.

1.6.3 Recovering Activity Patterns from Weeks Long Lifelog (photo-streams)

In lifelogging, egocentric videos are recorded across weeks to months. High-level analysis tasks over lifelogs include recognizing daily living activity (ADL), routine discovery, event detection, anomaly detection, etc. They require pre-processing with a self-supervised/unsupervised temporal segmentation and an activity indexing technique that deals with extremely long sequences. We show that traditional sequential models viz RNNs, LSTMs, GCNs, and Transformers fail to capture the long global dependencies required to address this problem, where similar events are often distributed across different days and even weeks. To this end, we propose a novel architecture named **S**emantic **A**ttention **T**rans**Former** (`SATFormer`), for representation learning in a very long photo-stream sequence.

Contributions:

1. We propose a novel `Transformer` architecture (`SATFormer`) based on the low-rank factorization of the self-attention matrix using proposed representative loss. The proposed architecture can exploit semantic cues to learn robust representation from extremely long video sequences.

2. We propose a self-supervised training scheme to discover activity patterns in extremely long egocentric lifelogs (recorded for up to 20 days). The approach does not rely on any priors, pre-trained networks to detect activities, objects, and/or places, and is specifically developed for unconstrained egocentric videos.

3. We demonstrate the performance of our contributions on the benchmark *Egoroutine* dataset. The proposed techniques using the `SATFormer` module gives a performance of 0.68/0.68/0.79 in terms of NMI/AMI/F-Score metrics, compared to 0.60/0.60/0.64 by the `SOTA`.

4. We also contribute annotations for the daily routines of all 7 subjects in the dataset comprising 104 days of life-logging data.

1.7 Thesis Structure

In chapter 2, we will discuss various SOTA works aligned to the three fundamental video analysis problems. Chapter 3 introduces the first problem titled temporal video segmentation. In this work, we demonstrate to temporarily segment the day long video (ranging up to 8 hrs) into possible events. In chapter 4, we introduce summarization and interactive summarization of day long video. This work demonstrates the summarization of day long sequence using a sliding window framework using three basic RL frameworks: policy gradient, Q learning, and Actor-critic. We also propose an interactive summarization framework that can personalize summaries based on the length, content, and interactive feedback from the user. In chapter 5, we introduce activity patterns recovery from weeks long photo-stream lifelogs using self-supervised learning. Chapter 6 presents the thesis conclusion by summarizing the contributions and proposing several perspectives about future research directions.

Chapter 2

Related Work

Most of the works in the analysis of egocentric videos focus on action recognition [52, 82, 122, 149, 186], and summarization [67, 91, 95, 104, 173, 179] tasks. Furthermore, the supervised methods [52, 102, 103, 105, 122, 148, 149, 186] have dominated the field, whereas relatively fewer works have been demonstrated in unsupervised settings [15, 53, 67, 82, 95, 104, 173]. We will elaborate upon these works in detail in subsequent sections.

On the other end, many interesting problems are addressed for egocentric photo-stream lifelogs [2–4, 20, 36, 66, 119, 136]. Most of the works focus on extracting social interaction patterns in egocentric photo-stream lifelogs [2–4, 66]. Herruzo et al. [66] use traditional classifiers such as kNN, SVM, and `SOTA` CNNs to classify the photo-stream into three patterns of interest, namely socializing, eating, and sedentary. Aghaei et al. in [2] employs LSTM based classification model for social interaction pattern extraction, and in [3] harness high-level image features and employ LSTM for detection and categorization of social interaction into formal and informal gatherings. Aghaei et al. [4] proposed an unsupervised agglomerative clustering approach to identify unique interaction in photo-streams. Furnari et al. [54] study how personal location from the user's lifelog can be recognized and localized from egocentric videos. They segment egocentric videos into fixed personal locations specified by the user like car, office, kitchen using Hidden Markov Model (HMM). Similarly, one of the preferences is to analyze people's food interaction for healthcare and geriatric care. Sarker et al. [136] introduce a new dataset titled 'EgoFoodPlaces' and trained an atrous CNN to recognize recurrences of a person on food places. Cartas et al. [20] use a CNN-LSTM model with the fixed batch size and overlap to capture the temporal evolution of high-level features in photo-stream for some predefined categories. The method uses a very short duration of temporal window size (5, 10, and 15 frames), which helps to detect temporal boundaries without explicitly knowing the boundaries. The techniques used for egocentric photo-stream analysis are not often applicable for day long egocentric video sequences (High Temporal Resolution) because they vary long and comprise a very smooth transition between the actions/events.

We now discuss various SOTA works related to temporal segmentation, summarization and personalized summarization, and activity patterns recovery.

2.1 Temporal Segmentation of Day Long Egocentric Videos

Related Tasks: We note that the solution to action localization, action detection, and scene segmentation results in the temporal segmentation of videos. Action localization refers to predicting the temporal bounds of pre-specified action categories in an input video. Researchers have looked at the problem of action localization, and action detection in both third person [6, 18, 22, 39, 48, 96, 110, 144, 145, 145], as well as the first person contexts [2, 15, 20, 71, 82]. Many temporal action localization/action detection works are supervised and demonstrated for untrimmed videos [56, 144, 145]. For example, Shou et al. [145] demonstrate action localization in untrimmed videos. The proposed framework is a three-stage framework that uses frame-level annotations. The untrimmed video is divided into small segments. In the first state, the proposal network classifies the foreground activities from the background activities. The proposal network eliminates the background activities to a large extent, and the foreground activities are given as input to the classification network. In the second state, the classification network uses C3D CNN to harness Spatio-temporal information and gives a confidence score for each segment. In the third stage, the classification network initializes the localization network. It better aligns the predictions in time with the ground truth using a novel loss function and outputs the confidence score. In the end, the NMS removes the redundancy to output the results. However, for the proposed problem, we do not have ground truth labels, so such works are not applicable. Furthermore, many works use segmentation-based approaches to action localization that rely on labeled data [73, 154]. Another class of approaches is based on the detection and tracking of active objects [170], where they use specialized methods such as the object detector [74] and human detector [112, 182]. The above approaches are mostly supervised, whereas our focus is on unsupervised segmentation with no prior knowledge of output categories. Similarly, in a scene segmentation task, one looks at the boundaries separating two visually different scenes. In a scene segmentation scenario, the boundaries are usually sharp, which is not true for the case of egocentric videos. Besides, the wearer's head motion and the resulting sharp viewpoint changes may induce false segmentation using a typical scene segmentation technique.

However, very few works discuss boundary localization/activity detection and localization in an unsupervised setting. Xu, [174], propose a new approach to train pre-trained video representation networks that is helpful for downstream localization tasks by incorporating boundary-sensitive information. They synthesized the temporal boundaries in existing large-scale video action classification datasets. They used these synthesized boundary and action labels in a supervised setting to generate more robust representations. Hou et al. [70] proposed an unsupervised action localization approach that discovers sub-actions for each action from the training videos and optimizes the temporal structure of sub-actions as the shortest path problem to locate the actions.

Unlike the above approaches, we aim to solve the temporal video segmentation problem in untrimmed and unconstrained videos where neither we have training videos nor

predefined actin/subactions classes. The proposed method can be used in a streaming mode without supervised/unsupervised training.

Deep Learning Techniques for Temporal Segmentation: In the last decade, DNNs have emerged as a leading technique for several computer vision problems, including the temporal video segmentation [1, 33, 40, 54, 84, 126]. Temporal Convolutional Networks (TCNs) and its variants [40, 41, 89] harness local motion information and use a hierarchy of temporal convolutional filters to capture longer range patterns. Similarly, Ding and Xu [40] propose a hybrid of LSTM and TCN to capture local motion and long range context. Most of the works use LSTM based generative model to predict the future context and track their evolution to decide the event boundaries in continuous video/photo-streams sequences [1, 33, 36]. These methods do not scale for hours long egocentric video segments, as the gradients during backpropagation vanish beyond a few hundred-time steps [93]. Besides, most of the techniques are supervised and require a large amount of training data, which is extremely hard in privacy-sensitive context.

Traditional Techniques for Temporal Segmentation: Traditional techniques for temporal segmentation of third person videos utilize variations of fixed-size sliding window approach to generate the start and end times of all the events in a video [48, 68, 73, 140, 145, 169]. These methods generally specify windows of different sizes and slide them across a video to generate event proposals of corresponding sizes. The overlapping proposals generated are further processed to remove overlap and select only the most relevant proposals. These methods are computationally expensive and require a large scale space search to handle events with significantly varying lengths, making them impractical for egocentric videos. For instance, in Disney egocentric dataset, events can be less than 5 minutes (social interactions), to more than 30 minutes (lunch).

Adaptive Windowing: Bifet and Gavalda [16] propose an adaptive windowing framework to detect distribution drift in streaming data. The adaptive windowing framework grows the window if the current distribution is long and drops a sub-window from the tail if a distribution drift is detected. The statistically bounded thresholds are calculated to detect distribution drift between two temporally adjacent sub-windows. However, this work is demonstrated on univariate and i.i.d data streams. Dimiccoli et al. [37] adapt [16] by using graph cut technique to look for the trade-off between the adaptive windowing [16] and agglomerative clustering. They further combine low-level features with high-level semantic labels and demonstrate event segmentation on egocentric photostream datasets. However, the method has been proposed for i.i.d. samples and heavy oversegments for dependent video streams.

Temporal Segmentation of Egocentric videos: Paci et al. [119] uses Siamese Neural Network to detect context change between two consecutive low-resolution images for egocentric photostream. Del et al. [33] and Dias et al. [36] use LSTM based generative model to predict the future context and track their evolution to decide the event boundaries in continuous photo streams. Dimiccoli et al. [37] also demonstrate temporal segmentation on egocentric photo-streams (as discussed in the previous paragraph).

2.2 Summarization of Day Long Egocentric Videos

Video Summarization: The majority of keyframe extraction techniques identify events using salient objects and video dynamics from various viewpoints and different degrees [194]. Zhang et al. [191] identify the content change in the video segment to extract keyframes. De et al. [31] find a cluster centroids as a representative of each cluster, which eventually derives the keyframes . However, video datasets exhibit lower inter-class and higher intra-class variance leading to difficulty in defining these clusters. Liu and Kender [97] have used a sequence reconstruction measure (SRM) to measure the degree to which selected keyframes can reconstruct the original video sequence. Dementhon et al. [88], and Latecki et al. [34] pick salient points of manifold formed by the representation of input frames as the keyframes. Dufaux [42] selects keyframes by considering high-level semantic criteria such as high motion, spatial activity, and the likelihood of having people. In contrast, Kang and Hua in [79] used attention, context dominance, and frame quality. These techniques work well for the targeted domain but do not generalize since the heuristic for frame selection is drawn from empirical observations. Video skims based summary generation typically require high-level context analysis and can be divided into four basic categories: (1) Redundancy elimination in a video by selecting a set of continuous frames that exhibit maximum similarity with input videos [156]. (2) Event/highlight detection and localization techniques which identify and locate the pre-defined events in a video sequence, such as sports videos, e.g. baseball [21], athletics [121], and cricket [161]. (3) Skim curve formulation techniques generate a curve that shows the likelihood of each base unit to include in the skim with respect to some user criteria. A threshold is used on the generated curve, and the segments above the threshold are assembled to form a final skim [107]. (4) Query context personalization which incorporates user feedback, either as a query or a personalized profile, e.g., [143] use human face, and caption text, and [11] use favorite players or a team preferred by the user.

Summarizing Short Hand-Held Videos: Supervised video summarization techniques have dominated the field of short video summarization [76, 188], where sequential determinantal point process, and LSTMs have been used to maximize various informative measures like representativeness, relevance, and uniformity in the learned summary. Unsupervised video summarization techniques have received more attention [61, 104, 109, 153, 199]. Some of the traditional works include low-level handcrafted informative measures like visual or motion cues for feature extraction and use various formulations for shot level importance scoring followed by variants of submodular function maximization to generating the summary [61, 111, 173, 179]. Higher-level informative measures, including diversity and representativeness, have been proposed recently [109, 151, 199]. Mahasseni et al. [109] use an adversarial learning framework for video summarization. Song et al. [151] proposed an RL technique to extract video category-specific keyframes. However, this work requires category information and keyframe labels during training. Zhou et al. [199] have extended the work with a reward function to

maximize diversity and representativeness in summary. This model is unsupervised but does not scale for videos longer than a few thousand frames.

Egocentric Video Summarization: Egocentric video summarization techniques often rely on important objects, and people present in the videos [91], and gaze tracking information (gaze provides a sense of the camera wearer's intent) [173]. Lin et al. [95] predict contexts of each video segment and use context-specific highlights to generate summaries. Similarly, Yao et al. [179] use a two-stream deep neural network (for spatio-temporal modeling) to generate highlight scores for each segment using the deep ranking model and generate summaries with these highlight scores. The inputs are a set of highlight and non-highlight video segment pairs, which are fed independently into two identical networks with shared parameters. A ranking layer is used in the end to evaluate the margin ranking loss of the pair. Both the streams are then late fused to generate the final highlight score. To overcome the scarcity of the first-person labeled data, Ho et al. [67] propose a deep neural network that transfers knowledge from third person video domain to egocentric videos for summarization. Lu et al. [104] propose story-driven summarization, which explicitly accounts for connectivity between the important entities. These entities are predefined important objects for the known environment and visual words for the unknown environment. Most of the techniques discussed above are specific to a video context (e.g., daily life or kitchen videos) and fail for the unseen environments.

Customizing Video Summaries: The summarization criteria are often user-specific viz inclusion of predefined object or event, presence of audio, duration of summary, etc. Hence generating customized summaries is an important sub-area of video summarization. Malino et al. [32] propose an interactive summarization framework that collects feedback from the user over the most frequent item in the original video. Then it iteratively refines the summary by a question asking interface. A probabilistic framework called active inference in the conditional random field (CRFs) is used to infer the summary preferred by the user. This work fine-tunes CNN on Places dataset [196] to detect most frequent objects or events, which is not feasible for the egocentric setting. Other works take user feedback in the form of natural language queries and use a mapping mechanism to bridge the gap between visual and language to personalize the summarization [181, 192]. Zhang et al. [192] select diverse sub-shots of a video that are representative of the whole video and yet related to a given user query in the natural language. They use a mapping network to connect visual and query space. This mapping network uses a relatedness reward to measure the distance between the predicted and ground truth query embedding for personalization. Similarly, Yousefi and Kuncheva [181] find all the frames related to the query using a semantic concept search. Jin et al. [77] segment video by analyzing visual features and speech detection and assign an importance score to each segment. It uses a variant of the knapsack problem to find an optimal video summary by fast-forwarding or removing unimportant segments. Han et al. [63] represent video by manifold embedding and assigns weights to each frame. Visual saliency features are applied between each pair of frames to learn the inherent

video structure. Darabi and Ghinea [30] use predefined categories to score each video segment using Scale Invariant Feature Transform (SIFT) features. The user feedback towards the high-level visual concepts is recorded in the vector form for personalization. After combining these two groups of data highest score video segments reflecting the user priority are returned. We emphasize that the techniques proposed in this work do not rely on the predefined objects or events and take user feedback in the form of video clips instead of text to reduce the overhead resulting from the use of cross-modality.

2.3 Recovering Activity Patterns from Weeks Long Lifelogs

Unsupervised activity recognition for egocentric videos: Kitani et al. [82] use a stacked Dirichlet process mixture (DPM) model. The first DPM learns the codebook of the motion histogram, and then the second DPM uses these codebooks to learn the ego-action for sports videos. This work follows the bag-of-words model and does not utilize long-term sequential or contextual information. Fathi et al. [50, 53] use a weakly supervised technique to model the active objects in a egocentric video sequence in an unconstrained environment when the domain-specific knowledge is not always available. Bhatnagar et al. [15] use CNN-LSTM based autoencoders to learn generic feature embedding by exploiting multi-resolution temporal information. Talavera et al. [160] use topic modeling to learn the activity patterns performed at different time intervals of the days. It uses dynamic-time-warping to classify a day-long photo-stream sequence into a routine/non-routine day (2-class classification problem). Yan et al. [176] formulate the problem as an optimization problem for multitasking clustering under the assumption that multiple individuals perform the same activities in similar environments e.g. working in the office often involves reading/writing papers and working on the computer. This framework uses low-level features such as optical flow and gaze information hence will not capture higher-level discriminative information necessary for unconstrained settings.

Self-supervised learning: Noroozi et al. [118] use a large network trained on a pretext task to generate pseudo labels for the target task and then train a smaller network with these pseudo labels by transferring the knowledge. For egocentric data, we do not have such large labeled data. Asano et al. [10] proposed the `SOTA` self-supervised representation learning framework that uses a fast variant of the Sinkhorn-Knopp algorithm to generate pseudo labels for large-scale datasets. However, the equipartition assumption used in the Sinkhorn-Knopp algorithm is not applicable for the problem as the distribution of activity patterns is highly skewed. Recently Zhan et al. [187] proposed an online deep clustering-based representation learning framework that steadily evolves the cluster centroids at each iteration and update the pseudo labels and simultaneously update the network parameters.

Representation Learning for modeling global dependencies: Sarfraz et al.

[134, 135] proposed a weighted hierarchical clustering approach that uses the 1-nearest neighbor graph to cluster the semantically consistent frames present in the video. Deep representation learning using graph autoencoder is getting attention for various NLP tasks [80, 168]. Park et al. [120] propose a symmetric GCN autoencoder for representation learning for NLP and image datasets. The work assumes a global relationship among the images and can also be adapted for video representation learning. However, all the GCN-based works require a pre-computed adjacency matrix which implicitly assumes a particular structure in the data. For example, Park et al. [120] use a pre-computed sparse affinity matrix using τ closest frames in the Euclidean space. In our problem, fixing a τ limits the generalization of our model to variable size events spanning across multiple days. Furthermore, computing the adjacency matrix requires a prior or semi-supervision, which is impractical in our setting.

Transformers: Recently, models based on `Transformer` architecture have shown `SOTA` performance in sequence modeling for various NLP tasks [166]. However, scalability of self-attention mechanism is a notable limitation of the transformer-based works for their applicability to long sequential inputs [35, 99]. The complexity of self-attention is $O(N^2)$ per layer (where N is sequence length) which quickly becomes intractable when N is large. Thus an active research area has emerged to gain compute and memory efficiency by approximating self-attention. A few notable works viz `Longformer`[14], `Reformer` [81], `Fast Transformer`[167], `Routing Attention`[132], `Long-Short Transformer` [200], and `Performer` [25] claim time complexities of $O(N)$, $O(N \log N)$, $O(NCm)$, $O(N^{1.5}m)$, $O(Nr)$, and $O(Nrm)$ respectively, where m, C and r are feature dimension, the number of clusters, and the dimension of the projection matrix, respectively.

Chapter 3

Temporal Segmentation of Day Long Egocentric Videos

3.1 Introduction

This chapter focuses on temporal video segmentation of day long egocentric video. Due to the task's utility as a pre-processing for many higher-level inference problems like indexing and summarization, the problem is a well-researched area in computer vision: both for the first person [16, 54, 124, 126, 172] as well as third person videos [84, 150, 185].

Common techniques for temporal segmentation of third person videos are based on either MRF formulation or deep neural network (DNN) with RNN/LSTM units. The former techniques [83] look for temporal discontinuities, and hence fail for egocentric videos when the segment boundaries are often slow with gradual changes in the scene. DNN based techniques [15, 39, 40, 110] use recurrent connections to capture the temporal context and do not scale well for long segments. To better understand the scales involved, a 10 minutes video segment captured at 30 frames per second (FPS) contains 18000 frames. Even with sophisticated back-propagation techniques [93], it is hard to train RNNs for such a long sequence. Multi-scale network designs [39, 41, 89] are possible but compromise temporal resolution to gain long term context.

For temporal segmentation of egocentric videos, researchers have suggested to use both generic (e.g. RGB, Optical flow, etc.) as well as egocentric specific cues (e.g. hand pose, handled object, etc.). However the techniques are often limited to either short segments [71] or segmentation based on long term activities but with short term signatures [15, 124, 126]. For example, to detect long term 'walking' activity, [126] independently classifies a video clip of 4 secs.

Another way to approach the problem is to use video compression works. Most of video compression works use motion information and image interpolation to reconstruct the frames in the original videos [92, 101, 171]. Egocentric videos are recorded using head/chest-mounted cameras in hands-free mode, which leads to very shaky videos; hence, we can not rely on the motion information. Furthermore, using this erroneous motion information for long segments is not obvious. For example, a subject walking

Figure 3.1: Challenges in temporal segmentation of egocentric videos. 1st row: Significant change in the scene due to head movement but there is no ground truth boundary. 2nd row: Segmentation boundary but no significant change in visuals.

from a building to outdoors can produce the same motion vectors in both contexts. Hence we need to include RGB information for temporal segmentation for egocentric videos.

We propose to formulate the problem of temporal segmentation as concept drift detection in multivariate time series data. In a concept drift detection task, one maintains two adjacent temporal windows of fixed size and estimate statistical summary (e.g. average) of the two windows separately. If the summary is significantly different for the two windows, the algorithm declares concept drift. The key challenges to use the formulation for temporal segmentation are: (1) Choosing window length for the statistical summary, as different activity/event lengths may require different temporal windows, and (2) Choosing threshold to declare a boundary, as real boundaries may have smooth visual changes, whereas sharp head motion may cause significant visual changes in non-boundary regions. We emphasize that the proposed formulation can incorporate various other cues suggested for temporal segmentation of egocentric videos viz optical flow, hand pose, and other objects present in the scene, etc. Our primary contribution is in suggesting a way to deal with smooth changes in the features at the real boundaries compared to sharper changes at the spurious boundaries as illustrated in Fig. 3.1.

Bifet and Gavalda [16] have suggested a technique, called ADWIN, to segment i.i.d. univariate sequences. Their method maintains an adaptive window, and for each of its various partitions into two sub-windows, a threshold is calculated based upon the harmonic mean of the length of the two sub-windows. A boundary is declared if the difference of the statistical summaries of the two sub-windows is larger than this threshold. The threshold is based on the Hoeffding's inequality and is valid for all probability distributions. ADWIN gives probabilistic bounds on the boundary detection error and works for univariate sequences with slow as well as abrupt changes.

The proposed concept drift detection approach is most appropriate for extremely long activities in untrimmed day long videos and significantly different from anomaly detection or similar works. Anomaly detection is the identification of rare items, events, or observations that deviate significantly from most of the data. An abundance of work

Methods	Unsupervised	Multivariate Data	Scalability to Long Sequences	Customized Granularity	Works with Extremely Shaky Videos
TCFPN [41]	✗	✓	✗	✗	✗
ADWIN [16]	✓	✗	✓	✓	✓
SR-Clustering [37]	✓	✓	✗	✓	✓
CES [33]	✓	✓	✗	✗	✗
Ours	✓	✓	✓	✓	✓

Table 3.1: Comparison of state of the art with our method on various criteria important for applicability to egocentric videos.

has been done for unsupervised anomaly detection [100, 184, 198]. Most of the works learn the dominant video structure within a model then anomalies are detected by either high reconstruction error or prediction error for some data samples. However, for the problem at hand, we deal with diverse and complex activities, so the assumption used in anomaly detection does not hold. Analogous to anomaly detection, many works focus on event detection in sports/movie videos and focus on predefined events/criteria. The predefined events for movies included the leading characters and action scenes for movies [23, 64]. For soccer videos, a goal, movement of the ball near the goal post, or movement of the players [27, 146] and for cricket, the boundaries and wickets [161]. The egocentric videos are recorded in an unconstrained setting, so we can not define events a priori so these approaches are not applicable for egocentric videos.

In this chapter, we propose a technique for concept drift detection in multivariate, and non-i.i.d. sequences such as egocentric videos, which can be used for temporal segmentation of such videos. Table 3.1 compares the key strengths of our approach with state of the art. The specific contributions of this work are as follows:

1. To the best of our knowledge, we are the first to suggest formulating the problem of temporal segmentation of extremely long egocentric videos as detecting concept drift in a time series data.

2. We use a multivariate generalization of Hoeffding's bound to compute distribution invariant segmentation threshold for multivariate time series arising out of a given frame sequence.

3. Hoeffding's bound as such assumes i.i.d. samples and can not be used for video sequences with a large correlation between temporal neighbors. We suggest a simple heuristic of jump factor to get around the problem.

4. In our experiments on both day long egocentric videos, as well as benchmark photo-stream datasets, the proposed technique successfully copes with two key egocentric

specific challenges viz continuous as well as extreme viewpoint variations, and long segments. Our technique gives significantly improved f-score of 59.44%, on HUJI [126], in comparison to current state of the art of 45.70% by [33].

3.2 Proposed Approach

We start this section with our theoretical contributions. Since the target of this work is detecting context drift in a stream of video frames, represented as vectors in $\mathbb{R}^d$, we first extend the standard Matrix Hoeffding's bound to the special case of $d \times 1$ matrices, which is our case. Then we use the derived bound for our novel concept drift detection formulation in multivariate sequence. While the discussion until here will assume the input samples (frames in our case) to be independent, we end the section with details on how to deal with temporally correlated data streams.

3.2.1 Multivariate Hoeffding's Bound

The standard result for Hoeffding's inequality for random symmetric matrices may be given as the following [108]:

Lemma 3.1. *Consider a finite sequence Z_i of independent, random, symmetric matrices with dimension d, and a sequence of fixed symmetric matrices P_i, such that $\mathbb{E}[Z_i] = 0$ and $Z_i^2 \preceq P_i^2$, almost surely. Here, $\preceq$ denotes the semi-definite order on symmetric matrices. Then for all $\epsilon \geq 0$, we have:*

$$\mathbb{P}\left(\left\|\sum_i Z_i\right\|_s \geq \epsilon\right) \leq d \ \exp\left(\frac{-\epsilon^2}{2\sigma^2}\right), \tag{3.1}$$

where $\sigma^2 = \frac{1}{2}\left\|\sum_i (P_i^2 + \mathbb{E}[Z_i^2])\right\|_s$, and $\|X\|_s$ denotes the spectral norm of X.

For our case, we assume that $\mathbb{E}[Z_i^2] \approx Z_i^2$, and $Z_i \approx P_i$, and hence compute σ^2 as simply $\left\|\sum_i P_i^2\right\|_s$. Note that the result as such is valid only for the symmetric matrices. We extend it to the vector data-streams using the Jordan-WieLaudt theorem [155] as described below. Consider a vector X of size $d \times 1$. Let A be a block matrix such that $A = \begin{bmatrix} 0 & X \\ X^T & 0 \end{bmatrix}$. Since, A is a symmetric matrix with dimension $(d+1) \times (d+1)$, we can use Eq. (3.1) for the matrix A, such that:

$$\mathbb{P}\left(\left\|\sum_i A_i\right\|_s \geq \epsilon\right) \leq (d+1) \exp\left(\frac{-\epsilon^2}{2\sigma^2}\right), \tag{3.2}$$

where $\sigma^2 = \left\|\sum_i A_i^2\right\|_s$. It can also be shown that: $A^2 = \begin{bmatrix} XX^T & 0 \\ 0 & X^TX \end{bmatrix}$, and that A's non-zero eigenvalues are ± 1 times the singular values of X. Hence $\|A\|_s = \|X\|_2$, where

$\|X\|_2$ denotes the ℓ_2 norm of the vector X. Using the result in the equation above:

$$\mathbb{P}\left(\left\|\sum_i X_i\right\|_2 \geq \epsilon\right) \leq (d+1)\exp\left(\frac{-\epsilon^2}{2\sigma^2}\right), \tag{3.3a}$$

$$\text{where} \quad \sigma^2 = \max\left(\left\|\sum_i \mathbb{E}\left[X_i X_i^T\right]\right\|_s, \left\|\sum_i \mathbb{E}\left[X_i^T X_i\right]\right\|_s\right) \tag{3.3b}$$

We use the above result to compute the bound for the average as:

$$\begin{aligned}
\mathbb{P}\left(\left\|\frac{1}{n}\sum_i X_i\right\|_2 \geq \epsilon\right) &= \mathbb{P}\left(\frac{1}{n}\left\|\sum_i X_i\right\|_2 \geq \epsilon\right) \\
&= \mathbb{P}\left(\left\|\sum_i X_i\right\|_2 \geq n\epsilon\right) \\
&\leq (d+1)\exp\left(\frac{-n^2\epsilon^2}{2\sigma^2}\right). \qquad \text{(Using Eq. (3.3a))}
\end{aligned}$$

Denoting $\overline{X} = \frac{1}{n}\sum_i X_i$, and $\overline{\sigma}^2 = \sigma^2/n$

$$\mathbb{P}\left(\left\|\overline{X}\right\|_2 \geq \epsilon\right) \leq (d+1)\exp\left(\frac{-n\epsilon^2}{2\overline{\sigma}^2}\right), \tag{3.4}$$

Note that, if we assume the ℓ_2 norm of X as 1, then $X_i^T X_i = 1$, and σ^2 as given in Eq. (3.3b) is always 1. We summarize our result below:

Theorem 3.2. *Let $X_1, \ldots, X_n$ be d dimensional, independent random vectors with $\mathbb{E}[X] = 0$, and unit ℓ_2 norm. Then:*

$$\mathbb{P}\left(\left\|\overline{X}\right\|_2 \geq \epsilon\right) \leq (d+1)\exp\left(\frac{-n\epsilon^2}{2}\right), \tag{3.5}$$

where $\overline{X}$ denotes the observed mean of the samples.

3.2.2 Concept Drift Detection

We formulate the temporal segmentation of egocentric videos as concept drift detection in a data stream. While in reality, the adjacent frames in the video stream are not conditionally independent of each other, for this section, we will assume so. In the next section, we describe our proposal to get around the assumption.

Concept Drift Detection Pipeline: For the concept drift detection, one maintains a sliding window, w, of dynamic length, n, over the sequence. Consider a hypothesis that there is a segment boundary at index t within the window, i.e., there is a particular segment, w_1, of length n_1, from $[0, t)$ and another segment, w_2, of length n_2, from $[t, n)$. We assume that the data in two segments is from two unknown distributions with the observed mean values of $\widehat{\mu}_1$ and $\widehat{\mu}_2$ respectively. If for a particular partition, the score

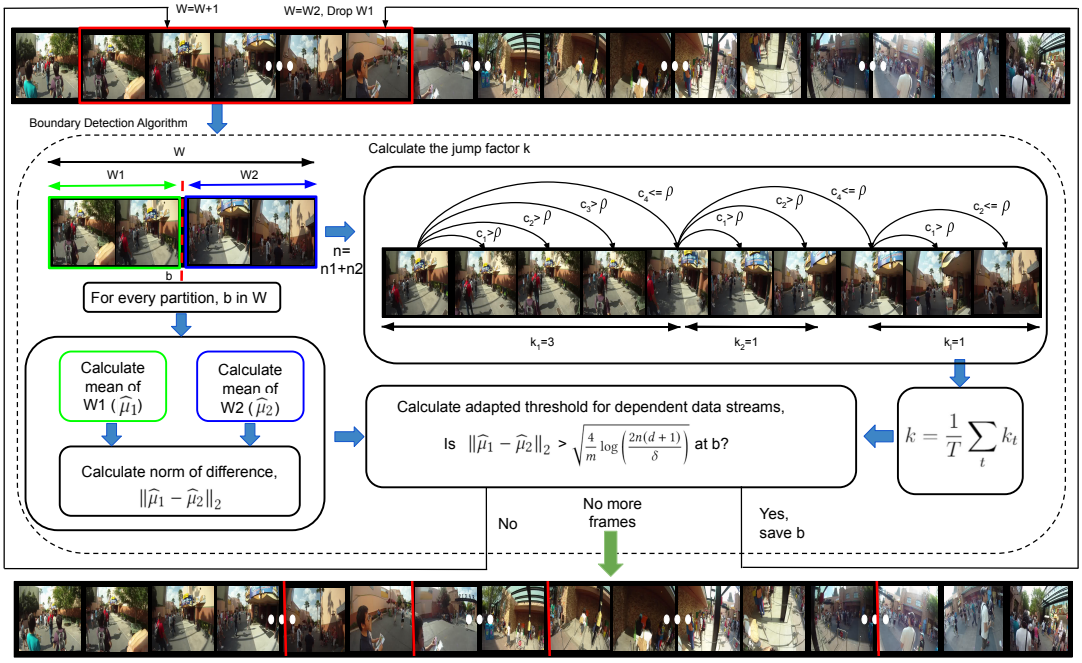

Figure 3.2: The block diagram describing major steps of the proposed approach. The c_i represents the correlation coefficient between the two frames. Please refer to the main paper for the details

$(\|\widehat{\mu}_1 - \widehat{\mu}_2\|_2)$ exceeds a threshold ϵ_{cut}, we would like to declare a detected boundary at t and the segment w_1 will be dropped from w. Otherwise, a new sample is added to the current window w, and the process is repeated for this new window of size $n+1$. For each window w, the boundary hypothesis is tested for all indices $t \in w$. Below we describe a way to compute the threshold ϵ_{cut} in a principled manner using multiple hypothesis testing.

Multiple Hypothesis Testing: One of the ways to calculate the threshold ϵ_{cut} is by bounding the error rate for declaring incorrect segment boundaries. Let us denote the observed mean of the segments, as $\widehat{\mu}_1$, $\widehat{\mu}_2$ respectively, and the true (unobserved) mean of the current window as μ_w. We perform hypothesis testing with $\widehat{\mu}_1 = \widehat{\mu}_2 = \mu_w$ as the null hypothesis. In other words, our null hypothesis is that the two segments come from the same, but unknown, distribution. Since we perform multiple tests in a single window for various values of t, hence as per the multiple hypothesis testing problem in the statistics, we would like to increase the threshold of accepting the hypothesis by n (size of the window or number of tests). For the hypothesis accepting the probability of

δ, we would like to set the ϵ_{cut} such that:

$$\mathbb{P}\Big(\|\widehat{\mu}_1 - \widehat{\mu}_2\|_2 \geq \epsilon_{\text{cut}}\Big) \leq \frac{\delta}{n}. \tag{3.6}$$

The following lemma bounds the probability of difference in the observed means:

Lemma 3.3. *For a sequence of d-dimensional random vectors, $\{X_1, \dots X_n\}$, sampled from an unknown but stationary probability distribution, and its arbitrary partition into two subsets w_1, and w_2, with lengths n_1, and n_2, and observed means $\widehat{\mu}_1$, and $\widehat{\mu}_2$ respectively:*

$$\mathbb{P}\Big(\|\widehat{\mu}_1 - \widehat{\mu}_2\|_2 \geq \epsilon\Big) \leq 2(d+1)\exp\left(\frac{-m\epsilon^2}{4}\right), \tag{3.7}$$

where m is the harmonic mean of n_1 and n_2.

Proof. Consider the following three events:

- **Event A:** $\|\widehat{\mu}_1 - \widehat{\mu}_2\|_2 < \epsilon$.
- **Event B:** $\|\widehat{\mu}_1\|_2 < k\ \epsilon$.
- **Event C:** $\|\widehat{\mu}_2\|_2 < (1-k)\epsilon$.

Here, k is a real number $\in (0, 1)$. Further, from triangle inequality:

$$\|\widehat{\mu}_1 - \widehat{\mu}_2\|_2 \leq \|\widehat{\mu}_1\|_2 + \|\widehat{\mu}_2\|_2 \tag{3.8}$$

Assuming Events B and C hold:

$$\Rightarrow \quad \|\widehat{\mu}_1 - \widehat{\mu}_2\|_2 < k\ \epsilon + (1-k)\epsilon \tag{3.9}$$

$$\Rightarrow \quad \|\widehat{\mu}_1 - \widehat{\mu}_2\|_2 < \epsilon. \tag{3.10}$$

Hence, we can say that $B \cap C \subseteq A$, which implies $A^c \subseteq B^c \cup C^c$, where S^c denotes the complement of the set S. Therefore, from union bound rule of the probability theory:

$$\mathbb{P}\left(A^c\right) \leq \mathbb{P}\left(B^c\right) + \mathbb{P}\left(C^c\right) \tag{3.11}$$

Using event defintions as given above:

$$\mathbb{P}\left(\|\widehat{\mu}_1 - \widehat{\mu}_2\|_2 \geq \epsilon\right) \leq \mathbb{P}\left(\|\widehat{\mu}_1\|_2 \geq k\ \epsilon\right) + \mathbb{P}\left(\|\widehat{\mu}_2\|_2 \geq (1-k)\epsilon\right)$$

Using Theorem 3.2

$$\begin{aligned}\mathbb{P}\left(\|\widehat{\mu}_1 - \widehat{\mu}_2\|_2 \geq \epsilon\right) \leq\ & (d+1)\exp\left(\frac{-n_1 k^2 \epsilon^2}{2}\right) \\ & + (d+1)\exp\left(\frac{-n_2(1-k)^2\epsilon^2}{2}\right)\end{aligned} \tag{3.12}$$

The equation above holds for all values of k. Hence, to get the tightest upper bound of the left hand side (l.h.s.) of the above equation, we minimize the right hand side (r.h.s.) with respect to k. Here, we note, and also done in [16], the r.h.s. is approximately minimized when the exponents of the two terms are equal:

$$k^2\epsilon^2 n_1 = (1-k)^2\epsilon^2 n_2 \tag{3.13}$$

$$\Rightarrow \quad k = \sqrt{(n_2/n_1)}/(1+\sqrt{(n_2/n_1)}) \tag{3.14}$$

For this value of k, we have:

$$k^2\epsilon^2 n_1 = (1-k)^2\epsilon^2 n_2 = \frac{n_2 n_1}{(\sqrt{n_1}+\sqrt{n_2})^2}\epsilon^2 \tag{3.15}$$

$$\leq \frac{n_2 n_1}{(n_1+n_2)}\epsilon^2 = \frac{m}{2}\epsilon^2, \tag{3.16}$$

where m is the harmonic mean of n_1 and n_2. We can use the values to get the tightest upper bound for the l.h.s. of Eq. (3.12) as:

$$\mathbb{P}\Big(\|\widehat{\mu}_1 - \widehat{\mu}_2\|_2 \geq \epsilon\Big) \leq 2(d+1)\exp\left(\frac{-m\epsilon^2}{4}\right) \tag{3.17}$$

Hence proved. □

Calculating ϵ_{cut}: As noted in Eq. (3.6), and the accompanying discussion, we would like to choose a value of ϵ which enables us to declare a concept drift and hence the segment boundary if the ℓ_2 norm of the difference of the observed means of the two segments goes beyond ϵ. Further, the hypothesis testing framework allows us to choose a value of ϵ according to the threshold of accepting the hypothesis δ, which bounds the error rate for declaring incorrect segment boundaries to δ. Since Lemma 3.3 bounds the probability of difference of observed means exceeding ϵ, we can use it to choose a value of ϵ (denoted as ϵ_{cut} hereon) such that we get the desired upper bound on declaring the false boundary:

$$2(d+1)\exp\left(\frac{-m\epsilon_{\text{cut}}^2}{4}\right) \leq \frac{\delta}{n} \tag{3.18}$$

$$\Rightarrow \quad \epsilon_{\text{cut}} \geq \sqrt{\frac{4}{m}\log\left(\frac{2n(d+1)}{\delta}\right)} \tag{3.19}$$

3.2.3 Handling Conditionally Dependent Data

It may be noted that the derivation of ϵ_{cut} using the Hoeffding's bound is valid only when the data is identically and independently distributed (i.i.d). The assumption is

invalid for egocentric video stream where a frame is highly correlated with its temporal neighbor. One way to resolve the problem is by making the data conditionally independent. We observe that the correlation between the frames decreases as the temporal distance between them increases. We fix a threshold and declare two frames independent if the correlation coefficient between them is below the threshold. This is effectively sub-sampling the video.

We discover the optimal sub-sampling rate from the data itself. For the first frame t in a given window W, we find the frame $t + k_t$ for which the correlation coefficient is less than a threshold ρ_c. The process is then repeated from frame $t = t + k_t$, and is continued until the end of the window is reached. We select the sub-sampling rate, k, as the average of k_t for all t.

We further optimize the proposed pipeline by observing that we do not really need to sub-sample the video, but the effect of sub-sampling can be incorporated in the threshold ϵ_{cut} itself. Consider an extreme scenario, when the original samples were conditionally independent, but we introduced a severe correlation by duplicating a sample r times. Note that in this case, the ground truth boundary should not shift but the length of the segments W_0 and W_1 just increases by r times. The harmonic mean m also increases by r times, thus effectively decreasing segmentation threshold ϵ_{cut}, and leading to over-segmentation. We compensate for the reduction in ϵ_{cut} by updating the expression to:

$$\epsilon_{\text{cut}} \geq \sqrt{\frac{4k}{m} \log\left(\frac{2n(d_1+1)}{k\delta}\right)} \tag{3.20}$$

where k is the sub-sampling rate for un-correlating the input data, as described earlier. Note that the exact choice of correct k is not very critical, but merely helps to virtually sub-sample a video such that the i.i.d. assumption starts to holds, by penalizing the effect to ϵ_{cut}. However, the role of k becomes more important to normalize videos taken at different temporal resolutions (frames per second). The proposed approach avoids over-segmentation of a video by adjusting the threshold for videos at the higher temporal resolution, leading to higher accuracy in boundary prediction. Note that the discussion above does not address the problems when videos are captured at extremely low temporal solution, which we discuss next.

3.2.4 Handling Photo-stream Data

Imagine we had a video, and have found an optimal sub-sampling rate k at which the adjacent frames become conditionally independent. Note that, any larger k will also satisfy the independence constraint, but will lead to under-segmentation. We observe that when the input is a photo-stream, the frames are indeed conditionally independent, but they would likely be independent (as per our correlation coefficient criterion) even when we insert an additional frame (by interpolating neighboring frames) in between. We believe that our method underestimates the length of the segment in the case of photo-streams due to the above reason. Therefore, for the photo-streams, we suggest

Algorithm 1 Temporal Segmentation Algorithm

Input $F_{i=1}^{N}$**:** Feature vector of video frames
Output $B_{i=1}^{M}$**:** Predicted Boundaries

Initialize the window W
for each frame x_{t} **do**
  $W \longleftarrow W \cup \{x_{\mathrm{t}}\}$
  Compute average skip factor k in current window by a user defined correlation coefficient ρ_c
  Flag=False
  Possible Boundaries B
  for each n split of W into W_1 . W_2 **do**
    Compute threshold, $\epsilon_{\mathrm{cut}} \geq \sqrt{\frac{4}{m}\log\left(\frac{2n(d+1)}{\delta}\right)}$
    if $\|\mu_1 - \mu_2\|_2 \geq \epsilon_{\mathrm{cut}}$ **then**
      splits $= \|\mu_1 - \mu_2\|_2 - \epsilon_{\mathrm{cut}}$
      $B \longleftarrow B \cup best(splits)$
      Flag = True
    end if
  end for
  if Flag==True **then**
    Drop window W_1 from W along best boundary B
  end if
end for

to look for the smallest number of k frames, which when inserted in the photo-stream still keeps the neighboring frames independent. We introduce these frames, or feature vectors as the case may be, by simply averaging the features of two consecutive frames. The process is continued until the correlation coefficient of the feature vectors remains below a user specified threshold.

However, similar to the way we handled correlated frames in the videos, we do not need to make the actual addition of frames to the dataset. We just need to know the length of the adaptive window, when the frames will be added to the window. This new window length is then used to modify the threshold. The modified threshold used for the photo-stream is as follows:

$$\epsilon_{\mathrm{cut}} \geq \sqrt{\frac{4}{mk}\log\left(\frac{2nk(d+1)}{\delta}\right)} \tag{3.21}$$

Fig. 3.2 shows the block diagram of the proposed approach and Algorithm 1 presents the pseudo-code.

3.3 Experiments

3.3.1 Datasets

We demonstrate the results of proposed approach on three extremely long egocentric video datasets, viz HUJI [124, 126], Disney [51], and UTEgo [90, 104], as well as on the standard photo-stream dataset, viz EDUB-Seg20 [37, 159]. The detailed description of datasets is as follows.

HUJI dataset: HUJI dataset consists of video sequences captured by GoPro camera by three users at a temporal resolution of $30fps$. The dataset comprises several small video clips of less than 30 minutes. For each user, we merged their corresponding small clips into one big video in the specified order. We have evaluated on the videos (of length 4 hours and 2 hours) recorded by only two users using the ground truth boundaries made available by [37]. This is due to the unavailability of the ground truth for the third one. The number of frames in the longest video sequence is 72217.

Disney dataset: Disney dataset consists of videos captured at Disney world by 6 individual for three days. Similar to the HUJI dataset, for each user, we have merged several small video clips in the order of the numbering provided by the user. After merging we have a total of 8 video sequences of 4-8 hours for each individual user. We have generated our own ground truth by three different annotators. The number of frames in the longest video sequence is 151695.

UTEgo dataset: UTEgo dataset comprises of 4 videos captured by Looxcie wearable camera at a temporal resolution of $15fps$. These videos are 3-5 hours long and captured in an unconstrained setting. We have manually labeled the ground truth for this dataset as well. We will make our annotations public, post acceptance. The number of frames in the longest video sequence is 92287.

EDUB-Seg20: We also demonstrate results on a photo-stream dataset namely EDUB-Seg20. The dataset comprises 18735 images captured through Narrative Clip which captures 2 pictures per minute. The pictures are taken by 7 different users over 20 days. The dataset comprises a variety of scene contexts, viz, attending a conference, traveling, working in the office, etc. EDUB-Seg dataset is released in two versions EDUB-Seg12 comprises 12 videos and EDUB-Seg20 which is the extension of EDUB-Seg12 with 8 new videos. Though our focus is on long videos and not short photo-streams, the evaluation of this dataset allows us to compare our technique against existing temporal segmentation methodologies for egocentric photo-streams.

Figure 3.3: The segmentation granularity increases as we increase δ in our approach. The three rows in the figure show the output from our approach at δ, 10^{-6}, 10^{-4}, and 10^{-2} respectively, on the 'Alireza Day 1' sequence from Disney dataset. The bars above each row indicates the time instance of frames chosen as a boundary, such that the length of the row shows the length of the sequence.

3.3.2 Implementation Details

Feature Vector and Initial Window Length: For all the video datasets, we use the input at 5fps and use frame-wise AlexNet [85] features as used by SR-Clustering [37]. However, for a fair comparison on the photo-stream datasets, we use LSTM features similar to one used by [33]. However, since we operate in the streaming mode in our application, instead of bi-direction features as suggested in [33], we use only unidirectional features. We set the initial window length to 20 frames in all the experiments.

Frame Correlation Coefficient: As discussed earlier, to make the frames independent for meeting the requirements of our theoretical results, we use the notion of skip factor. The learned skip factor requires a hyper-parameter correlation coefficient threshold, ρ_c to declare the two frames independent. We have chosen $\rho_c = 0.95$ for video datasets. However, we observe that LSTM features used for the photo-stream datasets exhibit a high correlation. Hence, we use $\rho_c = 0.999$ for the photo-stream datasets. Furthermore, the value of ρ_c should indeed be per video, depending upon the conditional independence and hence improve the performance. However, in practice we do not have access to such information, hence we have picked a particular value, which is fixed for all the videos.

Granularity: Any segmentation problem is inherently dependent upon the scale one is looking for. In our technique, the granularity at which the user wants their video to be segmented can be controlled by the δ. As seen in Fig. 3.3, as the value of δ increases, the number of segments increases, and boundaries are detected even for smaller changes. Similarly, upon decreasing the value of δ, the number of segments decreases, corresponding to capturing large heterogeneous context in a single event. In general application of our technique, we expect that such a granularity could be taken

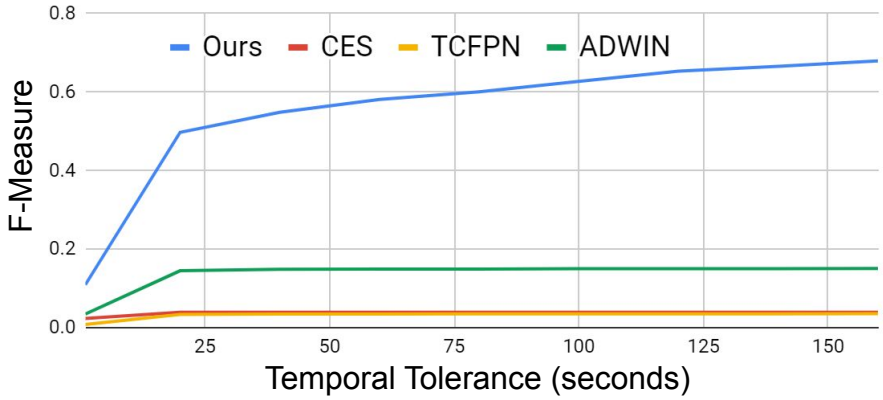

Figure 3.4: The figure shows the F-Measure comparison between SOTA and proposed approach for different values of temporal tolerance for the Disney dataset.

as feedback from the end-user. However, for comparing with benchmark datasets, we do not have such user-feedback available. Hence we use average segment length as the proxy for the segmentation granularity required. We define 2500-3000, 1600-2500, and 1000-1600 frames per segment as our ranges for low, medium, and high levels of granularity, respectively. We set the δ for the corresponding granularity as 10^{-6}, 10^{-4}, and 10^{-2} respectively. Similarly, for the photo-stream datasets of HUJI, UTEgo, Disney, and EDUB-Seg20, we experiment with δ values of 10^{-7}, 10^{-3}, and 10^{-1} for different levels of granularity.

Boundary Tolerance: As proposed by [33], when dealing with continuous boundaries in an egocentric video, there is an inherent ambiguity in annotating the exact frame which should be marked as the boundary, and many frames in the temporal vicinity could have been marked as a boundary as well. Hence, penalizing an algorithm for marking the exact frame as a boundary may not indicate the true strength of the technique. [33] has proposed the use of temporal tolerance, which allows a technique to be rewarded if it predicts a boundary within a certain range of the ground truth. We adopt the metric in our experiments and use a temporal tolerance (tol) of 2.5 minutes to calculate the performance (f-measure) of our technique. As shown in Fig. 3.4, the boundary detection accuracy improves as the value of temporal tolerance is increased.

Hardware Requirements: The proposed technique is implemented on Matlab with system architecture comprising of Quadro P5000 GPU and Intel i7 processor with 4 cores (32 GB RAM). It takes approximately 2 hrs (inclusive of feature extraction) and approximately 8GB CPU RAM to segment 8 hrs long video.

Methods	HUJI	UTEgo	Disney
TCFPN [41]	4.18	2.50	3.56
ADWIN [16]	12.44	0.83	15.01
CES[33]	4.52	9.31	3.96
Ours	**73.01**	**58.41**	**67.63**

Table 3.2: F-Measure comparison on video datasets

3.3.3 Evaluation Measure

We use the averaged F-measure to evaluate our performance. As proposed in [33], we consider a predicted boundary as true positive if it occurs within the tolerance(tol) neighborhood of a ground truth boundary, while taking into consideration that this ground truth boundary has not already been matched to a predicted boundary before. Analogously, all the ground truth boundaries, for which no frame within its tol range has been predicted, are referred to as false negative. We also evaluate our method based on the number of segments predicted. The metric is used to show reduction in over-segmentation achieved for video data using our method.

3.3.4 Comparative Evaluation

For comparison on video datasets, we pick two representative techniques to compare against, viz CES [33] and TCFPN [41]. We also compare against ADWIN [16] which is based on unsupervised concept drift detection but does not handle multivariate data or correlated samples. For comparison with ADWIN, we pretend the data is uncorrelated and convert a feature vector into a single scalar by taking its ℓ_2 norm. We ignore the SR-Clustering [37] for the video datasets because it doesn't scale for day-long video sequences.

Since many of the approaches we compare against were originally targeted for photo-streams and not videos, therefore, to ensure a fair comparison, we prepare two configurations for each dataset. In the first configuration, we resample a video at 2 frames per minute, thereby making it resemble a photo-stream. In the second configuration, each input video is resampled at 5fps to match the lowest temporal resolution of all the datasets. For photo-stream datasets, we also compare with SR-Clustering [37]. Table 3.2 shows the quantitative evaluation based on F-measure for $tol = 750$ for video datasets. We notice significant performance improvement over all the state of the art approaches as these techniques fail to handle the daylong video sequences.

Fig. 3.5 shows a qualitative visualization of the comparison between various state of the art techniques and the proposed approach. The bar chart shows the frames selected as a boundary by different techniques for a 30 minutes clip. It is clear that the state of

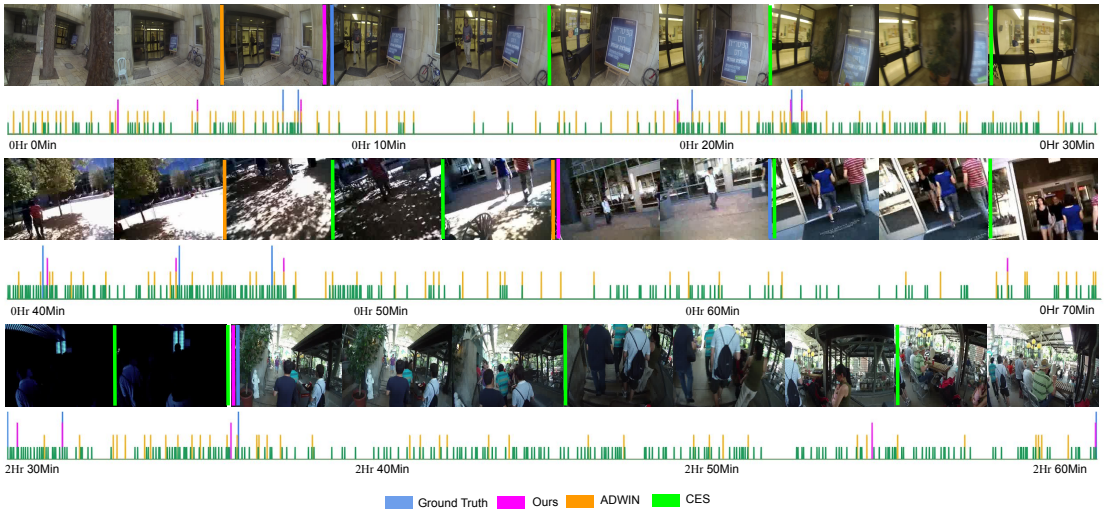

Figure 3.5: Temporal segmentation of long egocentric videos: The figure shows a qualitative representation of closeness of boundaries predicted by the proposed approach, ADWIN [16], CES [33] to ground truth boundaries from specific portions of Huji (first row), UTEgo (second row) and Disney (third row) datasets (better visualize in colors). Please see the text for details.

the art techniques severely over-segment all the video sequence datasets due to frequent scene changes accompanying the sharp head motion of the wearer. The images above each of the bar charts show representative frames from a short video segment from each of the clips. The boundaries selected by each technique are marked by thick colored lines between the frames. This is for visual comparison of the frames where different techniques choose to create a boundary. From the figure, we can observe that the proposed approach doesn't over-segment and precisely locates the temporal boundaries.

Table 3.3 shows the F-measure for $tol = 5$ for photo-stream datasets (EDUB-Seg as well as all the video datasets down-sampled to photo-streams as described earlier). For photo-stream datasets also we show considerable improvement. We report 13.74%, 24.42%, and 7.43% improvement in F-measure for HUJI, UTEgo, and Disney datasets respectively, however, for EDUB-Seg20 we under-perform marginally as CES [33] uses bidirectional features, whereas we use uni-directional features to maintain the online streaming mode property of our technique. Fig. 3.6 shows the visualization for photo-stream datasets. The first row shows the visualization for the EDUB-Seg20 dataset where the CES [33] performs competitively. For the HUJI dataset proposed method performs better than the CES [33].

As mentioned in section 3.3.2, we use frame-wise AlexNet [85] features as used by SR-clustering [37] for a fair comparison. We have also tried other pre-trained CNN; namely, VGG [147], GoogleNet [157], and ResNet101 [65], and achieved marginal performance

Methods	Features	EDUB	HUJI	UTEgo	Disney
TCFPN [41]	CNN	19.26	2.37	1.37	3.84
ADWIN [16]	CNN	35.37	44	11.47	23.21
CES [33]	LSTM-Bi	**69**	45.70	36.19	61.40
SR-Clustering [37]	CNN	49.93	44.06	9.44	55.81
Ours	LSTM-uni	63.96	**59.44**	**60.61**	**68.83**

Table 3.3: F-Measure comparison on photo-stream datasets

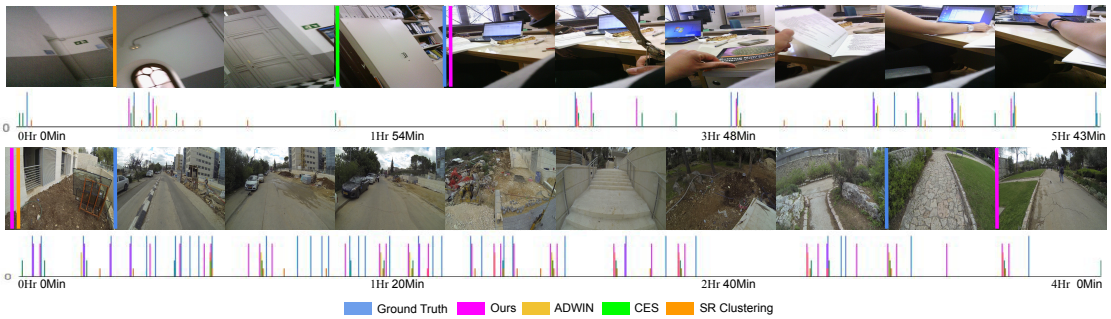

Figure 3.6: Temporal segmentation of photo-stream data: The figure shows a qualitative representation of the closeness of boundaries predicted by the proposed approach, ADWIN [16], CES [33] to ground truth boundaries from specific portions of EDUB-Seg (first row), and Huji (second row). Please see the text for details.

gain of 1.2%, 3% and 0.85% respectively with respect to AlexNet on UTE dataset (refer to Table 3.4).

Table 3.5 to 3.11 show the detailed F-Measure for photostream as well as the video sequence datasets. Tables show the δ, correlation coefficient threshold (ρ_c), and predicted segment for each video sample. We have used $\rho_c = 0.95$ for video datasets and $\rho_c = 0.999$ for the photo-stream datasets. Similarly, We set the δ for the corresponding granularity as 10^{-2}, 10^{-4}, and 10^{-6} respectively for video datasets and 10^{-1}, 10^{-3}, and 10^{-7} for photostream datasets.

3.3.5 Online Streaming vs Recorded Video

Our algorithm can be potentially used in the online streaming mode as well. Recall that for detecting a temporal boundary, we take a window w, split it at time instant t, in two windows w_1 and w_2, and then find the difference of means. Therefore, we effectively find the temporal boundary at t after looking at w_2 as well. This can be seen as detecting a

UTEgo dataset (video)				
SiD	Features used			
	AlexNet[85]	VGG16 [147]	Resnet101 [65]	GoogleNet[157]
P01	59.79	56.28	48.37	64.16
P02	59.01	61.57	60.82	55.80
P03	56.52	52.00	57.00	57.81
P04	58.33	68.66	70.85	68.00
Avg. Fscore	58.41	59.62	59.26	**61.44**

Table 3.4: F-Measure performance of our method on the features extracted from different pre-trained networks on UTEgo video dataset

HUJI dataset (video)					
Videos	F-score	δ	ρ_c	Pred.	GT
Yair	78.94	10^{-2}	0.95	37	38
Chetan	28	10^{-2}	0.95	4	5
Weighted Fscore	**73.01**				

Table 3.5: F-Measure performance of our method on HUJI video dataset

UTEgo dataset (video)					
Videos	F-score	δ	ρ_c	Pred.	GT
P01	59.79	10^{-4}	0.95	42	55
P02	59.01	10^{-6}	0.95	35	25
P03	56.52	10^{-4}	0.95	25	21
P04	58.33	10^{-4}	0.95	39	32
Avg. Fscore	**58.41**				

Table 3.6: F-Measure performance of our method on UTEgo video dataset

Disney dataset (video)					
Videos	F-score	δ	ρ_c	Pred.	GT
Alin Day 1	64.36	10^{-6}	0.95	54	32
Alireza Day 1	72.83	10^{-2}	0.95	86	77
Alireza Day 2	66	10^{-2}	0.95	131	72
Alireza Day 3	72.72	10^{-4}	0.95	32	33
Denis Day 1	68.42	10^{-6}	0.95	41	34
Hussein Day 1	65.67	10^{-2}	0.95	67	66
Michael Day 2	71.32	10^{-6}	0.95	77	65
Munehike Day 1	59.67	10^{-6}	0.95	68	57
Avg. Fscore	**67.63**				

Table 3.7: F-Measure performance of our method on Disney video dataset

HUJI dataset (Phtostream)					
Videos	F-score	δ	ρ_c	Pred.	GT
Yair	59.37	10^{-1}	0.999	27	38
Chetan	60	10^{-1}	0.999	7	5
Weighted Fscore	**59.44**				

Table 3.8: F-Measure performance of our method on HUJI photostream dataset

UTEgo dataset (Photostream)					
Videos	F-score	δ	ρ_c	Pred.	GT
P01	64.44	10^{-1}	0.999	45	55
P02	60	10^{-3}	0.999	29	25
P03	57.69	10^{-3}	0.999	32	21
P04	60.31	10^{-3}	0.999	35	32
Avg. Fscore	**60.61**				

Table 3.9: F-Measure performance of our method on UTEgo photostream dataset

Disney dataset (Photostream)					
Videos	F-score	δ	ρ_c	Pred.	GT
Alin Day 1	69.56	10^{-3}	0.999	38	32
Alireza Day 1	71.64	10^{-1}	0.999	68	77
Alireza Day 2	62.85	10^{-1}	0.999	72	72
Alireza Day 3	64.28	10^{-3}	0.999	24	33
Denis Day 1	69.84	10^{-3}	0.999	31	34
Hussein Day 1	73.33	10^{-1}	0.999	40	66
Michael Day 2	76.11	10^{-1}	0.999	65	65
Munehike Day 1	63.04	10^{-3}	0.999	44	57
Avg. Fscore	**68.83**				

Table 3.10: F-Measure performance of our method on Disney photostream dataset

EDUB-Seg20 dataset (Photostream)					
Subject-Set	F-score	δ	ρ_c	Pred.	GT
1-1	66.66	10^{-7}	0.999	28	16
1-2	45.71	10^{-7}	0.999	22	12
1-3	70.96	10^{-7}	0.999	17	13
1-4	70	10^{-7}	0.999	40	39
1-5	58.53	10^{-7}	0.999	43	38
2-1	64.615	10^{-7}	0.999	42	22
2-2	56.86	10^{-7}	0.999	67	34
2-3	60.46	10^{-7}	0.999	54	31
2-4	72.72	10^{-7}	0.999	49	38
3-1	71.23	10^{-7}	0.999	37	35
4-1	70.58	10^{-7}	0.999	21	12
5-1	64.70	10^{-7}	0.999	22	11
5-2	61.72	10^{-7}	0.999	36	44
5-3	62.22	10^{-7}	0.999	22	24
6-1	70.12	10^{-7}	0.999	42	34
6-2	69.69	10^{-7}	0.999	35	30
6-3	71.11	10^{-7}	0.999	39	50
6-4	56.75	10^{-7}	0.999	45	28
7-1	54.54	10^{-7}	0.999	70	28
7-2	60	10^{-7}	0.999	26	14
Avg. Fscore	**63.96**				

Table 3.11: F-Measure performance of our method on EDUB-Seg20 photostream dataset

Datasets	High	Medium	Low
UTEgo	2m19s	3m08s	3m29s
Disney	1m77s	2m54s	3m87s

Table 3.12: Latency analysis

boundary with a certain latency. Table 3.12 shows the average latency of our algorithm vs the average segment length in the video.

3.4 Conclusion

In this chapter, we have introduced a novel, principled, and theoretically justified technique for temporal segmentation of egocentric videos. We have adapted the univariate concept drift for i.i.d. data to multivariate correlated data using the adaptive windowing technique. We demonstrate the results on long videos as well as photo-stream datasets to prove the efficacy of the proposed approach. We have also shown that the adaptive windowing technique can generate superior results in video temporal segmentation when compared to the state-of-the-art deep CNN/LSTM models.

Chapter 4

Summarization and Personalized Summarization[1]

4.1 Introduction

Egocentric videos are often recorded in a hands-free mode to capture day long visual diaries from the first-person perspective. The captured videos are highly redundant and extremely shaky, making them difficult to watch from beginning to end, thus necessitating the use of summarization tools for their efficient browsing.

The objective of a video summarization algorithm is to create a compact yet comprehensive summary by selecting appropriate frames from an input video. The problem has been a well-studied area in computer vision with two styles for generated summary: *keyframes* and *video skims*. In the keyframes-based output, the summary is represented by a set of salient frames of the original video sequence. This is also called *still image abstract* or *static storyboard*. A video skim-based summary is generated as the collection of video segments extracted from the original video sequence. This is also called the *moving image abstract*, or *moving storyboard*. The focus of this paper is on generating video skims. Most of the work has targeted videos from static surveillance cameras [29, 152, 189]. The focus is not misplaced since surveillance videos form the majority among all kinds of videos captured and have long, uninteresting portions. This makes the use of video summarization attractive. However, from the algorithmic perspective, the summarization problem is much easier for surveillance videos and can be mostly done by subtracting static background and choosing frames with significant and important foreground objects.

The majority of the summarization techniques include predefined events/criteria such as action scenes and loud music for movies, anomaly detection in the surveillance video, and specific events in a sports video. On the other hand, videos from point and shoot cameras are typically triggered by user interest and do not have long uninteresting

[1]This work was done in collaboration with Anuj Rathore (IIIT Hyderabad) and resulted in two publications published in ACMMM and PAMI. This chapter includes the work titled "Generating Personalized Summaries of Day Long Egocentric Video" published in PAMI.

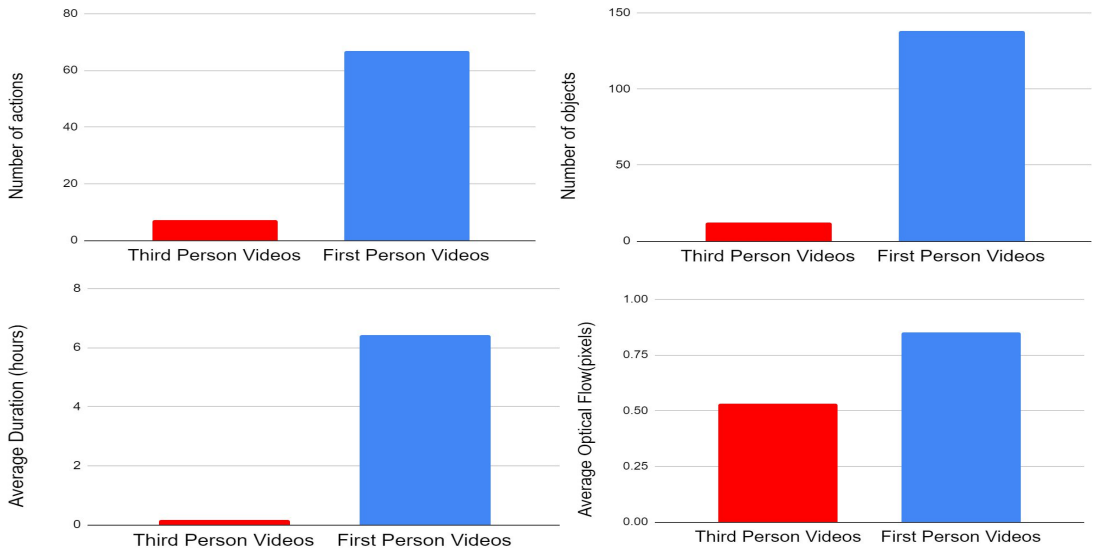

Figure 4.1: Egocentric videos are characterized by their long, redundant, and extremely shaky nature. The figure shows comparative statistics for benchmark egocentric and third person video. We use Disney, HUJI, and UTE datasets for first-person and TVSum and SumMe for third-person datasets to calculate the statistics. While other statistics are obvious, optical flow indicates frequent sharp changes in viewpoints due to the wearer's head motion. The typical characteristics make traditional summarization techniques unsuitable for egocentric videos.

portions. However, in a video captured using a moving camera, the background is also moving, and the task of determining which frames to include in a summary becomes much more challenging. Researchers have suggested various cues to select the summary frames such as motion [193], global image features [76, 109, 199], detecting important events, the presence of salient objects and people [91, 104], as well as role of a frame in a hypothetical storyline [162]. Most of these techniques give a score to each frame and then use a separate combinatorial algorithm [104, 173] to select the frames that maximize the score in a given summary length constraint. The major shortcomings of these techniques are in their pre-specified saliency definition, the restricted capability to model inter-frame interactions for global indicativeness of the summary and lack of scalability and customization for long videos.

The success of deep neural networks (DNNs) in learning complex frames and video representations has paved the way for supervised [76, 188] and unsupervised [109, 199] summarization techniques. Here, RNNs/LSTMs are typically used to model sequential dependency among frames. Given the numerical constraints on back-propagating gradients over many recurrent connections, such architectures cannot process input videos longer than a few hundred frames. Even hierarchical approaches [195] can handle up to

Methods	Unsup	Scalable	Customization			SR
			VL	US	Int	
K-Medoids	✓	✓	✓	✗	✗	✗
DR-DSN[199]	✓	✗	✓	✗	✗	✗
M-AVS[76]	✗	✗	✗	✗	✗	✗
dppLSTM[188]	✗	✗	✗	✗	✗	✗
FFNet[87]	✗	✗	✓	✗	✗	✗
SUM-GAN$_{dpp}$ [109]	✗	✗	✓	✗	✗	✗
Ours	✓	✓	✓	✓	✓	✓

Table 4.1: Comparison of SOTA techniques with the proposed method on various criteria important for applicability to egocentric videos. Abbreviations: Unsup = Unsupervised, VL: Variable Length, US: User Saliency, Int: Interactive, SR: Shake Resistance.

1600 frames only.

Egocentric videos contain extreme shakes and long uninteresting portions (see Fig. 4.1). The camera wearer often moves in a variety of scenes and performs various daily activities. These characteristics rule out techniques that rely on the detection of important pre-specified events or objects. Moreover, the task of obtaining annotated samples for summarization for third-person videos is hard. It is even harder for egocentric videos, which are often captured in enhanced privacy-sensitive scenarios. This rules out the supervised approach, rendering many SOTA techniques unsuitable [76, 87, 109, 188].

While generating visually diverse summaries, it is observed that the summarization criteria are inherently personal. Specifically, in the day long life-logging videos, the same user may want to explore the summary focusing on the different types of events like social interaction, having food, walking, etc. Hence, a key requirement of a summarization framework for egocentric videos is to personalize summaries by interactively collecting user feedback on the fly.

In this work, we formulate video summarization as a sequential decision making process over video frames, where each decision is binary (whether to include the frame in summary or not). The setup requires a sequential model to capture the temporal dependencies, which has been addressed using a bidirectional LSTM based architecture. The quality of the summary is available only for the whole set and not for individual frames. Hence we find the RL framework, which works with sparse rewards, suitable to solve this problem. Our experiments also show an ablation study with various RL optimization algorithms such as policy gradient, Q-Learning, and Actor-Critic styles. The key strengths of our approach are shown in Table 4.1. The specific contributions of our work are:

1. Our framework can work with arbitrary long input videos and can be trained to generate summaries of various lengths. We demonstrate it by generating 1, 5, 10,

and 15 minutes summaries of day long egocentric videos from several benchmark datasets [51, 90, 104, 125, 127].

2. Our approach can focus on various user-specified saliency criteria for the summary, such as distinctiveness, indicativeness, and object, or motion saliency.

3. We propose an interactive summarization framework that can personalize summaries based on the length, content as well as interactive feedback from the user.

4. We achieve state-of-the-art performance on benchmark egocentric video datasets. We report Relaxed F-score [33] of 29.60 against 19.21 from the `SOTA` [199]. We also report BLEU score of 11.55 from our approach in comparison to 10.64 by the `SOTA` on the Disney dataset [51].

5. Though our focus is on egocentric videos, our technique can summarize hand-held videos as well. We obtain F-score of 46.40 and 58.3 on SumMe [61] and TVSum [153] datasets respectively, against the `SOTA` scores of 41.4 and 57.6 respectively.

The first version of this work appears in [130] only demonstrates the naive RL framework, namely policy gradient, to summarize day long egocentric videos. The second version appears in [114] contains the following core contributions:

1. We propose an interactive summarization framework that can personalize summaries based on the feedback (video exemplars) provided by the user.

2. Advance RL frameworks, namely Q Learning and AC framework, are introduced with various plugins such as distinctiveness, indicativeness, and object or motion saliency.

4.2 Proposed Approach

The specific objectives of the proposed summarization approach are as follows:

1. **Unsupervised:** To handle enhanced privacy concerns.

2. **Scalable:** To handle day long egocentric videos.

3. **Customizable:** To handle vast variety of contexts in the *wild* egocentric videos.

4. **Interactive:** To accommodate user preferences.

To simplify the exposition, we first describe our architecture to generate summaries for short videos in an unsupervised manner. We then explain to scale-up of the architecture for day long videos, followed by the modifications required for customization and interactive summary generation.

4.2.1 Architecture

The proposed framework uses 3D convolutional neural networks (CNNs) for capturing spatio-temporal features from an egocentric video. We have used 3D CNN model [164], called *C3D* hereon, trained on Sports-1M dataset for feature extraction in our design. Other 3D CNN models such as [19, 75, 165] can be used as well. We first divide our video into *sub-shots* of 16 non-overlapping frames and extract 512 dimension features from pool5 layer: $\{x_t\}_{t=1}^{T}$ for each sub-shot from C3D. Here T denotes the total sub-shots extracted from a video. The extracted features are inputted to the reinforcement learning agent, which uses a bidirectional long short-term memory network (BiLSTM). The hidden state ($h_i = h^f \| h^b$) of BiLSTM encapsulates past and future information of i^{th} sub-shot using forward and backward stream respectively. Here h^f and h^b are hidden states of forward and backward layers of BiLSTM, respectively, and $\|$ indicates the concatenation of the two. We unroll the BiLSTM network M times for the training and give a sub-shot as input to each BiLSTM unit. .

4.2.2 Formulation

We formulate the summary generation as a Reinforcement Learning (RL) problem, where the state space comprises of input sub-shots features $\{x_m\}$, and the action set $\{a_m\}$ is a binary decision for selecting or not selecting a particular sub-shot in summary. To train the summarization agent, we experiment with the following RL optimization strategies: 1. Policy Gradient, 2. Q Learning, and 3. Actor-Critic.

Summarization with Policy Gradient: For the policy gradient framework, we design the agent as a BiLSTM network followed by a fully connected (FC) layer for final prediction as shown in Fig. 4.2. The BiLSTM takes C3D features $\{x_m\}_{m=1}^{M}$ as input and produces corresponding hidden states $\{h_m\}_{m=1}^{M}$. In the end, the FC layer is followed by a sigmoid function to predict the probability score $\{p_m\}_{m=1}^{M}$ corresponding to each sub-shot. The output summary corresponding to the input video is then selected by sampling each sub-shot based on the probability outputted by each LSTM unit. The reward for the agent is the score of the overall summary based upon the pre-specified or user-defined scoring functions as described later in Section 4.2.3, Section 4.2.5, and Section 4.2.6.

To train the summarization agent, we use the policy-based reinforcement learning to optimize the policy π_θ with parameter θ that maximizes the expected reward:

$$J^{\pi}(\theta) \;=\; \mathbb{E}_{\pi_\theta(a_{1:M}|h_{1:M})}\left[R(\mathcal{S})\right], \tag{4.1}$$

where $\mathcal{S}$ denotes the output summary. $\pi_\theta(a_{1:M}|h_{1:M})$ denotes probability distribution over the input sub-shots (M), where $a_m \in \{0, 1\}$ indicates whether the m^{th} sub-shot is selected or not. $R(\mathcal{S})$ is the reward function that measures the quality of generated summaries.

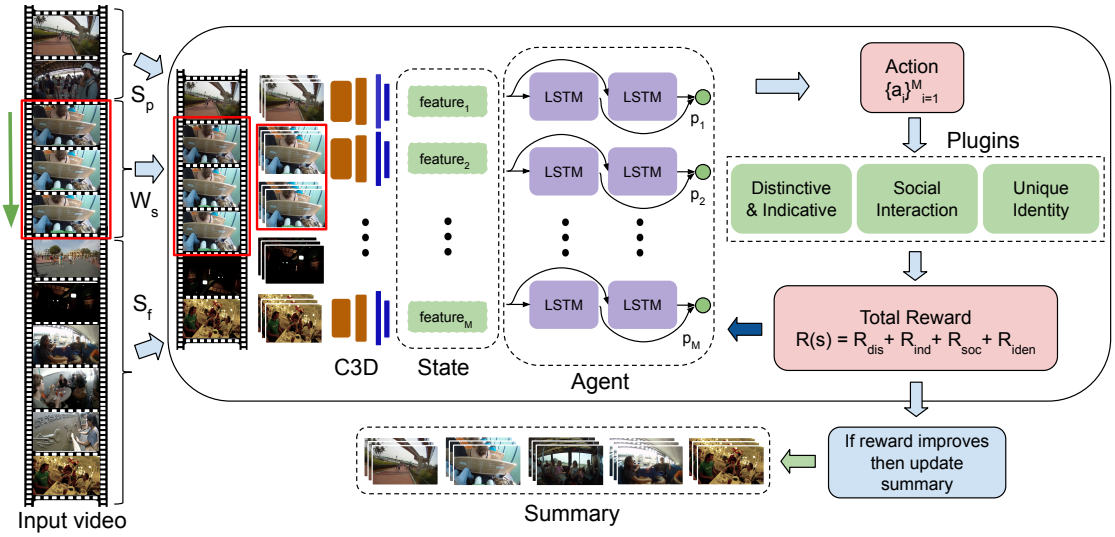

Figure 4.2: Illustration of the proposed technique to summarize day long egocentric videos based on policy gradient framework. The figure also demonstrates the sliding window framework. In that, as per the current position of the sliding window (W_s) we select a set of segments as a past summary (S_p) and future summary (S_f) (a global representative of input video) from the previously generated summary. The first column to the left of C3D shows the representation of past, current, and future segments of the video. The past and future segments are represented by their sub-shots in the current summary. Further, each sub-shot in the representation (whether coming from past, current, or future segments) is essentially a set of 16 consecutive frames from which we evaluate the C3D features. The second column to the left of C3D features indicates these sub-shots/sets. The RL agent takes actions on the input (S_p+W_s+S_f) to select the sub-shots for summary by maximizing the reward in each iteration. Based on various informative measures, the feedback reward R(S) assesses the goodness of the summary.

It can be shown that the derivative of objective function w.r.t. parameters θ is given as:

$$\nabla_\theta J(\theta) = \mathbb{E}_{p_\theta(a_{1:M})}\left[R(\mathcal{S})\sum_{m=1}^{M}\nabla_\theta \log \pi_\theta(a_m|h_m)\right], \tag{4.2}$$

where $p_\theta(a_{1:M})$ denotes probability distribution over possible action sequence. Since we calculate the expectation over the action sequence, which is difficult to compute directly. We approximate it by sampling actions for E episodes on the same input and output probability distribution and then calculate the average gradient:

$$\nabla_\theta J(\theta) = \frac{1}{E}\sum_{e=1}^{E}\sum_{m=1}^{M} R(\mathcal{S}_e)\nabla_\theta \log \pi_\theta(a_m|h_m), \tag{4.3}$$

where $R(\mathcal{S}_e)$ is the reward computed for summary S in the e^{th} episode. The high variability in cumulative reward and log probabilities make the network hard to converge. We use a common countermeasure to ensure smaller and stable gradient, which is to subtract a baseline, $\mathcal{B}$, from the cumulative reward:

$$\nabla_\theta J(\theta) = \frac{1}{E}\sum_{e=1}^{E}\sum_{m=1}^{M}(R(\mathcal{S}_e) - \mathcal{B})\nabla_\theta \log \pi_\theta(a_m|h_m) \tag{4.4}$$

where $\mathcal{B}$ is computed as the moving average of rewards experienced so far.

Policy gradient is a naive RL framework that uses the baseline function to calculate the episodic reward. The baseline functions are not learnable, which leads to high variance across video samples. We introduce the Q learning and AC framework that uses a Q value network that leads to a stable gradient across video samples. On the other end, the extra parameters required more training samples. For the proposed framework, each position of the sliding window (refer Section 4.2.4) constitutes one training sample, so we generate sufficient training samples (especially for day long videos) to train the Q learning and AC frameworks.

Summarization using Q Learning: In Q learning, instead of predicting the confidence score, p_m, we predict the Q values for selecting or not selecting a sub-shot for a particular state. The objective function of Q learning is to minimize the mean squared error between the target Q value and the approximate Q value with parameter θ over the input sequence:

$$J^\pi(\theta) = \mathbb{E}_\pi\left[\left(Q^\pi(s,a) - Q^\pi_\theta(s,a)\right)^2\right]. \tag{4.5}$$

Here $Q^\pi(s,a)$ and $Q^\pi_\theta(s,a)$ is the target Q value and approximate/predicted Q value respectively. As suggested in [158, Ch. 6], we use TD target to approximate the target Q values i.e $Q^\pi(s_m,a_m) = r + \gamma Q^\pi_{\theta^-}(s_{m+1},a_{m+1})$, where r is the current reward, γ is the discount factor, and $Q^\pi_{\theta^-}$ is the Q value of the target with parameters updated in the alternate epochs. With the approximation, the weight update is given by:

$$\Delta\theta = \alpha\delta_m \nabla_\theta Q^\pi_\theta(s_m,a_m), \tag{4.6}$$

where δ is the TD error computed as:

$$\delta_m = r + \gamma Q^\pi_{\theta^-}(s_{m+1},a_{m+1}) - Q^\pi_\theta(s_m,a_m) \tag{4.7}$$

We adopt the idea proposed by [113] to calculate the reward for 'm' steps of an episode, and calculate TD error for the entire episode as:

$$\delta_m = \sum_{m=1}^{M-1}\left[r_m + \gamma\ Q^\pi_{\theta^-}(s_{m+1},a_{m+1}) - Q^\pi_\theta(s_m,a_m)\right] \tag{4.8}$$

$$\delta_m = R(\mathcal{S}) + \gamma\sum_{m=1}^{M-1} Q^\pi_{\theta^-}(s_{m+1},a_{m+1}) - \sum_{m=1}^{M-1} Q^\pi_\theta(s_m,a_m) \tag{4.9}$$

where $R(\mathcal{S}) = \sum_{m=1}^{M-1} r_m$, is the total reward. And the weight update is given as:

$$\Delta\theta = \alpha\delta_m \sum_{m=1}^{M} \nabla Q_\theta(s_m, a_m), \tag{4.10}$$

where α is the learning rate for the parameters.

Summarization using Actor-Critic Framework: For the Actor-Critic framework, we propose a common BiLSTM network, with tied weights, followed by two separate fully connected layers for Actor and Critic as shown in Fig. 4.4. The common BiLSTM reduces the parameters and ensures fast convergence. The basic policy gradient in an actor-critic framework is given as follows:

$$\nabla_\theta J(\theta) = \mathbb{E}_{p_\theta(a_{1:M})}\left[R(\mathcal{S}) \sum_{m=1}^{M} \nabla_\theta \log \pi_\theta(a_m|h_m)\right] \tag{4.11}$$

The actor policy is denoted by π_a and its parameters θ are updated as follows:

$$\Delta\theta = \alpha_a \sum_{m=1}^{M} Q_c(s_m, a_m)\nabla_\theta \log \pi_a(s_m, a_m), \tag{4.12}$$

where Q_c is the Q-value for the state-action pair given by the critic, and α_a is the learning rate of the actor.

Denoting critic parameters by w, we update the critic parameters using TD target and calculate the TD error in the same way as done for Q learning:

$$\delta_m = R(S) + \gamma \sum_{m=1}^{M-1} Q_{w^-}(s_{m+1}, a_{m+1}) - \sum_{m=1}^{M-1} Q_w(s_m, a_m). \tag{4.13}$$

where Q_{w^-} indicates the Q value returned by the critic for the target. With the TD error computed as above, the weight update for the critic is given by:

$$\Delta w = \alpha_c \delta_m \sum_{m=1}^{M} \nabla Q_w(s_m, a_m) \tag{4.14}$$

4.2.3 Scoring a Summary and Basic RL Rewards

The proposed RL framework requires a summary scoring mechanism to compute the goodness of a summary. The goodness of the summary can be defined by selecting the most diverse sub-shots that can reproduce the original video with minimal loss. To implement this notion, distinctiveness and indicativeness rewards are used in literature [62, 199]. This score is used as a reward to train the agent using any of the training methodologies (policy gradient, Q learning, or actor-critic) discussed in the previous section. Though we describe many rewards to customize the summaries in the next section,

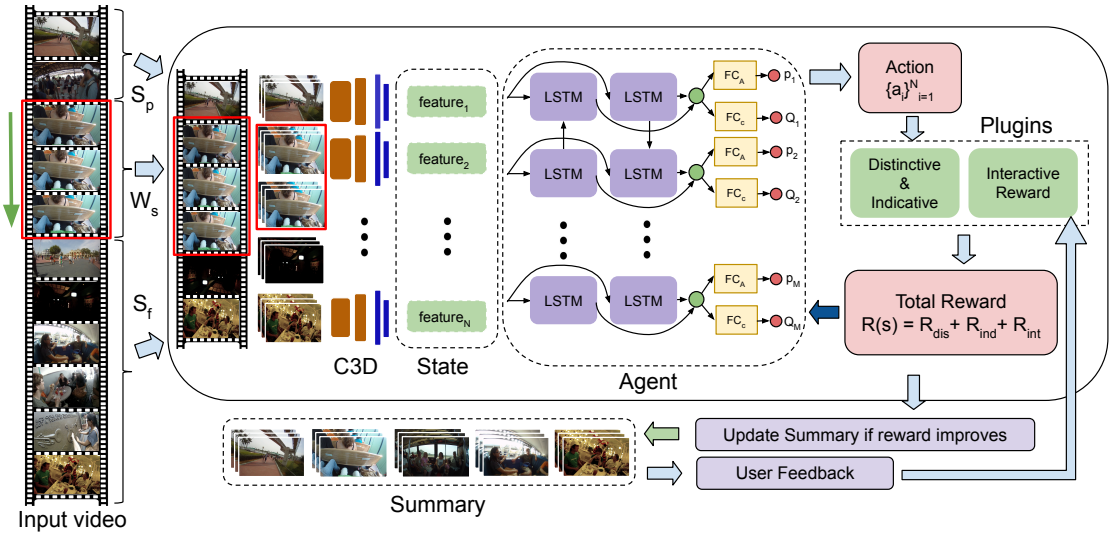

Figure 4.4: Illustration of the proposed framework using Actor-Critic framework along the interactive summarization plugin to summarize day long egocentric videos. After generating the initial summary as described in the last few sections, we ask the user to pick the sub-shots which the user certainly wants in summary. We call such sub-shots positive sub-shots. Similarly, we collect in negative sub-shots, the sub-shots which the user dislikes.

three basic rewards are common to all the summaries produced by our framework. Note that all these rewards do not require the notion of any pre-specified important objects or events.

Distinctiveness Reward: Let $\mathcal{V} = \{1, \ldots, M\}$, represents the set of input sub-shots, and $\mathcal{S} = \{i \mid i \in \mathcal{V}\}$ denotes the set of indices of the sub-shots included in the summary (hereinafter called *summary sub-shots*). Let x_m be the feature representation of m^{th} sub-shot. *Distinctiveness* reward measures the degree of uniqueness among the summary sub-shots, and is computed as the mean of pairwise distance among the selected video sub-shots using ℓ_2 norm:

$$R_{\text{dis}} = \frac{1}{|\mathcal{S}|(|\mathcal{S}|-1)} \sum_{i \in \mathcal{S}} \sum_{\substack{j \in \mathcal{S}, \\ j \neq i}} \|x_i - x_j\|_2 \tag{4.15}$$

Indicativeness Reward: The indicativeness reward measures how well the summary sub-shots represent the original input video. Here the assumption is that each input sub-shot can be described as a linear combination of a small subset of indicative sub-shots.

Hence, we define R_{ind} as:

$$R_{\text{ind}} = -\frac{1}{|V|}\sum_{i\in V}\min_{\mathbf{b}}\left(x_i - \sum_{j\in\mathcal{S}} b_j^i x_j\right)^2, \tag{4.16}$$

where V indicates the set of input sub-shots in the whole video and each variable b_j^i denotes the weight corresponding to sub-shot x_j in the summary, to best reconstruct an input sub-shot x_i. The set of weights $\mathbf{b} = \{b_j^i\}$ are found as the ones maximizing the indicativeness reward for a summary set $\mathcal{S}$.

Summary Length Reward: A trivial way to generate a summary that maximizes distinctiveness and indicativeness is to choose all the input sub-shots in the output summary. To prevent such a trivial solution and keep the summary concise, we introduce an additional constraint penalizing the length of the summary. We propose the following reward to generate a summary of the desired length:

$$R_{\text{length}} = -\left(\frac{1}{M}\sum_{m=1}^{M} p_m - \epsilon\right)^2, \tag{4.17}$$

where p_m denotes the probability outputted by our framework for selecting sub-shot m, and ϵ denotes the desired percentage of sub-shots (given as input to our system) to be selected in the summary.

4.2.4 Scalability to Day Long Egocentric Videos

The proposed technique, as described above, does not require the input sub-shots to be temporarily adjacent. Therefore, to scale it to long videos, instead of giving the whole video as input in one go, we use a sliding window approach (refer Fig. 4.2 or Fig. 4.4). We keep on moving a sliding window (containing temporally adjacent sub-shots), and at any temporal location, we give two sets of input to our model. The first input is 'all the sub-shots' covered by the current window, and the second is the most recently generated 'indicative sub-shots' (or the latest summary generated by our method minus the indicative sub-shots belonging to the current window). Note that we do not give the indicative sub-shots belonging to the current window since the current iteration will update which sub-shots will be selected as indicative sub-shots from the current window. However, the set of indicative sub-shots which do not belong to the current window remains as is. Further, note that our technique can choose any number of sub-shots as indicative from the current window based on the accrued reward. We divide the indicative sub-shots into $\mathcal{S}_p$ and $\mathcal{S}_f$ according to the current position of the sliding window, i.e., all the indicative sub-shots indexed before the sliding window belong to $\mathcal{S}_p$, and all the sub-shots indexed after sliding window belong to $\mathcal{S}_f$. We use the model described in the previous section to pick the most distinctive and indicative sub-shots with these two inputs.

Based on the trained weights, the network outputs probability scores corresponding to each sub-shot. We choose an action sequence of top-scoring sub-shots based on these probability scores to match the desired summary length. We compute the reward in feature space over the action sequence and back-propagate the gradient as per one of the RL techniques viz Policy Gradient, Q Learning, or Actor-Critic. Further, if the selected sub-shots get a better reward than the previous summary, we update the 'indicative sub-shots' of the video according to the current selection. The updated representation is then used in the next pass for the next sliding window, and the same process is repeated for all sliding windows of the video. We move the sliding window from the beginning to the end of any day long egocentric video. We call this one scan, and then we repeat this multiple times to better assimilate the information from all parts of the video. Furthermore, we observed no significant systematic bias in the output summary due to the initialization because of multiple scans.

The proposed framework is visually described in the Fig. 4.2 and Fig 4.4 for summarization and interactive summarization, respectively, and can work with arbitrarily long videos while still maintaining the global context for generating a consistent and concise summary.

4.2.5 Customizing Summaries

The unconstrained nature of egocentric videos makes it hard to pre-suppose the saliency criteria. We propose a plugin-based architecture where different plugins can bias the generated summaries using appropriate rewards. Apart from distinctiveness and indicativeness, we propose following two novel rewards, especially for the first-person context:

Social Interaction Reward: We propose a new reward emphasizing the social interactions present in egocentric videos. We integrate a FasterRCNN [131] model, fine-tuned for face detection, into the proposed network. We detect faces in each frame included in the summary and, add the ratio of faces in the summary to the length of the summary, as the reward. We observe that, during social interaction faces tend to occupy a larger area ($\text{face}^{\text{area}}$), and also have higher prediction confidence score ($\text{face}^{\text{conf}}$). The smaller faces with low confidence are usually far away from the wearer and are irrelevant from a social interaction perspective. Therefore, we threshold the bounding box area and confidence score, to eliminate the faces with no social interaction with the wearer. With this, we define social interaction reward as:

$$R_{\text{soc}} = \frac{\sum_{m \in \mathcal{S}} \text{face}_m^{\text{soc}}}{|\mathcal{S}|}, \quad \text{where}$$

$$\text{face}_m^{\text{soc}} = \begin{cases} 1, & \text{if } \text{face}_m^{\text{conf}} > 98\%, \text{ and } \text{face}_m^{\text{area}} > 4\% \\ 0, & \text{otherwise} \end{cases} \tag{4.18}$$

Face Identity Reward: We suggest this reward to generate a summary focusing on 'unique' interactions present in a video sequence. To evaluate this reward, we compute

OpenFace [7] features of the faces detected by FasterRCNN. However, apart from the usual distinctiveness and indicativeness reward on sub-shot features, we propose an additional reward for the distinctiveness of face features:

$$R_{\text{iden}} = \frac{1}{|\mathcal{S}|(|\mathcal{S}|-1)} \sum_{i \in \mathcal{S}} \sum_{\substack{j \in \mathcal{S}, \\ j \neq i}} \left(1 - \frac{f_i^T f_j}{\|f_i\|_2 \|f_j\|_2}\right) \tag{4.19}$$

where f_i corresponds to the facial features from the i^{th} sub-shot. The reward biases generated summary towards including all the people, with whom a wearer might have interacted within the video.

Customizing Summary Length: It is hard to predict the amount of important content in a day long egocentric video. Therefore, we propose to generate summaries of different lengths to cater to various kinds of content. Since our model is completely unsupervised, we merely need to change the desired percentage of sub-shots (epsilon) and retrain the network to output different length summaries. In the experiments section, we demonstrate the capability by outputting summaries of 1, 5, 10 and 15 minutes for hours long videos. Apart from showing the adaptability of the proposed model, the summaries also demonstrate how well the proposed technique select content at different granularity from the input videos.

4.2.6 Interactive Summarization

The variety of contexts in which an egocentric video is captured ensures that, despite the various customization proposed for the summary generation in the last few sections, a user may still find some interesting portions not included or some redundant portions included in the summary. Therefore, we propose to introduce a new module in our framework that can interact with the user in an online manner and personalize the summaries by collecting the feedback provided by the user as depicted in Fig. 4.4.

After generating the initial summary as described in the last few sections, we ask the user to pick the sub-shots which the user certainly wants in summary. We call such sub-shots positive sub-shots($\mathcal{S}_+$). Similarly, we collect in negative sub-shots($\mathcal{S}_-$), the sub-shots which the user dislikes. Kindly refer to the section A.2 in appendix for the verbatim text transferred to the subjects for the user study. Based upon the sets of positive and negative sub-shots, we define the *interactive* reward as follows:

$$\begin{aligned} R_{\text{int}} = {} & \frac{A}{|\mathcal{S}||\mathcal{S}_+|} \sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{S}_+} \frac{x_i^T x_j}{\|x_i\| \|x_j\|} \\ & + \frac{B}{|\mathcal{S}||\mathcal{S}_-|} \sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{S}_-} \left(1 - \frac{x_i^T x_j}{\|x_i\| \|x_j\|}\right) \end{aligned} \tag{4.20}$$

where A, and B are the weights to fine-tune the impact of the user feedback. We use the interactive reward just as the other rewards in our RL based summarization framework.

4.3 Experiments & Results

4.3.1 Datasets

We demonstrate the results on Disney [51], UT Egocentric (UTE) [90, 104], HUJI [125, 127], SumMe [61] and TVSum [153] datasets. Disney, UTE, and HUJI are long duration egocentric video datasets. Disney consists of videos captured at Disney World by six individuals for three days. Here, we have merged the small video segments, following the numbering order provided by their authors, into a day long video for each individual. After merging, we have eight sequences of 4 to 8 hrs for each individual. For Disney, Yeung et al. [180] have provided ground truth text and video summaries of three videos, namely 'Alin Day 1', 'Alireza Day 1' and 'Michael Day 2' by three annotators. UTE comprises four videos, each of 3 to 5 hrs long, and captured in an unconstrained setting. To evaluate the proposed approach on UTE, we have used the annotations provided by Yeung et al. [180]. The HUJI dataset comprises 44 egocentric videos of less than 30 minutes each and captures daily activities performed by three subjects, both indoor and outdoor. HUJI dataset do not have any ground truth summaries (neither text nor video).

SumMe and TVSum are benchmark datasets containing small-duration video sequences. SumMe consists of 25 video sequences ranging from 1 to 6 minutes videos of various domains such as sports, holidays, etc., in both third person and egocentric perspectives. It is annotated by 15 to 18 individuals with multiple summaries. TVSum contains 50 video sequences of 2 to 10 minutes, covering news, documentaries etc. It is also annotated by 20 persons with multiple summaries.

4.3.2 Evaluation Methodology

To prove the efficacy of the proposed framework, we use four evaluation measures. We observe that egocentric videos are highly redundant, especially in a temporal neighborhood. Therefore, picking any of the frames from a local neighborhood leads to perceptually similar summaries. However, the commonly used F-score [188] for evaluating summary does not capture this aspect, leading to arbitrary scores with little perceptual correlation. In the first evaluation measure, we use the metric proposed by Molino et al. [33], called *Relaxed F-score* (RFS). In Relaxed F-score, given a pair of predicted summary, $\mathcal{S}$ and ground truth summary, $\mathcal{G}$; instead of taking exact overlap, we take a fixed temporal relaxation (Δt) around $\mathcal{G}$, while calculating true positive (TP) and then remove these frames from the false positive (FP) and false negative (FN) calculations. The relaxed precision (P_r), recall (R_r) and F-score (F_r) are defined as:

$$P_r = \frac{\text{Relaxed TP}}{\text{Relaxed TP} + \text{FP}} \quad \text{, and} \quad R_r = \frac{\text{Relaxed TP}}{\text{Relaxed TP} + \text{FN}}$$

$$F_r = \frac{2 \times P_r \times R_r}{P_r + R_r} \times 100\% \tag{4.21}$$

For long sequence egocentric videos, the semantic information can be more accurately expressed in texts [180]. Therefore, in the second evaluation measure, we perform the natural language description based evaluation of video summaries as proposed by [180]. We convert the predicted summary to text using the text description provided for the entire video by [180] and then use BiLingual Evaluation Understudy (BLEU) [141] score for evaluation.

In the third evaluation named Average Human Rating (AHR), we follow [106, 107, 116] to rate the summary based on *informativeness* and *enjoyability* with a confidence score by 10 participants. The participants were recruited using *purposive sampling* [163], where the participants have a different background, with three of them having professional experience in recording videos. The demographic information about the participants is given in Table A.3 in the appendix. The *informativeness* and *enjoyability* of each participant are weighted by the normalized confidence score, and the average over participants is reported. Please refer to section A.2 in the appendix for the detailed verbatim text transferred to the subjects for the user study.

In the last evaluation measure, we score the generated summary based on the number of unique events captured and the jerks present. To calculate the unique events, we have used the text description of the input videos (three videos of the Disney dataset) provided by Yeung et al. [180]. The consecutive sentences are merged if the BLEU score between them is greater than 0.5. Each unique sentence is then identified as a unique event. To calculate the number of jerks, we count the number of temporally discontinuous shots in the summary. The final score is calculated as:

$$\text{Score}_{ue} = \text{Unique Events} - \alpha_j \times \text{Number of Jerks} \tag{4.22}$$

where α_j is weight to penalize unique events by the number of jerks. We use $\alpha_j = 0.3$ in our experiments.

For small duration video datasets, we follow [188] and use traditional F-score to measure the quality. Note that the traditional F-score can also be seen as a special case of Relaxed F-score (RFS) with temporal relaxation of 0. For SumMe and TVSum, we generate a summary ($\mathcal{S}$) which is 15% of original video length, and report the mean F-score generated from multiple ground truth summaries.

As suggested by [32], we did a qualitative evaluation of personalized summaries in two scenarios by 10 participants. In the first scenario, a participant was asked to evaluate a system-generated summary while being unaware of the video content. Here, the system iteratively personalized the generated summary by taking into account the participant's feedback. In the second scenario, we assume that the user is aware of the video content (*e.g.*, the user may be the camera wearer) *a priori*. Please refer to section A.2 in the appendix for the detailed verbatim text transferred to the participants for the user study. Once the personalized summary is generated, then the participants rate the summary by the quality of personalization compared to the default summary on the Likert scale (1: very poor, 2: poor, 3: ok, 4: good, 5: excellent) along with their confidence score (1: Not confident to 5: Completely confident).

Methods	Alin				Michael				Alireza			
	RFS	BLEU	AHR		RFS	BLEU	AHR		RFS	BLEU	AHR	
			INF	ENJ			INF	ENJ			INF	ENJ
Uniform samp.	20.60	0.76	2.95	1.91	17.23	0.69	2.62	1.64	17.05	0.56	2.48	1.65
K-medoids	22.08	0.74	2.82	2.53	17.73	0.71	2.32	2.22	17.84	**0.57**	2.68	2.28
dppLSTM[188]	10.87	0.63	2.42	2.68	20.13	0.58	2.73	2.01	15.80	0.44	3.12	2.50
DR-DSN[199]	11.44	0.76	2.53	2.75	16.30	**0.74**	2.63	2.86	16.79	0.53	2.44	3.04
FFNet[87]	19.18	0.59	1.91	1.91	19.76	0.70	2.80	2.88	18.52	0.26	2.33	2.62
SUM-GAN[109]	12.27	0.53	1.17	2.26	16.53	0.64	2.14	2.48	14.14	0.41	3.18	2.78
Ours_{PG}	32.59	0.72	2.88	**3.22**	25.40	**0.74**	2.86	2.75	**27.65**	0.54	2.68	3.17
Ours_{Q}	30.38	**0.77**	3.26	2.66	23.89	0.72	2.93	**3.00**	23.89	0.56	3.46	**3.55**
Ours_{AC}	**35.65**	0.74	**3.68**	2.74	**30.00**	0.73	**3.46**	2.95	23.16	**0.57**	**4.06**	2.90

Table 4.2: Performance comparison between SOTA approaches and the variations of the proposed method. PG, Q, AC show our framework trained with Policy Gradient, Q Learning, and Actor-Critic learning techniques, respectively.

4.3.3 Implementation details

After experiments with a few different sizes, we set sliding window lengths to 25 percent of the desired summary length (please refer A.1 in appendix for detailed abalation study). For all the frameworks, we set the learning rate (α) to 10^{-5}, learning rate decay to 0.1, number of episodes to 5, number of sliding window pass per video to 4, ϵ to 0.5, hidden units in the BiLSTM to 256, and mini-batch size to 16. We set the discount factor (γ) to .99 for Q learning and AC framework. The actor (α_θ) and critic (α_w) learning rate are set to 10^{-3}. The maximum epochs used to train the network is 20. We also add l_2 regularization on the weights to avoid overfitting.

The proposed technique is implemented in PyTorch and tested on a regular workstation containing Nvidia Quadro P5000 GPU. It takes approximately 2 hrs (inclusive of feature extraction) to summarize an 8 hrs long video. The GPU memory required to generate a 5 minutes summary is approximately 1500MB.

4.3.4 Results on Long Egocentric Videos

Table 4.2 shows the quantitative evaluation between SOTA approaches and the variations of the proposed method on the three samples of Disney dataset. We compare various performance measures such as Relaxed F-score (RFS) with the temporal relaxation of 50 units (RFS-50), BLEU score, and Average Human Rating (AHR). For comparison with DR-DSN [199], we unroll the network for the whole video at the test time and generate the probability of picking each frame. Top scoring frames according to the summary length are then outputted as the summary. We notice significant performance improvement over all the SOTA approaches. We report an average of 10% improvement

against DR-DSN [199] in relaxed F-score for 50 units of temporal relaxation for three videos of the Disney dataset. We perform only marginally better in terms of BLEU score because, for many events, the text description of visually different events overlapped. For example, *"My friends and I walked through the park"* and *"My friends and I walked through the line"* are two visually different events but exhibit close BLEU score. Hence, even if our technique picks more unique events, the BLEU score is marginally better. However, the AHR shows significant performance improvement for all the videos in terms of *informativeness* and *enjoyability* score. The `SOTA` approaches typically pick a cluster of frames in summary from the same location (refer Fig. 4.7), which lowers the *informativeness* and *enjoyability* score compared to the proposed framework. The same is validated through our user study where one of the participants expressed for 'Alin Day 1' video when FFNet [87] is used,

> *"Kept focussing on scenes for far too long and because of this, it missed many other scenes. For example, lunch and dinner sequences were longer than required."*

Similarly, the summaries generated by uniform sampling and K-medoids, show sudden changes that lead to poor comprehension and reduces the *informativeness* and *enjoyability* score. The following quote from one of our participants (for 'Michale Day 2' video when 'uniform sampling' is used) supports the finding:

> *"Informativeness: I could not make sense of the whole summary as it felt more like a slide show of images. Although most of the events were included as compared to ground truth, still I reduced my score as I felt that multiple pics (frames) were depicting one event, which could be avoided given the slow rate and the fact that few frames were not adding any new information. Enjoyability: I did not enjoy this! It was not smooth and felt like I am watching a slide show of images. It was so slow and boring! "*

Table 4.3 shows the summary score for the unique events covered by 1, 2.5, and 5 minutes summaries. The numbers show that the proposed approach significantly improves compared to all the `SOTA` approaches for all cases except for one case of where uniform sampling performing better for 'Alireza Day 1' video when the summary length is 2.5.

In Fig. 4.5, we compare various `SOTA` approaches based on Relaxed F-score for various amounts of temporal relaxation (Δt). As we increase the relaxation, the Relaxed F-score increases linearly for all the methods, and from the graph, it is evident that our techniques outperform `SOTA` approaches by a huge margin for all relaxations.

Fig. 4.6 shows a qualitative comparison between DR-DSN [199] and the summaries generated by our method. The 1st row in the figure shows the original frames, and the numbers on the top show frame numbers (from **140Kth** frame to **300Kth** in the original video. The 2nd row shows the predicted summary frames by the DR-DSN method. The

Methods	1 minute			2.5 minutes			5 minutes		
	Al	Mi	Az	Al	Mi	Az	Al	Mi	Az
Uniform samp.	21	30	27	40	52	**60**	38	56	70
K-medoids	25	28	27	32	48	46	19	49	66
FFNet[87]	21.4	14.4	10.9	20.5	43	4.7	13.3	0.5	6.7
DR-DSN[199]	17.5	21.5	20.2	19.1	15.7	22.8	5.2	14.4	20.9
Ours_{PG}	27.6	28.9	31.1	48.6	57.6	49.9	41.2	58.5	63.1
Ours_{Q}	28.4	**43**	30.9	42.2	66.6	48.6	56.6	62.5	69
Ours_{AC}	**33.7**	33	**33.4**	**57.7**	**74.8**	56.6	**70.4**	**99.9**	**75.2**

Table 4.3: Performance comparison between `SOTA` and the variations of the proposed method for the number of unique events covered. We demonstrate the results for 1, 2.5, and 5 minute summaries on the three samples of the Disney dataset using basic rewards (distinctiveness, indicativeness, and summary length).

Subjects	Video Name	Dataset	Events		Score (1 to 5)
			Included	Excluded	
S01-S1	Alin	Disney	'Dinner'	'In Dark'	3
S03-S1	Alin	Disney	'Dinner'	'Tram ride'	5
S02-S2	P01	UTE	'Driving'	'Prep. Food'	4
S01-S2	Yair	HUJI	'Driving'	'Sitting'	4

Table 4.4: The table shows the Likert score when specific events are included or excluded in summary. S0X-SY represents subject 'X' in scenario 'Y'.

3rd, 4th, and 5th rows show output from the proposed method using distinctiveness-indicativeness, social interaction, and unique identity based rewards, respectively. We observe that, due to the specific rewards used, the summaries generated by our technique ignore the video segments like approaching the building, walking over the pool, etc., which do not involve social interaction or faces. The summaries are correctly centered towards their desired objective.

We observed in Fig. 4.6 that DR-DSN [199] picks a cluster of frames from a particular location in summary. On the other hand, our distinctiveness and indicativeness reward is able to distribute the summary frames from all over the video correctly. Fig. 4.7 gives a better visualization by showing the distribution of the summary frames with respect to the ground truth summary for various frameworks, including ours for the full video. The figure also indicates that most of the selected summary frames are common despite using different RL frameworks as the reward is the same for all the frameworks.

In Fig. 4.8, we compare 1 minute, 5 minutes, 10 minutes and 15 minutes summaries generated by our framework using the policy gradient method for the 'Michael Day 2'

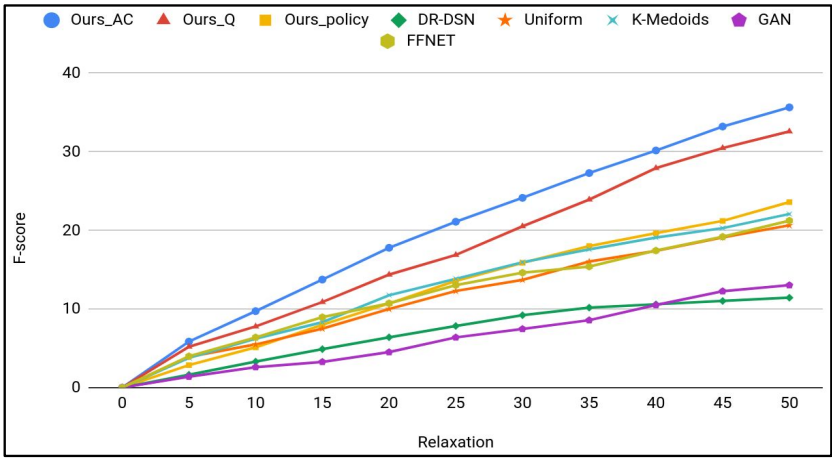

Figure 4.5: Commonly used F-score do not correlate well with goodness of a summary for long videos. We use Relaxed F-score to evaluate the summaries. The plot above shows Relaxed F-score for different units of temporal relaxation (Δt) for 'Alin Day 1' video sequence of Disney dataset.

sequence of the Disney dataset. Similarly, Fig. A.5 in the appendix compares different length summaries generated by the Actor-Critic framework for the 'P04' sequence of the UTE dataset. As can be seen, our network can adapt to different desired summary lengths. We observe, and as expected, most of the frames present in the shorter summaries are also present in, the longer ones along with some additional frames.

Fig. 4.9 shows the qualitative analysis of the interactive summarization using Interactive Summarization reward along with the basic RL rewards. From the visualization, it is evident that the summary is indeed biased towards user feedback. Similarly, Fig. A.2 in appendix demonstrates the interactive summarization framework on the 'P01' video sequence of the UTE dataset.

The UTE dataset comprises small video sequences ($<$ 5 hrs) and is less complex than the Disney dataset. Due to the aforementioned reason, Table 4.5 shows significant improvement over `SOTA` in terms of RFS-50 measure for all the UTE videos.

Table 4.4 shows the results from a user study as discussed in the evaluation section. The detailed results with the comments for all 10 participants are shown in the Table A.2 in the appendix. It is evident that the users like personalized summaries generated by our method.

4.3.5 Results on Short Hand-held Videos

Though not the focus of this paper, we also evaluate our method over short hand-held videos. Table 4.6 shows the comparison. Our method outperforms all unsupervised

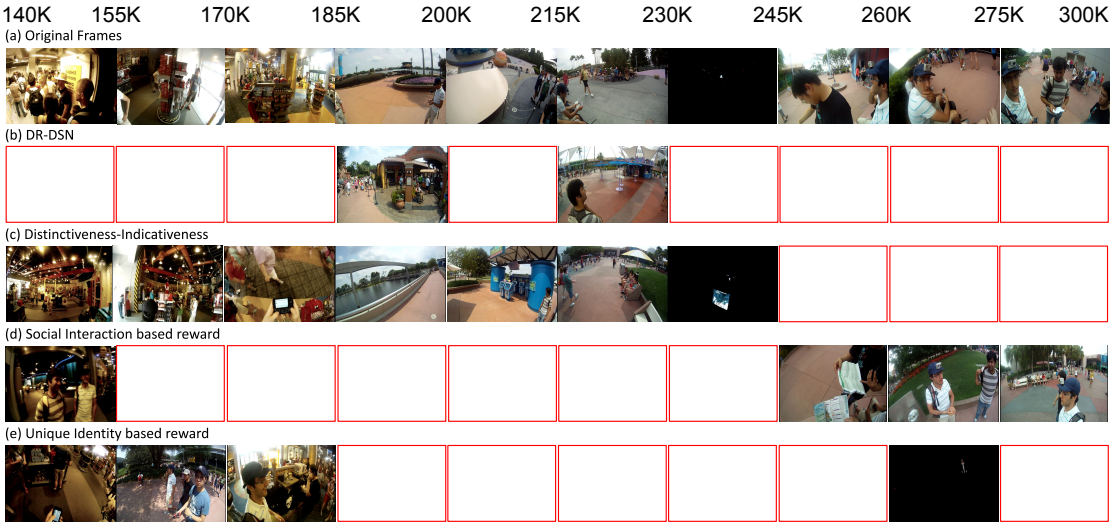

Figure 4.6: The figure shows a comparison between DR-DSN [199] and proposed approach for the **10** minutes summaries of 'Michael Day 2' sequence using basic RL rewards. The blank rectangles indicate that no frames were picked from those frame ranges.

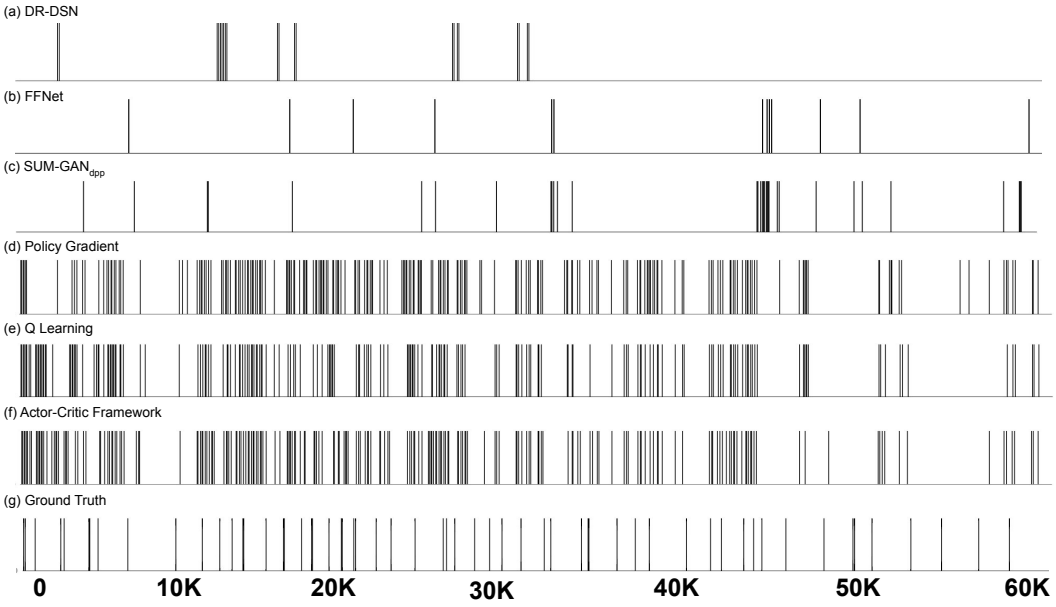

Figure 4.7: We also observe in our experiments that the SOTA often gets biased towards a short temporal segment in the video. In contrast, ours can distribute the summary frames from all over the video same as ground truth.

methods. Though the proposed method is unsupervised and comparison with supervised techniques may not be fair. We still made a comparison and except for H-RNN [195] and M-AVS [76], where we perform close, our method improved SOTA supervised techniques as well.

Comparing the performance of three configurations of our technique corresponding to different RL optimization techniques, we observe that Q learning performs better than the policy gradient, and the actor-critic performs better than Q learning. The policy gradient uses a baseline function that reduces the cumulative reward variance and leads to smaller gradients. In contrast, the Q learning and actor-critic techniques use a Q-value network instead of a baseline function to calculate TD error. This ensures higher gradients across multiple video samples, leading to better and faster reward

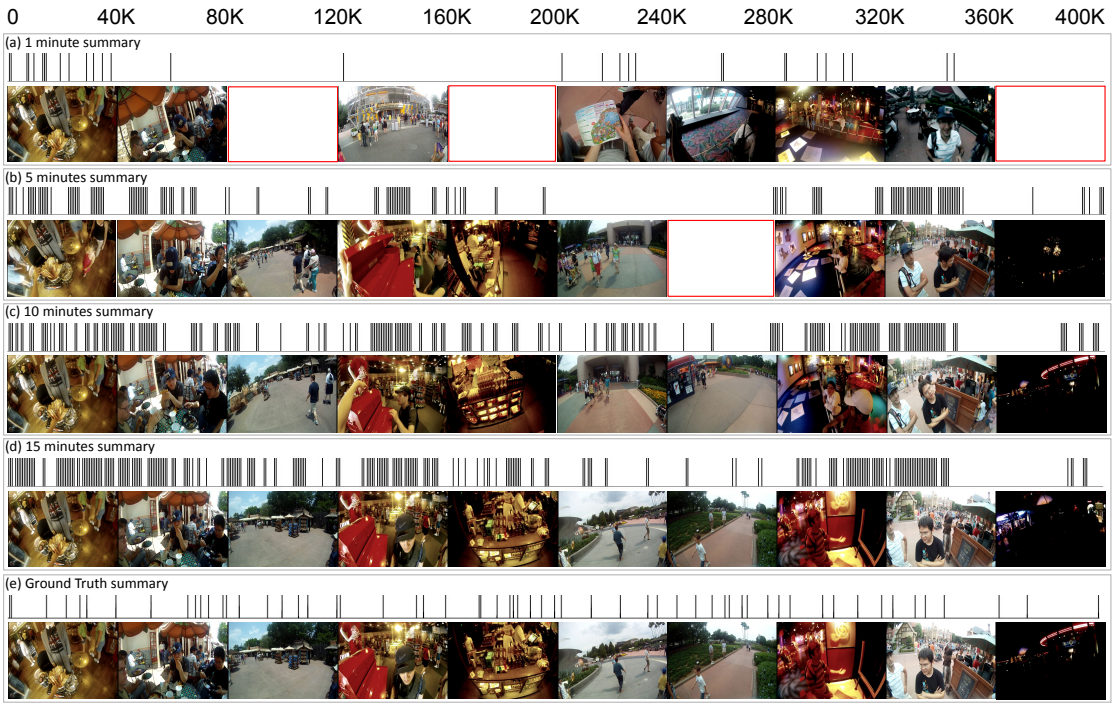

Figure 4.8: Comparing 1, 5, 10, and 15 minutes summaries (row 1-4) based on the basic RL rewards using Policy Gradient framework on 'Michael Day 2' sequence from Disney dataset with the ground truth summary (row 5). Note that the ground truth summary length is approximately 5 minutes. The numbers on the top show frame numbers (from 0 to 400K). The pictures show indicative frames in summary from the corresponding frame range. The blank rectangles indicate no frames were picked from those frame ranges. The black vertical bars indicate a frame was picked from a corresponding temporal window of 70 frames in each row. The bar serves to indicate the distribution of summary frames in the video.

Methods	P01	P02	P03	P04
Uniform samp.	27.78	25.11	36.56	20.79
K-medoids	30.50	22.86	39.66	22.59
FFNet [87]	30.78	19.37	35.92	27.43
SUM-GAN$_{dpp}$ [109]	31.68	10.91	35.85	25.44
dppLSTM [188]	32.47	26.78	41.66	26.93
DR-DSN [199]	36.36	28.21	42.54	27.81
Ours$_{pol}$	43.64	46.39	51.16	39.41
Ours$_{Q}$	41.94	**48.24**	48.47	39.65
Ours$_{AC}$	**47.50**	36.26	**58.86**	**48.10**

Table 4.5: Comparison on UTE dataset based on basic RL rewards using RFS-50 metric.

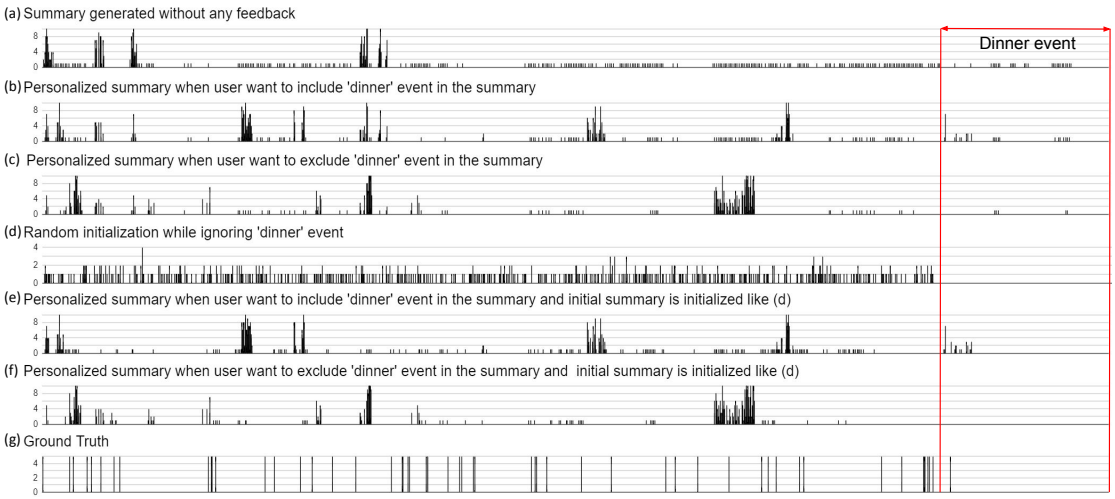

Figure 4.9: The figure demonstrates the visualization of the interactive summarization of the 'Alin Day 1' video sequence of the Disney dataset for 10 minutes summaries. Each bar represents 10 seconds of a time interval. (a)-(f) shows different summaries when the user asks to exclude/include 'dinner' event in summary, and (g) shows the ground truth summary distribution. We observe that (b) shows big peaks in the 'dinner' event area, whereas (c) shows very few spikes because of the negative feedback. As an ablation study, we initialized the summary by random frames but not included any frame from the 'dinner' event in the initialization, as shown in (d). When we personalized the summary to include the 'dinner', with the initialization as done in (d), we observe that the summary changes to select sub-shots from the 'dinner' event as shown in (e).

Methods	SumMe	TVSum	Category
dppLSTM[188]	38.6	54.7	supervised
SUM-GAN$_{sup}$ [109]	41.7	56.3	supervised
DR-DSN$_{sup}$ [199]	42.1	58.1	supervised
Li et al. [94]	43.1	52.7	supervised
M-AVS [76]	44.4	**61.0**	supervised
H-RNN [195]	44.3	**62.1**	supervised
Uniform samping	29.3	15.5	unsupervised
K-medoids	33.4	28.8	unsupervised
Elhamifar et al. [47]	37.8	42.0	unsupervised
Song [153]	-	50.0	unsupervised
SUM-GAN [109]	39.1	51.7	unsupervised
DR-DSN [199]	41.4	57.6	unsupervised
Ours$_{pol}$	44.48	56.40	unsupervised
Ours$_{Q}$	44.56	56.44	unsupervised
Ours$_{AC}$	**46.40**	**58.30**	unsupervised

Table 4.6: Though not the focus of this paper, we evaluate our method on short video benchmarks as well for a thorough comparison. The table shows F-scores for various techniques on SumMe and TVSum datasets using basic RL rewards. Mentioned results are from respective original papers. We choose 5 fold validation (fixed five splits of both the dataset by the script provided by [199]) and reported an average F-score for all the proposed frameworks.

maximization. For a detailed comparison between all the frameworks and stability of the RL framework (training plots), we direct the reader to section A.3 and section A.4 respectively in the appendix.

Datasets	SumMe			TVSum		
Methods	DIST	IND	Both	DIST	IND	Both
Policy Gradient	44.5	44.74	44.76	56.10	56.3	56.40
Q Learning	45.1	45.2	45.62	55.72	55.72	56.44
Actor-Critic	46.36	46.48	46.40	55.77	56.66	58.30

Table 4.7: The table shows the F-scores measure of different techniques for various combinations of rewards for SumMe and TVSum datasets. DIST and IND represent the Distinctiveness and Indicativeness rewards, respectively. We choose 5 fold validation (fixed five splits of both the dataset by the script provided by [199]) and reported an average F-score for all the experiments.

Datasets	Disney			UTE		
Methods	DIST	IND	Both	DIST	IND	Both
Policy Gradient	26.77	27.23	28.54	42.87	43.4	45.15
Q learning	24.24	25.77	26.05	41.91	42.39	44.57
Actor-Critic	27.36	28.99	29.60	45.27	45.79	47.68

Table 4.8: The table shows the average RFS-50 (Relaxed F Score with temporal relaxation of 50) for three video sequences of Disney and UTE datasets for different rewards. DIST and IND represent the Distinctiveness and Indicativeness rewards, respectively. Note that the summary length reward is fixed to generate 5 minutes summary for all the experiments.

4.3.6 Ablation Study using various rewards

We have conducted extensive experiments to demonstrate the contribution of each reward in the final summary. We consider two basic rewards, namely distinctiveness, and indicativeness rewards, and did all the ablation for small and day long datasets in Table 4.7 and 4.8 respectively. The results show that both rewards individually cater complementary information, and when used together, we get performance improvement in all the experimental setups. For other plugins such as social interaction and interactive summarization, we did an extensive qualitative analysis. Furthermore, user feedback for interactive summarization is inherently subjective and dynamic, so we cannot demonstrate any quantitative analysis.

4.4 Conclusion

In this chapter, we have proposed a reinforcement learning based technique to generate personalized summaries of day long egocentric videos. Ours is the first technique with the capability to summarize such long sequences. We train our model end-to-end in a completely unsupervised manner and demonstrate the scalability of our technique on Disney, UTE, and HUJI datasets. To claim the superiority of our technique, we have performed extensive quantitative and qualitative evaluation, demonstrating significant improvement over `SOTA` results on long and short video sequences. Our framework allows the inclusion of various kinds of rewards in a plug-and-play manner, which can influence the selection of frames for the summary. We have shown the performance of our framework using visual diversity, representativeness, social saliency, faces, and summary length-based rewards. We also demonstrated how these rewards could be exploited to incorporate exemplar-based user preferences.

Chapter 5

Recovering Activity Patterns from Weeks Long Lifelogs

5.1 Introduction

Egocentric lifelogging applications typically require capturing and analysis of the huge volume of data. The data is often captured over weeks to months for a particular subject and contains long-term dependencies. For example, an activity may be performed only once daily but at a certain time of the day. For efficient indexing and browsing of lifelogging videos from an egocentric camera, we need automated tools for learning the activity patterns in an unsupervised setting. The focus of our work is to recover the activity patterns from photo-stream sequences recorded for multiple days (up to 20 days). The two critical challenges while solving the mentioned problem are: (a) extremely long sequences generated over multiple days, and (b) unavailability of annotated data due to enhanced privacy concerns in egocentric settings, and massive human effort required.

Recently self-attention based deep neural network models (referred to as `Transformer`) [166] have shown their superiority over convolutional architecture in a variety of tasks [49, 58, 59, 98]. Motivated by this, we explore the use of `Transformer` architecture for the task of activity clustering in extremely long egocentric videos for discovering activity patterns of the wearer. Multiple researchers have pointed out the inability of standard `Transformer` architecture to scale for extremely long sequences. This is primarily because the self-attention mechanism suffers quadratic compute and memory requirements with the sequence length. Further, `Transformer` models typically need large supervised data, and the unavailability of supervision in typical long sequence tasks makes it challenging for the application of `Transformers`.

A few works in the domain of natural language processing have proposed a sparse attention mechanism for `Transformers` to reduce the quadratic complexity to linear and handle long sequences [14, 183]. However, these works do not provide any theoretical guarantees and typically use fixed locations to compute global attention, affecting generalization capability. Choromanski et al. [25] have proposed a theoretically bounded linear-complexity attention mechanism (called `Performer`) that factorizes the regular

quadratic-complexity self-attention matrix proposed in [166]. This makes the `Performer` model most suitable for handling extremely long sequences. Broadly, `Performer` projects the query and key vectors into a fixed random subspace, and the projections conceptualize the factorization of a full-rank attention matrix. The main observation and a major contribution of this work is that the random subspace-based factorization is inadequate for attention modeling in extremely long video sequences.

We formulate the rank reduction of the attention matrix $\mathbf{A}$ as a non-negative matrix factorization (NMF). It has been shown that k-means clustering is a tractable approximation to the non-negative low-rank matrix factorization problem [38]. Motivated by this, instead of learning the low-rank factorization using random projections, we first find out the representative frames from all the input frames, and then use the features vectors from the representative frames, R, to learn low-rank matrices Q and K such that self-attention matrix $A = QR^\top RK^\top$. The use of representative frames allows us to integrate various semantic cues into the factorization process. In this work, we choose representative frames using a particular representative loss as described later. However, we note that other kinds of semantic loss functions could have been easily integrated into the proposed framework as well. We call the proposed architecture based on the proposed representative loss-based self-attention factorization as Semantic Attention Transformer or `SATFormer`.

We use the `SATFormer` for the self-supervised discovery of activity patterns using the following pipeline. First, we initialize c clusters from n frames using a process described later in the paper. Then in the first step, we use cluster membership to assign a pseudo-label to each frame. The pseudo labels are used to train the `SATFormer` with the representative frames used for factorizing the self-attention matrix instead of random vectors. This allows the `SATFormer` to learn a robust and meaningful frame representation. Then in the second step, we use the `SATFormer` representation to generate updated clusters. The two steps are iterated alternately until convergence. We use spectral clustering on the embeddings generate at the convergence to output the activity patterns.

Contributions: The key contributions of our work are:

1. We propose a novel `Transformer` architecture (`SATFormer`) based on the low-rank factorization of the self-attention matrix using proposed representative loss. The proposed architecture can exploit semantic cues to learn robust representation from extremely long video sequences.

2. We propose a self-supervised training scheme to discover activity patterns in extremely long egocentric lifelogs (recorded for up to 20 days). The approach does not rely on any priors, pre-trained networks to detect activities, objects, and/or places, and is specifically developed for unconstrained egocentric videos.

3. We demonstrate the performance of our contributions on the benchmark *Egoroutine* dataset and *Epic Kitchens* dataset. Compared to the current state-of-the-art, we report significant improvement in terms of (NMI, AMI, F-Score) of

(0.68,0.68,0.79) compared to (0.60,0.60,0.64) on the EgoRoutine photo-stream dataset, and (0.47,0.47,0.48) compared to (0.39,0.39,0.31) on Epic Kitchens video dataset.

4. We also contribute annotations for the daily routines of all 7 subjects in the dataset comprising 104 days of life-logging data.

5.2 Proposed Approach

We consider the photo-stream lifelogs recorded from wearable cameras of several subjects performing daily activities. The objective of this work is to recover activity patterns of one's lifelog recorded over multiple days. For the purpose of analysis, multiple sequences from a subject over multiple days are temporarily concatenated, resulting in a single colossal sequence per subject. We formulate the problem as a representation learning for a massively long temporal sequence in an unsupervised setting. The sequence representation learning formulation is motivated by the intuition that similar activity patterns should exhibit similar structures in latent space. The formulation demands explicit modeling of global dependencies as the activity patterns typically repeat only over a long interval (hours/days). The core technical contribution of this work is to learn an *embedding network* (f_θ^{emb}) for sequence representation learning that can handle extremely long sequences and model the global dependencies among similar activity patterns scattered across such sequences. Furthermore, we find that the clustering information plays a vital role in representation learning [10]. Therefore, we include it in the latent embedding using self-supervised learning.

5.2.1 Overview

Consider photo-stream lifelog of a subject recorder over D days. We concatenate these sequences in time, $\mathbf{X} = \{X_d\}_{d=1}^{D}$, to create a single sequence per subject spanning across days. The concatenation is required to discover and link the activities happening even only once a day. Let the number of frames in $\mathbf{X}$ be denoted by N. We use a BiLSTM model suggested in [57] to extract frame-wise features and use Principal Component Analysis (PCA) to reduce the feature dimension, and generate a 512 dimensional vector for each frame. The vector for the i^{th} frame in the sequence is denoted as $\mathbf{x}_i$. Our objective is to find c activity patterns/clusters from the week-long sequence of a subject. There is no assumption on order among a pair of activities, nor are all activities necessarily performed each day. Further, our technique does not impose any constraint that number of clusters should be same for every subject. Fig. 5.1 and Fig. 5.2 shows an overview of proposed `SATFormer`, and overall pipeline respectively.

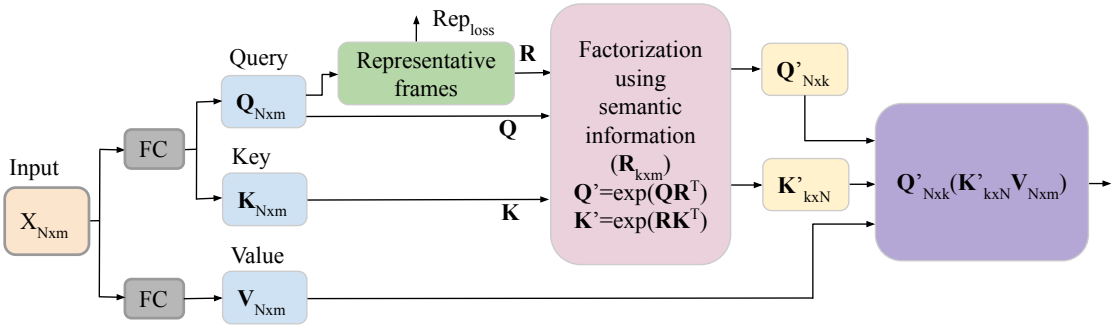

Figure 5.1: The figure depicts the proposed semantic attention that factorizes the self-attention matrix using semantically meaningful subspace by harnessing the latent characteristics of the data. It uses the representative frames sampled from the query $\mathbf{Q}$ instead of the fixed random vectors used in the `Performer`. The resulting projections $\mathbf{Q}'$ and $\mathbf{K}'$ are the *membership* matrices showing the distance of representative frames from the input sequence. Naturally, these two factorizations are the low-rank decomposition of the self-attention matrix using the saliency of the data. Intuitively, semantic attention generates a semantically meaningful subspace of k centroids learned by the inherent characteristics of the data. Our experiments reveal that these meaningful semantic centroids help disseminate better information compared to random frames used in the `Performer`. Furthermore, the representative frames are learned while training the network. We use the representative loss to ensure that the representative frames can reconstruct the query $\mathbf{Q}$.

5.2.2 `SATFormer`: Semantic Factorization of Self-attention Matrix

Self-attention in `Transformers`

To draw global dependencies between the input sequence, we take inspiration from the `Transformer` network [166] and borrow the self-attention mechanism in our *embedding network* (f_θ^{emb}) (see Fig. 5.2) which generates an embedding vector for each frame in the sequence. Once the input sequence $\mathbf{X}$ of length N is linearly projected as query $\mathbf{Q} = \{\mathbf{q}_i \mid \mathbf{q}_i \in \mathbb{R}^m, i \in [N]\}$, key $\mathbf{K} = \{\mathbf{k}_i \mid \mathbf{k}_i \in \mathbb{R}^m, i \in [N]\}$, and value $\mathbf{V} = \{\mathbf{v}_i \mid \mathbf{v}_i \in \mathbb{R}^m, i \in [N]\}$, where m is the query, key, and value dimensions, then the self-attention mechanism is given as follows:

$$\text{Att}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \mathbf{A}_{N\times N}\mathbf{V}_{N\times m}, \quad \mathbf{A}_{N\times N} = \text{softmax}\left(\frac{\mathbf{Q}_{N\times m}\mathbf{K}^T_{N\times m}}{\sqrt{m}}\right), \tag{5.1}$$

Here $\mathbf{A} \in \mathbb{R}^{N\times N}$ is the *attention matrix*. The vanilla self-attention used in `Transformers` leads to $\mathcal{O}(N^2)$ space and time complexity, and does not scale to long sequences.

Why Factorization of Self-attention Matrix?

The quadratic time complexity of the self-attention matrix should be addressed effectively to model the global dependencies in long sequence data. Our experiments also confirm that the self-attention mechanism [166] fails miserably for long sequences and gives memory error beyond a sequence length of $14k$. An active research area has emerged to gain compute and memory efficiency by approximating self-attention using various heuristics [14, 81, 132]. For example, Beltagy et al. [14] (`Longformer`) have proposed a sparse attention mechanism that uses two types of attention- the local attention for contextual representation and global attention for disseminating information across the full sequence. Kitaev et al. [81] (`Reformer`) have proposed a locality-sensitive hashing under the assumption that the nearby vectors assign the same hash value with high probability. In contrast, for lifelogs, our emphasis is on linking similar activity patterns scattered across the extremely long sequence. A fundamental approach to addressing this issue without relying on any heuristics and prior information is by factorizing the attention matrix into the low-rank query and key matrix pairs and changing the order of matrix matrix multiplication $\mathbf{Q}(\mathbf{K}^T\mathbf{V})$ for achieving linear space and time complexities [25, 142]. `Performer` [25] does the same by projecting the query-key pair onto a random subspace [25]. Our experiments reveal that a simple factorization shows moderate performance gain but is inadequate to harness the important visual information present in extremely long and repetitious video sequences. Hence, we propose a novel semantic factorization based on representative frames that harnesses the latent characteristics of the data for factorizing the attention matrix.

Semantic Factorization of Self-attention

To overcome the quadratic complexity of self-attention, we formulate the low-rank decomposition of attention matrix $\mathbf{A}$ as a non-negative matrix factorization (`NMF`) problem. We approximate the `NMF` using k-means, as it is a tractable approximation to the non-negative low-rank matrix factorization problem [38]. Precisely, we factorize a full rank attention matrix $\mathbf{A}$ to the low-rank matrices: *membership* matrix, $\mathbf{K}'$, and *reconstruction* matrix, $\mathbf{Q}'$, such that: $\mathbf{A} = \mathbf{Q}'\mathbf{K}'$. We first compute k representative frames from $\mathbf{Q}$, and then stack them to generate $\mathbf{R} \in \mathbb{R}^{k \times m}$. Then we learn a $k \times N$ matrix, $\mathbf{K}'$, such that $\exp(\mathbf{R}\mathbf{K}^\top)$ can be interpreted as the distance or membership coefficient of each sample from/of each of the k clusters (represented by the corresponding representative frame), where $\exp(.)$ is applied element-wise. Multiplication with $\mathbf{V}$, i.e. $\exp(\mathbf{R}\mathbf{K}^\top)\mathbf{V}$, can then be interpreted as finding k cluster centroids as the weighted sum of the samples according to their cluster membership. We interpret multiplication with $\mathbf{Q}'$, i.e., $\mathbf{Q}'\mathbf{R}\mathbf{K}^\top\mathbf{V}$, as reconstructing a sample as the weighted sum of cluster centroids. Since conceptually we expect the reconstruction weights to be the same as the cluster membership coefficients, $\exp(\mathbf{R}\mathbf{K}^\top)$, hence we enforce $\mathbf{Q}' = \mathbf{K}'$.

Mathematical Formulation of Semantic Factorization

It is instructive to note that while our proposed factorization provides rich conceptual motivation, we are basically suggesting to factorize $\mathbf{A} = \mathbf{Q'K'}$, such that $\mathbf{Q'} = \exp(\mathbf{QR}^\top)$, and $\mathbf{K'} = \exp(\mathbf{RK})$. Here $\mathbf{R}$ is a matrix comprising of a set of k vectors chosen in a particular way (using representative loss). Mathematically this is not much different from the `Performer`, in which the vectors are chosen as random vectors orthogonal to each other. Hence, mathematical justification for the `Performer` style factorization translates to ours as well. We give the detailed mathematical description for semantic factorization as follows.

The (i, j) element of *attention matrix* $\mathbf{A} \in \mathbb{R}^{N\times N}$ is the dot product of the row i of $\mathbf{Q}$ and row j of $\mathbf{K}$. We can equivalently denote it as $\mathbf{A}(i, j) = \kappa(\mathbf{q}_i, \mathbf{k}_j)$, where $\kappa : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^+$ is the kernel function. Kernel approximation is a powerful technique to approximate the quadratic complexity kernel by projecting the input features into a new space where dot products approximate the kernel well. Formally, given a kernel κ, kernel approximation methods seek to find a nonlinear transformation $\phi : \mathbb{R}^m \to \mathbb{R}^k$, for any $q_i, k_j \in \mathbb{R}^m$

$$\kappa(\mathbf{q}_i, \mathbf{k}_j) = \phi(\mathbf{q}_i^\top)\phi(\mathbf{k}_j). \tag{5.2}$$

Mathematical Results

We first prove the following two mathematical results before using them in our formulation.

Lemma 5.1. *For a random vector $w \in \mathbb{R}^m$ sampled from a Gaussian distribution with zero mean and identity covariance matrix (I_m), and vectors $x, y \in \mathbb{R}^m$, we have:*

$$exp\left(\frac{\|x+y\|^2}{2}\right) = \mathbb{E}_{w\sim\mathcal{N}(0,I_m)}\, exp(w^T(x+y)) \tag{5.3}$$

Proof.

$$\exp\left(\frac{\|x+y\|^2}{2}\right) = \exp\left(\frac{\|x+y\|^2}{2}\right)\cdot 1 \tag{5.4}$$

$$= \exp\left(\frac{\|x+y\|^2}{2}\right)\frac{1}{(2\pi)^{m/2}}\int \exp\left(\frac{-\|w-(x+y)\|^2}{2}\right)dw \tag{5.5}$$

Since w is a Gaussian distributed vector in $\mathbb{R}^m$ with zero mean and identity covariance matrix, the second term represents the total probability and hence should be 1.

$$\exp\left(\frac{\|x+y\|^2}{2}\right) = \exp\left(\frac{\|x+y\|^2}{2}\right)(2\pi)^{-m/2}\int \exp\left(\frac{-\|w-(x+y)\|^2}{2}\right)dw \tag{5.6}$$

$$= (2\pi)^{-m/2}\int \exp\left(\frac{\|x+y\|^2 - \|w-(x+y)\|^2}{2}\right)dw \tag{5.7}$$

$$= (2\pi)^{-m/2}\int \exp\left(\frac{\|x+y\|^2 - (w^Tw + \|x+y\|^2 - 2w^T(x+y))}{2}\right)dw \tag{5.8}$$

$$= (2\pi)^{-m/2}\int \exp\left(\frac{-(w^Tw - 2w^T(x+y))}{2}\right)dw \tag{5.9}$$

$$= (2\pi)^{-m/2}\int \exp\left(\frac{-\|w\|^2}{2}\right)\exp(w^T(x+y))dw \tag{5.10}$$

$$= \mathbb{E}_{w\sim\mathcal{N}(0,I_m)}\exp(w^T(x+y)) \tag{5.11}$$

Hence proved. □

Lemma 5.2. *For $x, y \in \mathbb{R}^m$, we have: $exp(x^Ty) = \kappa(x,y) = \phi(x)\phi(y)$, where:*

$$\phi(x) = \mathbb{E}_{w\sim\mathcal{N}(0,I_m)}\left[exp\left(-\frac{\|x\|^2}{2}\right)exp(w^Tx)\right], \tag{5.12}$$

$$\phi(y) = \mathbb{E}_{w\sim\mathcal{N}(0,I_m)}\left[exp\left(-\frac{\|y\|^2}{2}\right)exp(w^Ty)\right], \tag{5.13}$$

and w is a Gaussian distributed vector in $\mathbb{R}^m$ with zero mean and identity covariance matrix (I_m).

Proof.

$$\exp(x^Ty) = \exp\left(\frac{1}{2}(-x^Tx - y^Ty + x^Tx + y^Ty + x^Ty + y^Tx)\right) \tag{5.14}$$

$$= \exp\left(\frac{1}{2}(-\|x\|^2 - \|y\|^2 + (x+y)^T(x+y)\right) \tag{5.15}$$

$$= \exp\left(\frac{1}{2}(-\|x\|^2 - \|y\|^2 + \|x+y\|^2)\right) \tag{5.16}$$

$$= \exp\left(\frac{(-\|x\|^2 - \|y\|^2}{2}\right)\exp\left(\frac{\|x+y\|^2}{2}\right) \tag{5.17}$$

Using Theorem 5.1 to replace second term in the R.H.S.

$$= \exp\left(\frac{(-\|x\|^2 - \|y\|^2}{2}\right)\mathbb{E}_{w\sim\mathcal{N}(0,I_m)}\exp(w^T(x+y)) \tag{5.18}$$

$$= \mathbb{E}_{w\sim\mathcal{N}(0,I_m)}\left[\exp\left(w^T x - \frac{\|x\|}{2}^2\right)\exp\left(w^T y - \frac{\|y\|}{2}^2\right)\right] \tag{5.19}$$

$$= \phi(x)\phi(y) \tag{5.20}$$

where $\phi(x)$ and $\phi(y)$ are as given by Equations 5.12 and 5.13 respectively. Hence proved. □

Softmax Kernel Approximation using Semantic Kernel

We can use Theorem 5.2 to write the the attention matrix $\mathbf{A}$ as *softmax-kernel* as follows:

$$\mathbf{A}(x,y) = \exp(x^\top y) = \kappa(x,y) = \phi(x)\phi(y), \tag{5.21}$$

where we have ignored the scaling factor of softmax. We have also ignored $\sqrt{m}$-normalization, which can be equivalently done by normalizing query and key matrices accordingly.

Instead of fixed random Fourier feature transform using random vector w as proposed in [25, 26, 129] to approximate the *kernel* $\kappa(x,y)$, we use representative frames ($\mathbf{R}$). The proposed semantic kernel (ϕ_{sem}) defined as below projects the data into a semantically meaningful space:

$$\phi_{\text{sem}}(x) = \sum_{R_i\in\mathbf{R}} \exp\left(-\frac{\|x\|^2}{2}\right)\exp(R_i^T x), \tag{5.22}$$

where $\mathbf{Q} \overset{iid}{\sim} \mathcal{D}$ (a standard normalized input distribution) and $\mathbf{R} \in \mathbb{R}^{k\times m}$, $\mathbf{R} \subset \mathbf{Q}$. Here, we pretend that the feature vectors of representative frames are sampled from zero mean, unit covariance Gaussian. Intuitively, the semantic kernel reduces the rank of the attention matrix from N to k by projecting into the space of representative frames.

Now we compute $\mathbf{Q'} = \phi_{sem}(\mathbf{Q})$ and $\mathbf{K'} = \phi_{sem}(\mathbf{K})$, where $\mathbf{Q'}, \mathbf{K'}^\top \in \mathbb{R}^{N\times k}$ are the factorization of attention matrix $\mathbf{A}$ and *exp* is applied element-wise. With this kernel trick, we can change the order of multiplication of query $\mathbf{Q'}$, key $\mathbf{K'}$ and value vectors $\mathbf{V}$.

$$\widehat{\text{Att}}_{sem}(\mathbf{Q},\mathbf{K},\mathbf{V}) = \mathbf{Q'}(\mathbf{K'}.\mathbf{V}) \tag{5.23}$$

This multiplication is characterized by the time complexity of $\mathcal{O}(Nkm)$ and space complexity of $\mathcal{O}(Nk+Nm+km)$ compared to $\mathcal{O}(N^2+Nm)$ and $\mathcal{O}(N^2m)$ of the self-attention [166] and allows us to scale it to long egocentric sequences.

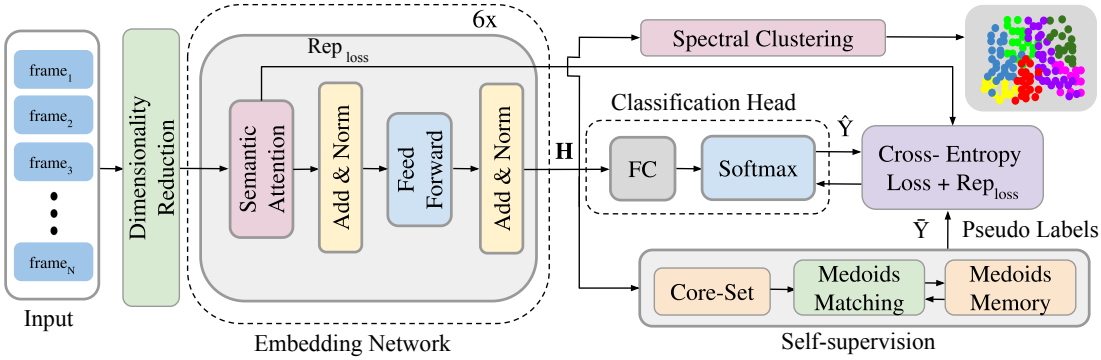

Figure 5.2: Illustration of flow chart of proposed `SATFormer`. Our technique consists of a neural network f_θ parameterized by θ that is further divided into two parts. The first part is an *embedding network*, $f_\theta^{\text{emb}} : \mathbb{R}^m \to \mathbb{R}^m$, that generates an embedding vector $\mathbf{H} \in \mathbb{R}^{N \times m}$. The second part is a *classification head*, $f_\theta^{\text{cls}} : \mathbb{R}^m \to \mathbb{R}^c$, consisting of a linear layer followed by the softmax operator, which generates the predicted labels $\hat{\mathbf{Y}} \in \mathbb{R}^{N \times c}$ corresponding to the input sequence of length N. We train the network using the pseudo labels $\tilde{\mathbf{Y}} \in \mathbb{R}^{N \times c}$ generated using the proposed self-supervised learning framework. Once the network is trained, we perform spectral clustering [115], with the number of clusters c, using the affinity matrix generated by the latent representation given by the *embedding network*.

Finding Set of Representative Frames

Whereas the `Performer` uses random projection vectors to learn $\mathbf{Q}'$, and $\mathbf{K}'$, we enforce that $\mathbf{Q}' = \exp(\mathbf{Q}\mathbf{R}^\top)$, and $\mathbf{K}' = \exp(\mathbf{R}\mathbf{K}^\top)$, where $\mathbf{R}$ is a matrix of features of representative frames. This ensures that the factorization proceeds by first projecting to meaningful cluster centers and then reconstructing based on these projections. Given the above motivation, any implementation to find good representative frames out of $\mathbf{Q}$ could have been applied. In our implementation, we use the following specific technique, where we update the representative frames (denoted by the vectors $\mathbf{q}_j$ in the equation below) in each epoch using the latest feature embedding learnt so far, and the ones which optimize the following loss function:

$$\mathcal{L}_{\text{Rep}} = \min_{\{\mathbf{q}_j\} \text{ s.t. } \|\{\mathbf{q}_j\}\|=k} \sum_{i \in N} \min_{\mathbf{b}^i} \left\| \mathbf{q}_i - \sum_{j=}^{k} b_j^i \mathbf{q}_j \right\|_{22}. \tag{5.24}$$

Here, $\mathbf{q}_i$ indicates the i^{th} feature embedding learned for sample i, i.e., the i^{th} row of $\mathbf{Q}$ (recall that in our implementation $\mathbf{q} = \mathbf{K}$). Further, b_j^i denotes the weight corresponding to the query vector $\mathbf{q}_j$ computed for best reconstructing query vector $\mathbf{q}_i$. The set of weights $\mathbf{b}^i = \{b_j^i\}$ are found as the ones which can best reconstruct a sample $\mathbf{q}_i$ using the selected representative vectors $\mathbf{q}_j$.

5.2.3 Activity Patterns Clustering using Self-supervised Learning

The *embedding network* (f_θ^{emb}) uses the proposed semantic attention based factorization in a `Transformer` architecture as suggested in [166]. In a supervised setting we could have trained using the labels $y_1, ...y_N \in \{1, ..., c\}$ for each frame, drawn from the space of c possible labels of a subject. We can compute predicted class probability vector, $\widehat{\mathbf{y}}_i$, for each sample $\mathbf{x}_i$ by passing the output of the network f_θ from the softmax layer:

$$\widehat{\mathbf{y}}_i = f_\theta(\mathbf{x}_i)$$

The model can be trained using the cross-entropy loss computed as:

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{c}\mathbf{y}_i[j]\,\log\widehat{\mathbf{y}}_i[j], \tag{5.25}$$

where $\mathbf{y}$ is the one-hot vector corresponding to label y_i. In our settings long sequences and privacy-sensitive nature of the egocentric data prohibits availability of the ground truth label. Hence, we adopt a self-supervised approach where we first cluster the samples into c cluster based on the learned embeddings and use the cluster membership to generate pseudo-labels $\widetilde{\mathbf{y}}_i$ for each sample. We then train the embedding network using cross-entropy loss with respect to the pseudo-labels:

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{c}\widetilde{\mathbf{y}}_i[j]\,\log\widehat{\mathbf{y}}_i[j], \tag{5.26}$$

For clustering, we use the core-set algorithm [137] to generate 'c' medoids indices using the latest embeddings generated. The core-set algorithm is an efficient approximation of the k-center problem [137]. These medoids are aligned/matched to the previously generated medoids, and medoids memory (comprises indices of medoids) is updated. Once the medoids memory is updated, the pseudo labels ($\widetilde{\mathbf{y}}$) are generated with the current embedding and the latest medoids. To initialize the medoids memory, we apply the core-set for input features. The proposed framework is trained similarly to Expectation-Maximization (EM). The two steps, namely representation learning, and self-labeling are, as follows: (1) Freeze the current label assignment matrix $\widetilde{\mathbf{y}}$, and update the model f_θ by minimizing the Equation 5.26. (2) Freeze the current embedding ($\mathbf{H}$) compute the 'c' medoids and update the medoids memory.

5.3 Experiments & Results

5.3.1 Experimental Setup

Dataset: We demonstrate the results on a publicly available *EgoRoutine* dataset [160], comprising life-logging of seven subjects, for a total of 104 days. The dataset is captured

by a wearable camera, fixed on the chest of a subject, capturing at 2 frames-per-minute (fpm), constituting 115,685 captured frames in total. Compared to conventional egocentric datasets, this dataset is recorded in a highly unconstrained environment that includes a variety of indoor and outdoor scene contexts. The activities are shopping, visiting restaurants/museums/concerts, traveling on flight/bus/cab/metro, working in a lab, attending conferences, cycling, sitting at the beach, etc. The dataset does not provide activity annotations. However, we have annotated all seven subjects for our experiments. We will release the annotations post-publication. Furthermore, we have tested the framework on the *Epic Kitchens* dataset to check the efficacy of the proposed approach. To demonstrate the proposed framework on the *Epic Kitchens* dataset [28], we synthesize a long video sequence (approx. 20k frames) using the *Epic Kitchens* dataset. This dataset is divided into high-level categories based on the occurrences of 'noun' classes. We pick equal video snippets of each category: 'appliances,' 'cleaning,' 'crockery,' 'drinks,' 'furniture,' 'meat,' and 'vegetables' across all subjects and concatenate them to form a long colossal sequence. The subset 'noun' classes selected for each category are listed in Table 5.1.

Category Id	Category Name	Nouns selected
1	Appliances	Washing Machine, Fridge
2	Cleaning	Cloth, Towel
3	Crockery	Plate, bowl
4	Drinks	Tea, Juice, Wine, Drink, Beer, Whisky
5	Furniture	Floor, Chair, Wall
6	Meat	Meat, Chicken, Sausage, Fish, Pork, Bacon, Beef
7	Vegetables	Onion, Potato, Carrot, Tomato, Mushroom, Cucumber, Vegetables

Table 5.1: Nouns selected corresponding to the categories for *Epic Kitchens* dataset.

Annotations: We have recruited three participants from different backgrounds (ECE undergraduate, CSE undergraduate, and CSE graduate) for annotation. We have generated codebooks of each subject of the *EgoRoutine* [160] dataset separately and shared it with participants to annotate videos on the same granularity. Each *label file* comprises the activity number and the corresponding activity name (refer to the Table 5.2 for the label file of subject-1 of *EgoRoutine* dataset). We share an annotation file with the participants, comprises two columns titled *start time* and the *activity number* for each day of the subject. The activities span for short to very long duration, so we just collect the activity's start time with its corresponding activity number (from the label file). Precisely, for a particular day of photostream sequence, the user needs to start from the first fame of the sequence and identify the activity performed from the activity codebook shared. The timestamp of the frame and the activity number is filled in the two columns discussed. The timestamp can be obtained from the frame name itself. Each frame is

named *xxxxxxxx_HHMMSS_xxx.jpg*, where $HHMMSS$ is the timestamp of the frame.

Activity Number	Activity Name
1	buying
2	having food in restaurant
3	having meeting and food at round table
4	working in lab
5	in metro
6	walking in lab and chitchatting
7	in gym
8	outdoor walking in day
9	outdoor walking in night
10	in lab kitchen
11	at metro station
12	walking in the building
13	in room
14	in cab
15	class room

Table 5.2: Activity labels for subject-1 of *EgoRoutine* dataset.

Table 5.3 demonstrates the details of the activity patterns used to annotate each subject. We can observe that the activity patterns are vast and similar to the real world. For each subject, the number and type of activity patterns differ significantly. The annotations also allow us to generate ground truth at multiple granularities as we annotated at high granularity. For example, we can always merge 'in metro', 'in cab' and 'in bus' activities to 'traveling' at low granularity.

Evaluation: For evaluation, we use the commonly used clustering evaluation metrics: Adjusted Mutual Information (AMI), Normalized Mutual Information (NMI), and F-score [148, 177]. These matrics range in $[0, 1]$, where larger values indicate better performance.
Adjusted Mutual Information (AMI): Suppose that the sequence of length N is partitioned in to predicted clusters $A = \{A_1, A_2, ...A_{K_p}\}$ and ground truth clusters $B = \{B_1, B_2, ..., B_{K_g}\}$, where, K_p an K_g are number of clusters in ground truth and predicted clusters. The clusters are pairwise disjoint i.e. $|\cup_{i=1}^{K_p} A_i| = |\cup_{i=1}^{K_g} B_i| = N$. Then the mutual information between two clusters can be defined as:

$$MI(A, B) = \sum_{i=1}^{K_p}\sum_{j=1}^{K_g} P_{AB}(i, j) log \frac{P_{AB}(i, j)}{P_A(i)P_B(j)} \tag{5.27}$$

where $P_{AB}(i, j) = \frac{|A_i \cap B_j|}{N}$ denotes the probability that frame belong to both the clusters

Id	Number of Activities	Name of Activity Patterns
S1	15	buying, having food in restaurant, having meeting and food at round table, working in lab, in metro, walking in lab and chitchatting, in gym, outdoor walking in day, outdoor walking in night, in lab kitchen, at metro-station, walking in the building, in room, in cab, in class room
S2	25	in home kitchen, in balcony (tea), working on laptop at home, having food, walking in building, walking in day (walking outdoor), cycling, operating vending machine/ATM, at metro station, in metro, walking in night, purchasing, using mobile/kindle (in room), washroom, sitting at beach, in mall /hotel having food, bus, room view, using laptop in lab, using mobile/kindle but not in room, having tea (in room), walking in lab and chitchatting, in library, working on laptop at library, in museum
S3	16	room view, in kitchen, having food (room/restaurant/cafe), at metro station, in metro, in washroom, outdoor walk in day, walking in building/ taking printout from printer, walking in lab and chitchatting, working on laptop (watching movie on laptop), in class room/ attending presentation, using mobile,purchasing (in mall/food/bakery/watch), outdoor walk in night, at airport, in advisor's room
S4	31	in room (walking, kitchen), walking outdoor (day), walking in building, working on laptop or desktop (in room or lab), riding bike, in hospital waiting room, with doctor, having food/in restaurant, walking in lab and chitchatting, using mobile (outdoor/restaurant/airport), in classroom, in washroom, watching TV and using mobile in room, purchasing toys, veggies, fruits, at airport, walking outdoor in night, at metro station, in metro, driving car, in swimming pool, Blur images, in school, in plane, attending a presentation, coffee/tea break at conference/at lounge, giving presentation, in cab, hosting a conference as receptionist, in open-bus/bus, at poster, at beach and mountains
S5	24	in room, outdoor walk in day, walking in building, working on laptop, driving car, in metro, at metro station, walking in lab and chitchatting, class room/attending presentation in audi/conference), having food at (home/restaurant/in conference), outdoor walk in night, in cab, at airport, in flight, purchasing (on stores at airport/local shops/mall/tickets at bus station), in hotel room/ hotel, at conference venue/ lounge/ reception, in bus, on beach, archaeological zone, at poster, monument visit, bus station, watching television
S6	19	at home, outdoor walk in day, walking in building, working on laptop, walking in lab and chitchatting, purchasing (food, shoes, toys, books), having food (in lab, restaurant), washroom, outdoor walk in night, in kitchen, in classroom, in Bus, in hotel room, museum, in car, visiting a old township and mountains, at metro station, in metro, at circus
S7	25	at home, outdoor walk in day, walking in building, working on laptop, walking in lab and chitchatting, purchasing (food, cloths, sweets,fruits, in supermarket), having food (in lab, restaurant), at metro station, in metro, at fair, washroom, outdoor walk in night, blank frame, in bus, in hospital/clinic/medical facility, in classroom, at concert, meeting with professor, sitting in park (picnic), at conference venue, at poster , attending presentation, in kitchen, in car, walking in hill area/trekking

Table 5.3: Table demonstrates the number of activities and the name of activity patterns used to annotated the life-logs of the subject.

$A_i \in A$ and $B_i \in V$, $P_A(i) = \frac{|A_i|}{N}$, and $P_B(j) = \frac{|B_j|}{N}$.

The expected mutual information can be defined as:

$$E\{MI(A,B)\} = \sum_{i=1}^{K_p}\sum_{j=1}^{K_g}\sum_{n_{ij}=(a+b-N)^+}^{\min(a_i,b_j)} \frac{n_{ij}}{N} log(\frac{N.n_{ij}}{a_i b_j}) \times \frac{a_i! b_j! (N-a_i)! (N-b_j)!}{N! n_{ij}! (a_i - n_{ij})! (b_j - n_{ij})! (N - a_i - b_j + n_{ij})!} \tag{5.28}$$

where $(a_i + b_j - N)^+ = max(1, a_i + b_j - N)$, n_{ij} denotes the number of common frames in clusters A_i and B_j, $a_i = \sum_{j=1}^{K_g} n_{ij}$, and $b_j = \sum_{i=1}^{K_p} n_{ij}$.
From Equation 5.27 and 5.28, the AMI is defined as [117]:

$$AMI(A,B) = \frac{MI(A,B) - E\{MI(A,B)\}}{max\{H(A), H(B)\} - E\{MI(A,B)\}} \tag{5.29}$$

Normalized Mutual Information (NMI): Similarly, from equation 5.27 and 5.28, the NMI is defined as [117]:

$$NMI(A,B) = \frac{MI(A,B)}{(H(A) + H(B))/2} \tag{5.30}$$

F-score: Similar to [15, 82], we use the greedy approach [86] for a one-to-one mapping between the predicted clusters and ground truth clusters. The cost of assigning cluster i to class label j is computed as the F1 score weighted by population for class j when i is assigned to j. The precision (P_r), recall (R_r) and F-score (F_r) are defined as:

$$P_r = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad \text{, and} \quad R_r = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$F_r = \frac{2 \times P_r \times R_r}{P_r + R_r} \times 100\% \tag{5.31}$$

where TP, FP, and FN represent the true positive, false positive, and false negative calculations.

Baselines: We compare with a `SOTA` egocentric work [37] to demonstrate the efficacy of `SATFormer`. Dimiccoli et al. [37] use a threshold to control the granularity of segmentation. We tweak the threshold to generate the appropriate clusters for each subject. Due to the scarcity of recent works for activity pattern recovery, we select five Vision/NLP works aligned to our problem [10, 12, 120, 166]. Part et al. [120] propose a novel convolutional graph autoencoder called GALA (Graph convolutional Autoencoder using LAplacian smoothing and sharpening) for representation learning, and have validated their technique using various backbone networks on different image datasets. The training of graph convolutional autoencoder required a sparse adjacency matrix computed apriori to embed the underlying structure of the data in node embeddings. This is not feasible for an unsupervised setting. For establishing the baseline, we generate a sparse adjacency matrix by considering τ closest frames for an input frame in Euclidean

space under the assumption that the events are of equal length. We choose $\tau = 30$ to demonstrate the results. Similarly, Bai et al. [12] propose Deep Autoencoding Predictive Components (DAPC) that mask the feature dimension and temporal dimension of the input sequence and reconstruct the masked component from the latent representations. The DAPC uses multiple configurations for the encoder, such as `linear`, `lstm`, `bgru`, `blstm`, and `Transformer`. The `Transformer` encoder shows memory error in our case due to long sequences. Hence, we show results on `bgru` configuration. Sarfraz et al. proposed a hierarchical clustering algorithm that groups semantically related frames of a video using a 1-nearest neighbor graph. The algorithm partitioned the data at multiple granularities. We picked the partition closest to ground-truth clusters for comparison. Chen et al. [24] proposed a contrastive action representation learning (CARL) framework that uses a novel sequence contrastive loss. We trained the architecture on our datasets and used spectral clustering on the frame-wise representations generated. Furthermore, to prove the efficacy of the proposed semantic attention, we replace it with three `SOTA` attention mechanisms, namely `Transformer`, `Longformer`, and `Performer` in the proposed architecture, and call them as `SATFormer-Trans`, `SATFormer-Long`, and `SATFormer-Perf`, respectively. We also compare `SATFormer` with `SOTA` self-supervised framework proposed by Asano et al. [10]. We replace the fully connected layer in [10] with the proposed representative frame-based attention transformer and have named it `SATFormer-SeLa`.

Implementation Details: The proposed `SATFormer` architecture uses six layers of transformer encoder, each of which uses one attention-head with the proposed semantic attention mechanism for the embedding network. We use Principal Component Analysis for dimensionality reduction for all the experiments, which resulted in a 512 dimensional feature vector. We utilize $m/2$ frames to compute the representative loss at each layer and backpropagate along with cross-entropy loss. For medoids matching, we use bipartite matching between the previously generated medoids (extract the current embedding corresponding to the previously generated indices stored in medoids memory) and current medoids in Euclidean space. We generate pseudo labels for every 50^{th} epoch. We set the learning rate as 0.01, the number of neurons at the feedforward network as 2048, and the adam optimizer with a 4000 epoch of warmup [166]. We use $f = \text{ReLU}$ for better generalization similar to `Performer`. We remove the positional encoding as the sequence of the events is stochastic for the problem at hand. For `Performer` attention, we use 8 parallel attention heads and the generalized ReLU kernel. For `Longformer` attention, we use 8 parallel attention heads, 500 uniformly distributed indexes for global attention, and a sliding window size of 60 for local attention.

5.3.2 Results and Discussion

Quantitative Comparison for Different Number of Clusters: Table 5.4 shows the quantitative evaluation based on AMI, NMI, and F-score for different granularities of clusters. We demonstrate that `SATFormer` outperforms all the `SOTA` frameworks

Methods	c = 12			c = 13			c=15		
	F1↑	AMI↑	NMI↑	F1↑	AMI↑	NMI↑	F1↑	AMI↑	NMI↑
SR-clustering [37]	0.3044	0.0913	0.0924	0.2697	0.1294	0.1312	0.2614	0.1537	0.1557
TWHC [134]	0.3132	0.1548	0.1603	0.3259	0.1649	0.1655	0.3072	0.1530	0.1545
SeLa [10]	0.6642	0.6291	0.6299	0.6662	0.6150	0.6158	0.5855	0.5954	0.5963
DAPC + GRU [12]	0.7135	0.6129	0.6135	0.6152	0.6040	0.6048	0.6343	0.6080	0.6089
GALA [120]	0.6357	0.6079	0.6085	0.6458	0.6084	0.6093	0.5381	0.5932	0.5941
SATFormer-Trans⋆ [166]	0.2262	0.1651	0.1674	0.2257	0.1749	0.1769	0.2292	0.1423	0.1451
SATFormer-Long [14]	0.5576	0.5989	0.5995	0.6212	0.6066	0.6073	0.6575	0.5982	0.5990
SATFormer-Perf [25]	0.6955	0.6219	0.6224	0.6001	0.5938	0.5944	0.6842	0.5996	0.6006
SATFormer-SeLa [10]	0.6478	0.5991	0.6025	0.6573.	0.6152	0.6160	0.7185	0.6276	0.6286
SATFormer	**0.7482**	**0.6510**	**0.6515**	**0.7976**	**0.6837**	**0.6842**	**0.7960**	**0.6806**	**0.6814**

Table 5.4: Comparison between various SOTA approaches for subject S1 in *EgoRoutine* dataset. For $K = 13$, we merge 'in cab' and 'in metro' to 'transportation' class and 'in lab kitchen' to 'walking in lab and chitchatting' class in the ground truth annotations. For $K = 12$, we further merge the 'food in lab' to 'at restaurant' class. $\star$ represents that Transformer gives memory error after 14000 sequence length, the results are evaluated for less than 14000 sequence length.

with a considerable margin for 14 days long sequence of subject S1. When we replace the proposed semantic attention with SOTA attention mechanisms such as Transformer, Longformer, and Performer, the performance drops considerably as the SOTA mechanism fails to harness the rich semantic information. Furthermore, in SATFormer-SeLa, we use the self-supervised learning framework proposed by [10] instead of our proposed self-supervised framework. They use the equipartition assumption for generating the pseudo labels. However, the equipartition assumption does not work, as the activity patterns in egocentric lifelogs are highly skewed. Due to poor pseudo labels, the SATformer-SeLa can not harness the clustering information and significantly underperform compared to SATFormer.

Qualitative Results: Fig. 5.3 demonstrates visualization of the results obtained for the sequence corresponding to subject S1 (all 14 days concatenated sequentially). The figure shows that SATFormer performs robustly for all activity patterns. We observed that the most repetitious activity pattern, 'working in lab' is handled and significantly recovered. Furthermore, the SATFormer is robust for minority classes as well and precisely recovers 'in cab' (appeared once on day 10, refer Fig. 5.3) and 'at metro station'. However, we observe misclassifications due to high overlap among the context and the objects involved in the activity patterns. For example, 'food in lab' is frequently misclassified as 'walking in lab and chitchatting' or 'kitchen' as the former shares the common context (the lab) and the latter shares common objects (the food). Furthermore, 'walking in lab and chitchatting' shows confusion with 'walking in building' and 'working in lab' at the boundaries due to the smooth transition between the activity patterns. The same can be validated by the confusion matrix in Fig. 5.4.

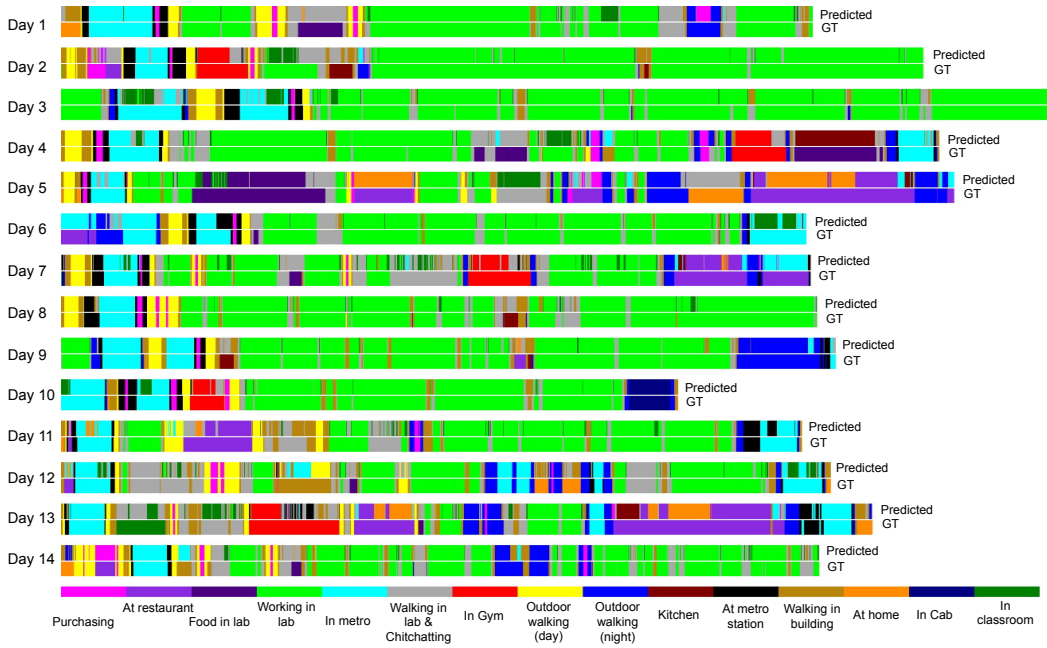

Figure 5.3: The figure demonstrates the visualization of a comparison between the predicted class and ground truth for different days (for better visualization, we have divided the concatenated sequence into multiple days). We use Hungarian matching for a one-to-one mapping between ground truth and predicted clusters. (Figure best visible in color.)

Quantitative Comparison for All Subjects: Table 5.5 demonstrates the quantitative comparison with the top-performing `SOTA` frameworks for all the seven subjects of the *EgoRoutine* dataset. We show significant performance improvement in terms of F1-score, AMI, and NMI for all the subjects. We observe that the GALA [197] performs comparably to the proposed framework for $S2$ as it uses a sparse adjacency matrix with τ closest frames, and the choice of τ seems best for this subject. Table 5.7 demonstrates significant performance gain compared to SOTA techniques for the *Epic Kitchens* datasets in terms of F1-score, AMI, and NMI.

Ablation Study: Table 5.6 shows an exhaustive ablation analysis demonstrating the contribution of various design choices in `SATFormer`. We first replace the novel factorized attention with `Performer` attention [25] and demonstrate the results generated by uni-head and multi-head attention. The results reveal that multi-head attention performs better than uni-head attention, as claimed in [25]. However, for the proposed semantic attention, uni-head attention performs significantly better than multi-head version. This is because in multi-head attention, when we split the feature vector along the feature dimension and select the most representative frames, then the global information of the image is compromised. Each head focuses on a small part of the feature embedding that lacks the global context of the activities essential for the problem at hand. Fur-

Id	Score	TW-FINCH (CVPR'21)	SeLa (ICLR'20)	DAPC (ICLR'21)	GALA (CVPR'19)	SATFormer -perf	SATFormer
S1	AMI	0.1530	0.5954	0.6080	0.5932	0.5939	**0.6806**
	NMI	0.1545	0.5963	0.6089	0.5941	0.5948	**0.6814**
	F1	0.3072	0.5855	0.6343	0.5381	0.6423	**0.7960**
S2	AMI	0.3489	0.4832	0.4794	**0.4901**	0.4765	**0.4901**
	NMI	0.3551	0.4889	0.4852	0.4932	0.4824	**0.4957**
	F1	0.2541	0.4497	0.4504	0.4901	0.4395	**0.4960**
S3	AMI	0.1038	0.4704	0.5083	0.5262	0.4891	**0.5756**
	NMI	0.1055	0.4717	0.5096	0.5275	0.4905	**0.5768**
	F1	0.2227	0.4885	0.5546	0.5965	0.5208	**0.7202**
S4	AMI	0.4640	0.5474	0.5518	0.5630	0.5663	**0.5750**
	NMI	0.4699	0.5513	0.5557	0.5668	0.5699	**0.5786**
	F1	0.2882	0.4200	0.4415	0.5117	0.4575	**0.5821**
S5	AMI	0.4722	0.5845	0.5868	0.5658	0.5787	**0.5913**
	NMI	0.4769	0.5870	0.5892	0.5685	0.5812	**0.5937**
	F1	0.3230	0.4808	0.4907	0.4707	0.4671	**0.6074**
S6	AMI	0.1801	0.5371	0.5078	0.5838	0.5277	**0.6252**
	NMI	0.1823	0.5392	0.5101	0.5857	0.5297	**0.6272**
	F1	0.2645	0.5453	0.4213	0.6720	0.4928	**0.6813**
S7	AMI	0.3057	0.5510	0.5625	0.5630	0.5569	**0.5833**
	NMI	0.3078	0.5553	0.5667	0.5675	0.5612	**0.5873**
	F1	0.3584	0.4764	0.4953	0.5093	0.5264	**0.5745**

Table 5.5: Performance comparison with the top performing SOTA in terms of F1 score, AMI, and NMI for all the subjects of the *EgoRoutine* dataset.

Model	Network Hyperparams			Performance		
	Sem Attn	SharedQK	Attn Heads	F1	AMI	NMI
SATFormer-Perf	NA	NA	8	0.6842	0.5996	0.6006
SATFormer-Perf	NA	NA	1	0.6317	0.5900	0.5909
SATFormer	✓	✗	8	0.7076	0.6079	0.6096
SATFormer	✓	✗	1	0.7235	0.6319	0.6328
SATFormer	✓	✓	8	0.7096	0.6231	0.6240
SATFormer	✓	✓	1	**0.7960**	**0.6806**	**0.6814**

Table 5.6: Performance comparison the proposed framework SATFormer with various desing choises for subject 'S1' for 'c' =15. SharedQK, Sem Attn, Attn Heads, and NA represent the linear layers of query and key is shared, the semantic attention, the number of attention heads, and not applicable.

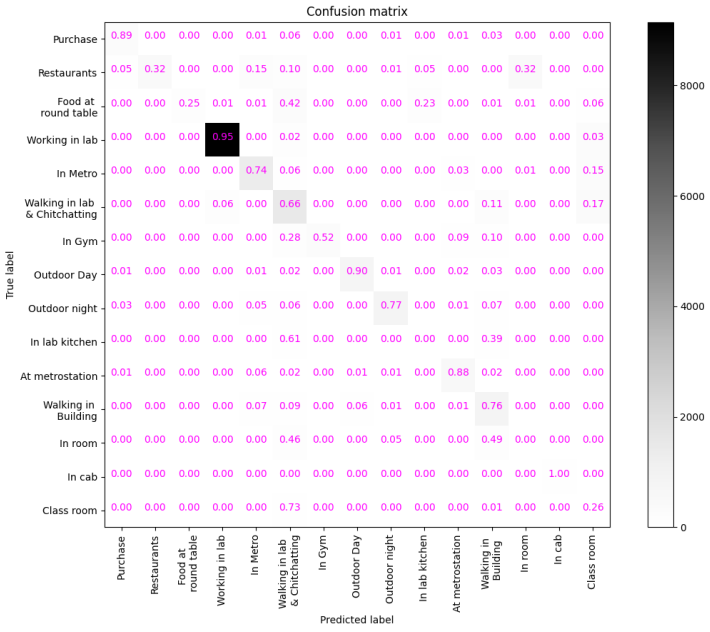

Figure 5.4: The confusion matrix demonstrates that inter-class confusion is marginal for most of the activity patterns.

thermore, we propose to enforce $\mathbf{Q}' = \mathbf{K}'$. Not only it makes conceptual sense, the results demonstrate substantial performance improvement using the proposal compared to when $\mathbf{Q}'$, and $\mathbf{K}'$ are allowed to be different. The bandwidth of the representative frames ($m/2$ used for all the subjects) is a latent characteristic of the data and depends upon the diversity of the lifelogs, so the performance may vary. With shared $\mathbf{Q}'$ and $\mathbf{K}'$, and uni-head attention, the proposed attention outperform `SOTA` frameworks with a considerable margin. We also visualize, and compare the attention map generated by `SATFormer`, and `Performer` in the supplementary material.

5.4 Conclusion

We focus on the problem of activity pattern clustering from the week-long recordings of a subject from an egocentric camera in a completely unsupervised setting. Our experiments with state-of-the-art revealed that current `Transformer` models can not handle such long sequences. Hence, we have introduced a novel semantic attention transformer that can exploit the redundancy present in the lifelogs for scaling to such long sequences. In the proposed `SATFormer`, we factorize the attention matrix into the low-rank query

Score	SeLa (ICLR'20)	DAPC (ICLR'21)	GALA (CVPR'19)	CARL (CVPR'22)	SATFormer -perf	SAT -Former
AMI	0.3229	0.0267	0.3900	0.3158	0.3884	**0.4710**
NMI	0.3234	0.0271	0.3904	0.3140	0.3887	**0.4713**
F1	0.3161	0.2051	0.3154	0.2992	0.4543	**0.4830**

Table 5.7: Performance comparison with SOTA in terms of F1 score, AMI, and NMI for the *Epic Kitchens* dataset.

and key matrices using a learnable and parameter-free semantic attention. Furthermore, we use a novel fast medoids-based self-supervised learning framework that incorporates clustering information into the generated representations. We provide detailed ablations for choosing uni-head attention and shared query-key projections for the proposed semantic attention. Our results on the *EgoRoutine* dataset recorded in a highly unconstrained setting demonstrate the efficacy of SATFormer. We also believe that our proposed semantic factorization of attention in Transformers will be useful for other computer vision tasks involving long sequential data as well.

Chapter 6

Conclusion and Future Research

Scalability and unlabeled data are two main challenges for analyzing egocentric videos in real-life environments. For efficient consumption, egocentric videos require robust video analysis techniques dealing with extremely long sequences in self-supervised/unsupervised settings. We demonstrate that SOTA sequential models viz Temporal Convolutional Networks (TCNs) [13], Recurrent Neural Networks (RNNs) [133], Long Short-Term Memory Networks (LSTMs) [69], Graph Convolutional Networks (GCNs) [45], and Transformers networks [166], fail to handle the massively long sequence.

This dissertation develops various video analysis techniques to analyze day to weeks long egocentric videos. Specifically, we address the three fundamental video analysis tasks: temporal segmentation, summarization, and recovering activity patterns, and establish the SOTA performance with a huge margin. The proposed frameworks use SOTA statistical and deep learning-based frameworks and demonstrate on real-life egocentric datasets. To the best of our knowledge, we are the first to work on the Disney (comprises 4 to 8 hrs long video samples) and UTE datasets (comprises 3 to 5 hrs long video samples) for temporal segmentation and summarization and EgoRoutine dataset (up to 20 days long photostream sequence) for activity patterns recovery.

6.1 Future Research

Despite the tremendous progress in supervised video analysis, the literature lacks robust works for long video analysis (first and third-person videos) in an unsupervised setting. Recent literature works use pre-trained CNN/LSTM networks for feature extraction that result in coarse-level information. They failed to model fine-level information, such as the evolution of the active objects, people, scenes, and their relationships. Due to the reason mentioned above, the community has not explored high-level video analysis tasks such as action/activity recovery at multiple granularities, visual question answering (VQA), and personalized summarization. Some of the future lines of research are as follows:

6.1.1 End-to-End Representation Learning for Long Videos

Describing a video by the sequence of events performed is crucial and serves as a general-purpose backbone for various video analysis tasks. E.g., we can extract recipes from multiple cooking videos or recover the routine of one's weeks-long lifelogging. This also helps identify any missing event, such as some ingredients or the order of the ingredient in the cooking video or the order of placement of components in assembling a machine or surgical video analysis. The task requires high-level semantic understanding from the videos in the wild. Most unsupervised video analysis frameworks are not end-to-end and extract spatial information from pre-trained CNN followed by sequential models (LSTMs, GCN, and Transformers) for temporal modeling. Few works use pre-trained active (manipulated objects) objects recognition models to harness fine-grained information. However, CNN's trained on the ImageNet dataset generate poor representations for cooking videos, and the error further escalates by sequential models used at the second level. Furthermore, using pre-trained object recognition networks raised the scalability issue. Motivated by the Vision Transformer (ViT) [44], in the future, we seek to expand the applicability of the Transformer for video analysis to model the fine-level information by exploiting the pixel-level information efficiently. We can train Transformer for videos by taking inspiration from NLP representation learning frameworks such as Masked Language Model (MLM) and Autoregressive (GPT). However, the striking difference between the videos and the text is that the atomic units (the objects) in the videos are not brittle compared to the text (the words). Each frame in the video contains multiple objects, and their relative positions lead to different meanings (leads to words in the text). This smooth continuum makes fine-level representation learning in videos makes very challenging. In MLM, we mask a random word in the sentence and try to reconstruct the masked word by harnessing the context. To mimic this concept in videos, we need to locate the objects present in the video in an unsupervised setting. We can use optical flow to locate the active object. Once we have the object location, we can mask it similarly to MLM. The possible future research direction of this work is to locate all the possible objects and use MLM to generate better embeddings. Once we have more instrumental and discriminative feature representation, we can more effectively approach tasks requiring higher-level semantic understanding.

6.1.2 Query-based Content Retrieval in Videos

Due to the popularity and affordability of video-capturing devices, the amount of video data created per day has increased tremendously in the last few years. Query-based content retrieval in videos currently relies on the video description (often manually generated) and meta-information available with the video. Query-based content retrieval from videos in the wild is still an ambitious problem and has enormous potential for handling such massive data. While working on personalized summarization, we observed that text-based video analysis increases user experience; however, from the algorithmic perspective, it is very challenging to bridge the gap between the two modalities. Similar

to VQA for images, we need a VQA for videos for efficient storage and consumption. In this problem, the user provides feedback in the form of text (query), and we need to retrieve the content in the videos aligned to the text feedback. This problem is closely related to personalized video summarization discussed in the dissertation; however, in this problem, the feedback is given in the form of text instead of video exemplars.

Many works demonstrate query-based content retrieval on very small-length videos [8, 55]. As the summary includes most of the significant events present in the day-long boring lifelogs, we can use these works on the generated summaries to locate the video segment that best matches the language query. However, the summary might miss the query events; in that case, we need efficient query-based content retrieval works to process the entire video.

The problem requires precise modeling of the scene, objects, and their relationship (structure of the video) in the video and reason these structures using text queries. We can harness GCNs to harness fine-level details of the objects and their manipulation at the frame level. Once we have frame-level structures, we can adapt the self-attention mechanism of the Transformer to exploit these frame-level structures (in the form of graphs) and disseminate information of similar frames/events across the long video to generate information embeddings. The core contribution is to redefine the self-attention mechanism that will work on GCN embeddings instead of a compact frame-level representation generated by pre-trained networks. The attention should handle the temporal evolution/manipulation of various objects and background necessary for VQA.

Appendices

Appendix A

Summarization and Personalized Summarization

In this appendix, we provide the following details omitted in the chapter 4:

- Section A.1: More qualitative analysis for predicted summaries
- Section A.2: Information Sheet
- Section A.3: Comparison between All the Frameworks
- Section A.4: Stability of RL Frameworks
- Section A.5 Detailed results on Personalized Summarization
- Section A.6: Demographic Information
- Section A.7: Algorithms
- Section A.8: Video Demonstration

A.1 More qualitative analysis for predicted summaries:

We further add more visualization to deeply inspect the proposed frameworks with all the ground truth summaries when using basic rewards. We choose 5 minutes summaries predicted by the proposed frameworks to compare against the three ground truth summaries ranging from 3 to 6 minutes. In Fig. A.1 and A.2, we demonstrate the visualization for the 'Alin Day 1' video sequence of the Dinsey dataset and the 'P01' video sequence of the UTE dataset, respectively, with all the three ground truth summaries. We operate on 1fps (a C3D feature is extracted per second) and get a binary mask as an output indicating the selected shots (of one second). In contrast, the ground truth summaries comprise a set of sentences, each corresponding to a 5 seconds clip. We map the clips to the original video sequence and generate the binary mast at one fps, similar to our predicted binary mask.

Fig. A.3 shows the comparison of 1 minute, 3 minutes, and 5 minutes summary generated by AC framework using the distinctiveness-indicativeness reward of 'HUJI Ariel 1' video.

We have also prepared the GUI of the proposed work to conduct a user study for personalized summarization. The GUI is shown in Fig. A.7. As discussed in section 4.3, the detail table with user comments on the personalized summary is shown in Table A.2

Table A.1 shows the summary length and sliding window size for two long sequence datasets, namely Disney and HUJI. As mentioned in section 4.3, we take sliding window size 25% of the desired summary length. To generate one-minute summaries, our summary length and sliding window size are 120 sub-shots (i.e. 2 sub-shots/second) and 30 sub-shots respectively. Similarly, for 10 minutes summaries, summary length and sliding window size are 120 and 30 respectively and so on for 3, 5, and 15 minutes summaries. For the Disney dataset, we train the network for 1, 5, and 15 minutes summaries, whereas for the HUJI dataset, we train the network for 1, 3, and 5 minutes summaries.

Summary length	Sliding window size
120 (1 min)	30
180 (3 mins)	45
600 (5 mins)	150
1200 (10 mins)	300
1800 (15 mins)	450

Table A.1: Summary length and sliding window size for summaries of various time durations.

A.2 Information Sheet

Below we give the verbatim text transferred to the subjects for the user study.

A.2.1 Information Sheet

You are being invited to take part in this study. Before you make a decision, it is important for you to understand why this study is being done and what it will involve. Please take time to understand the following information carefully. Please do not hesitate to ask us if there is anything that is not clear or if you would like more information. If you do take part, you will be asked to sign a consent form.

Figure A.1: The figure demonstrates the comparison between ground truth summaries and the summaries generated by the different frameworks for the 'Alin Day 1' video sequence of the Disney dataset. In each row, the black vertical bars indicate a frame was picked from a corresponding temporal window of 70 frames as it is not possible to visualize the video sequences at 1fps. As the annotations are done at 1/5 fps, pooling over a window of length 70 makes the ground truth summaries sparse. We can observe that in the first half and middle of the video, all three ground truth summary frames are uniformly distributed, whereas the selection is significantly less toward the end. The Actor-Critic framework also exhibits the same behavior, whereas the policy gradient and Q-learning perform slightly poorly compared to the Actor-Critic.

Objective:

We are conducting a study to understand how the system-generated summary of a day long egocentric video satisfies a user. We further extend our work to personalize the summary by taking user feedback and then ask the user to evaluate the personalized summary.

Risk:

The study is time-consuming. You may feel exhausted while participating in the study.

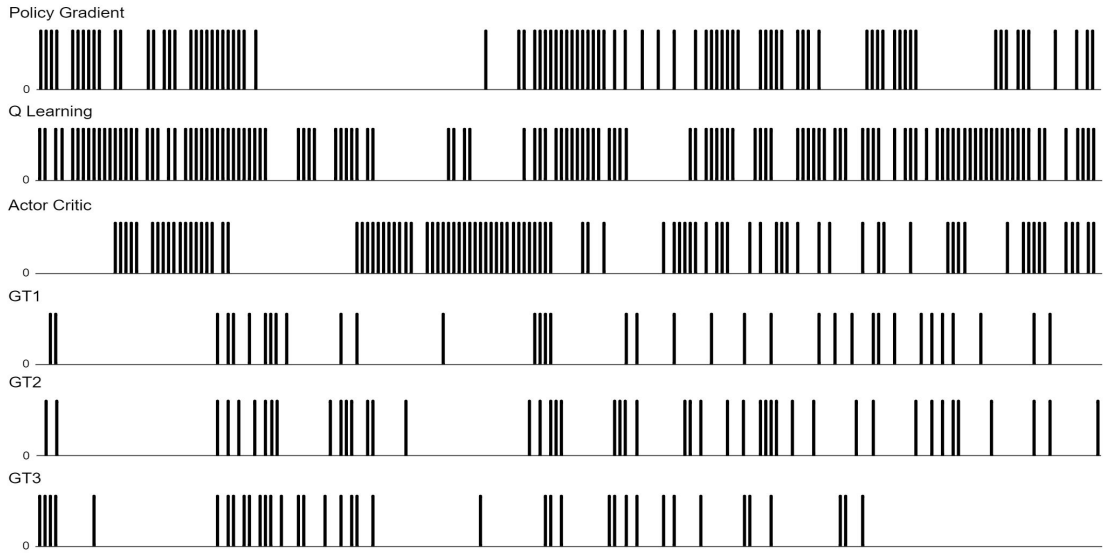

Figure A.2: The figure demonstrates the comparison between ground truth summaries and the summaries generated by the different frameworks for the 'P01' video sequence of the UTE dataset. In each row, the black vertical bars indicate a frame was picked from a corresponding temporal window of 70 frames as it is not possible to visualize the video sequences at 1fps. We can observe that the ground truth summary frames are approximately uniformly distributed in the second half of the video. The same distribution is observed for the predicted summaries from all the frameworks.

Benefits of study

You will not directly benefit from taking part in this study however as the summaries are inherently subjective so helping us out in the evaluation will open a new area of research. Additionally, you will receive incentive of INR 500 for your valuable time.

Confidentiality of research information

Taking part in this study is voluntary and you can stop at any time. We will be collecting demographic details of our participants. However no identifying information will be included in any publication or presentation, and your responses remain confidential.

Meaning of Terms

- **Informativeness** Informativeness score evaluates how many objects/events of the original video are included in the summarized video.

- **Enjoyability** The enjoyability assesses only the smoothness(jerk) of a video sequence.
- **Informativeness and Enjoyability** Rate the Informativeness and Enjoyability of the summary on the following scale.
  extremely dissatisfied = 1
  dissatisfied = 2
  neutral = 3
  satisfied = 4
  extremely satisfied = 5
- **Confidence score** This shows the confidence of the subject by which he/she provides the informativeness and enjoyability. The likert scale for the confidence is
  Not confident at all = 1
  Slightly confident = 2
  Somewhat confident = 3
  Fairly confident = 4
  Completely confident = 5

A.2.2 Evaluation Procedure

You would be evaluating summaries of three videos namely Alin Day 1, Alireza Day 1 and Michael Day 2. We have two step evaluation procedure, You are supposed to fill everything in the google form:

1. In the first step you will be asked to evaluate the generated summary. Once you finish viewing the summary then you will be asked to score the same for informativeness and enjoyability using the likert scale mentioned above (in the Google form). You will also be asked for a confidence score for informativeness and enjoyability together.
2. We will show you the GT text summaries (by three users). Once you read the GT text summaries, you will be asked to revisit the generated summary and modify your informativeness and enjoyability scores along with the confidence (if required). Kindly briefly justify your modification.

A.2.3 Generating personalized summary:

You are supposed to personalize and evaluate two videos. There are two scenarios for the personalization of the summary for each video.

1. In the first scenario, you are asked to choose the events from the system-generated summary (while being unaware of the video content). The detailed personalization procedure is as follows:

(a) You will select a video sequence and click the button "Generate Summary without Feedback". Once a default summary is generated you would be picking the interesting events which you want to include/exclude in the summary. You have to specify the time stamp as a feedback for positive as well as negative feedback. Kindly refer Figure A.7.

(b) When you click on the 'Generate Summary with Feedback' the personalized summary incorporating the suggested feedback is generated.

2. In the second scenario, we believe that you are aware of the video content.

(a) We ask you to see the original video and choose the events you want to include/exclude in the summary. You have to specify the time stamp as a feedback for positive as well as negative feedback. Kindly refer Figure A.9.

(b) When you click on the 'Generate Summary with Feedback' the personalized summary incorporating the suggested feedback is generated.

A.2.4 Evaluation procedure for personalized summary

1. Once the personalised summary is generated then you will rate the summary by the quality of personalization compared to default summary on the likert scale (1: very poor, 2: poor, 3: ok, 4: good, 5: excellent) with confidence (1: Not confident to 5: Completely confident).

2. To gauge your experience kindly answer the following question.

(a) "which events you wanted to include/exclude in the summary?"

(b) "why are you satisfied/not satisfied with the generated summary?"

A.3 Comparison between all the frameworks:

Fig. A.4 shows the training plot of policy gradient, Q learning, and Actor-Critic framework. As discussed in section 4.3, the episodic reward plot for the policy gradient shows high variance across video samples due to baseline function. Whereas Q learning and AC framework use Q value network leads to stable gradient across video samples. On the other end, the extra parameters required more training samples. If we have less training data, then the policy gradient is easy to train. For the proposed framework, each position of the sliding window constitutes one training sample, so we generate sufficient training samples (especially for day long videos) to train the Q learning and AC frameworks. The same is validated by Fig. A.4 shows the training plot of policy gradient, Q learning, and Actor-Critic framework.

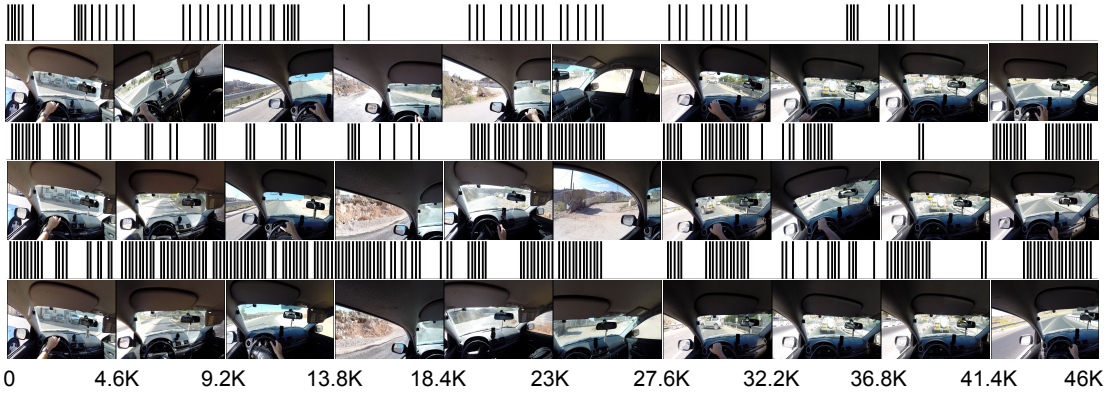

Figure A.3: Comparing 1, 3 and 5 minutes summaries (row 1-3) based on distinctiveness-indicativeness reward of 'HUJI Ariel 1' video.

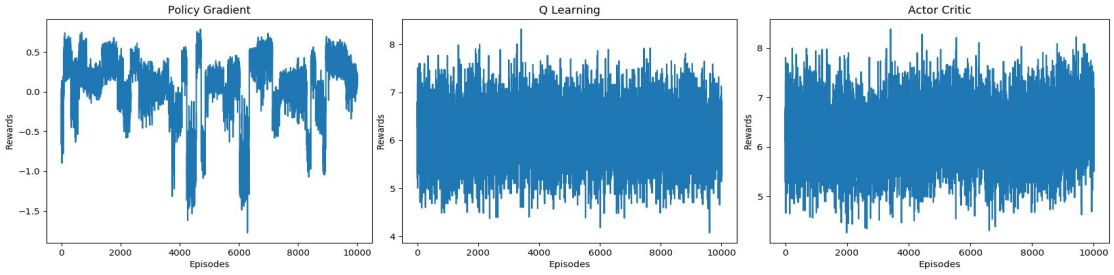

Figure A.4: The episodic reward plot of the policy gradient shows that we get clusters corresponding to each video sample as the baseline is not parameterized.

A.4 Stability of RL frameworks:

As we move the sliding window over the input video sample, it generates enough training samples to train any RL framework. We are successfully able to train policy gradient and Q learning. We also used experience replay for efficient convergence. Ideally, for the Actor-Critic framework, we have separate networks for actor and critic, but due to the diverse nature of each video sample, we are not able to train the AC framework. To get around the problem, we have used a common backbone LSTM network for actor and critic network followed by two fully connected heads for actor and critic, respectively.

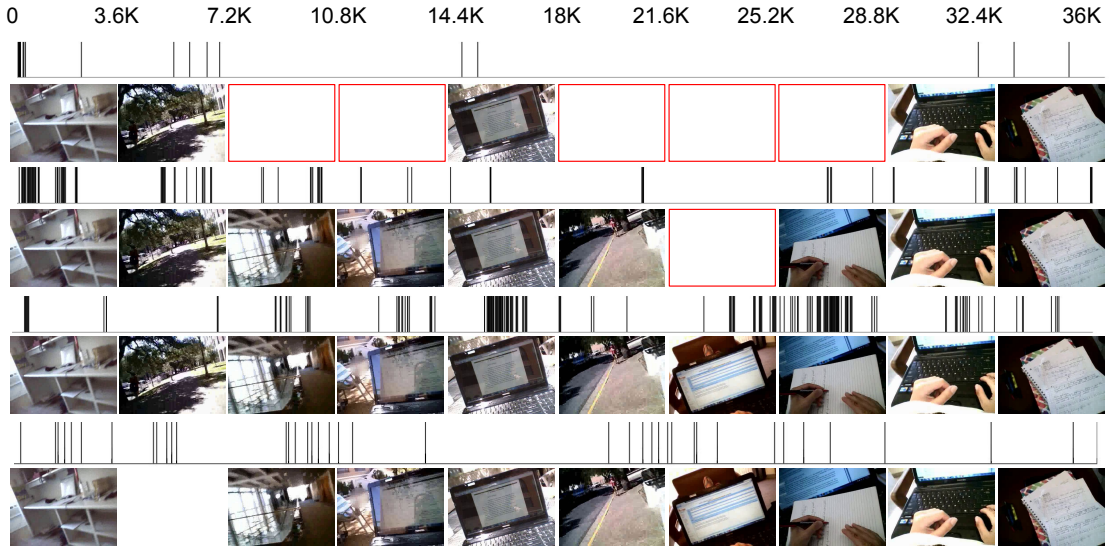

Figure A.5: Similar to Fig. 4.8, we compare 1, 5, 10 minutes summaries with the ground truth summary in rows 1 to 4, respectively. The summaries are generated using the basic reward using the Actor-Critic framework on the 'P04' sequence of the UTE dataset. We observe that the 1-minute summary does not capture the redundant part in which the subject is 'working on a laptop' (from 18K to 28.8K), whereas the redundant frames increase as the length of the summary increases.

A.5 Detailed Results for Personalized Summarization

The detailed results for all 10 participants in two different scenarios with participant's feedback are shown in Table A.2. The Likert score, along with confidence and participant's comments, shows that the participants are satisfied with the personalization to a large extent. We get 2.88 average (normalized by confidence) Likert score over 20 participants. Furthermore, it's clear from the participant's feedback that the frameworks struggle to completely eliminate the dark scenes when the participants want to exclude them from the summary. This happens because there are many dark scenes scattered throughout the video sequence.

A.6 Demographic Information

As discussed in section 4.3, the demographic details are shown in Table A.3.

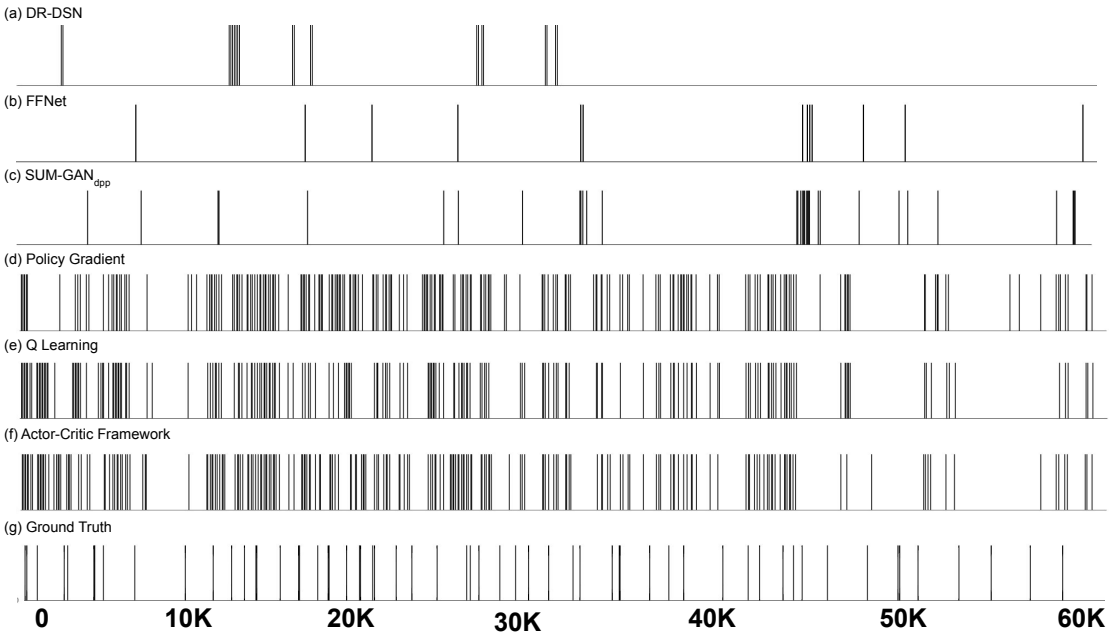

Figure A.6: We observed that DR-DSN [199] picks a cluster of frames from a particular location in summary, whereas the proposed frameworks effectively distribute the summary frame from all over the video. This figure gives a better visualization by showing the distribution of the summary frames for the full video. The bar chart from top to bottom represents the summary generated by DR-DSN [199], FFNet [87], SUM-GAN$_{dpp}$ [109], and our technique with Policy Gradient, Q Learning, and Actor-Critic framework respectively. The figure also indicates that despite using different RL frameworks, most of the selected summary frames are common as the reward is the same for all the frameworks.

A.7 Algorithms

We have discussed the proposed approaches in section 4.2. We give the exact algorithm steps here. Algorithm 2 elaborate the sliding window framework and Algorithm 3, Algorithm 4, and Algorithm 5 describes the training process of Policy Gradient, Q Learning, and AC framework respectively.

A.8 Video Demonstration

Please find the video demonstration of the interactive summarization module on this link. We have created GUI for this module. The video demonstrates how we can provide positive feedback (events you want to include) and/or negative feedback (events

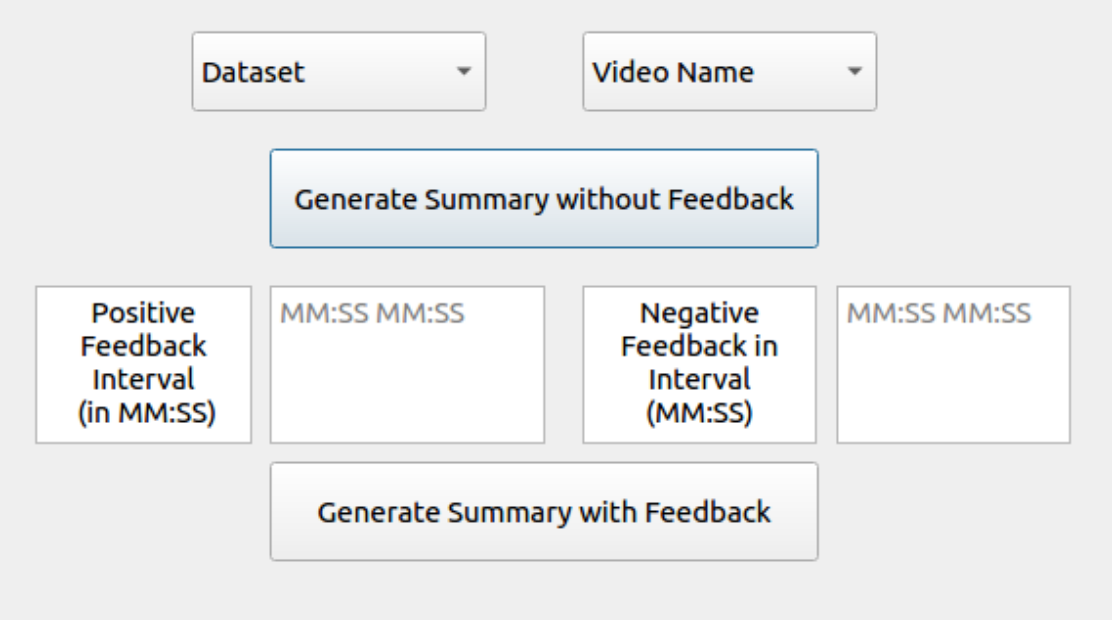

Figure A.7: Figure shows the GUI of the proposed work.

you want to exclude) to customize the generated summaries.

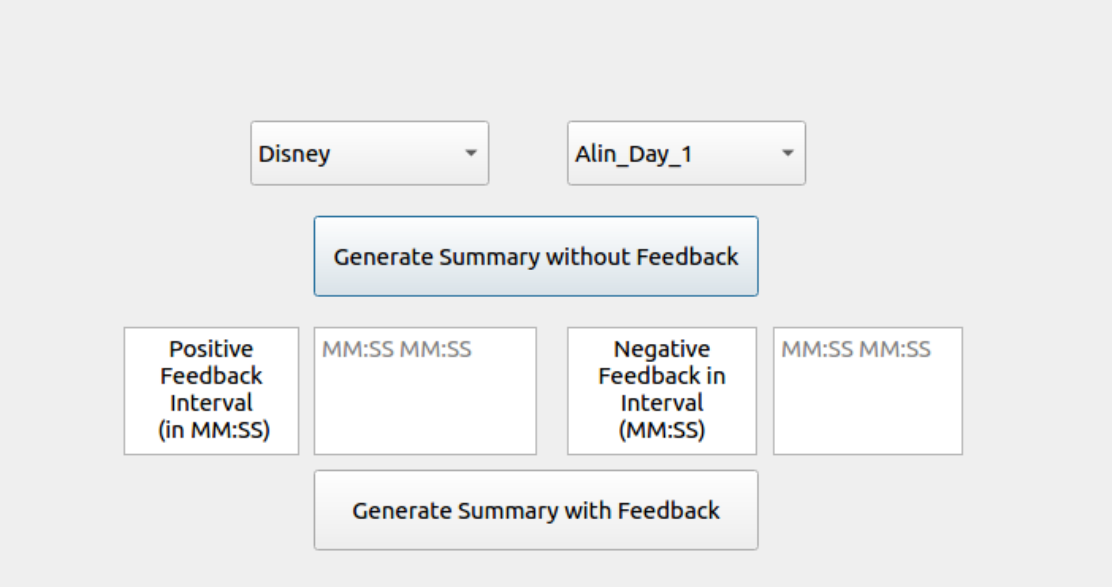

Figure A.8: GUI of the first scenario for personalization of summary.

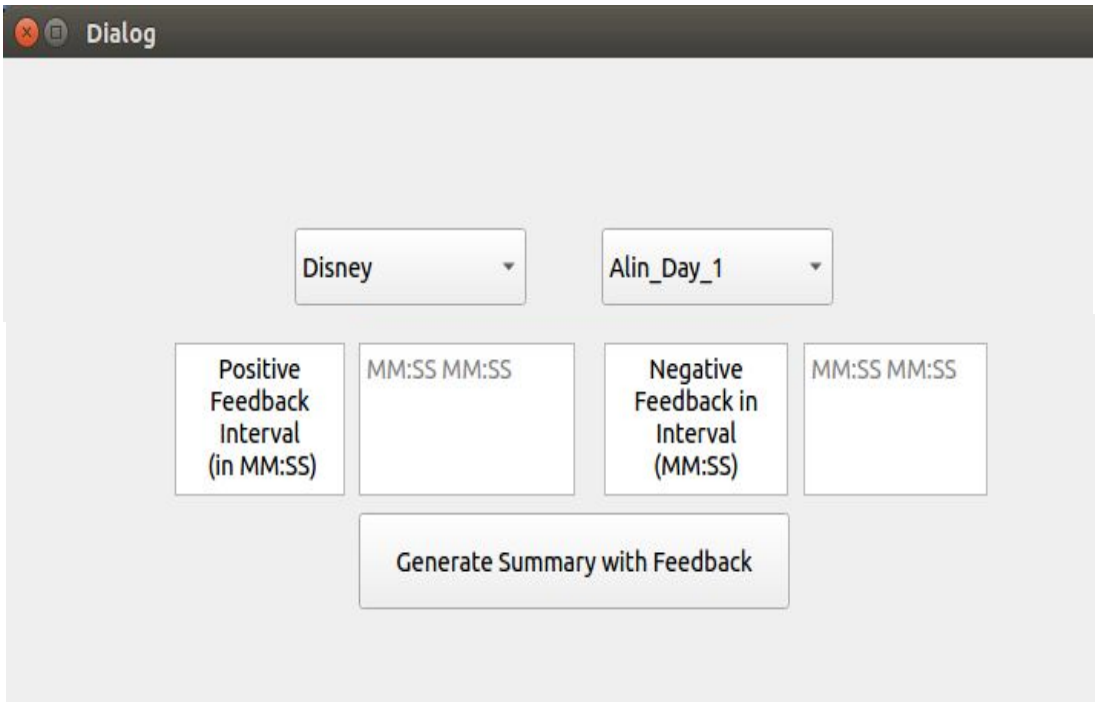

Figure A.9: GUI of the second scenario for personalization of summary.

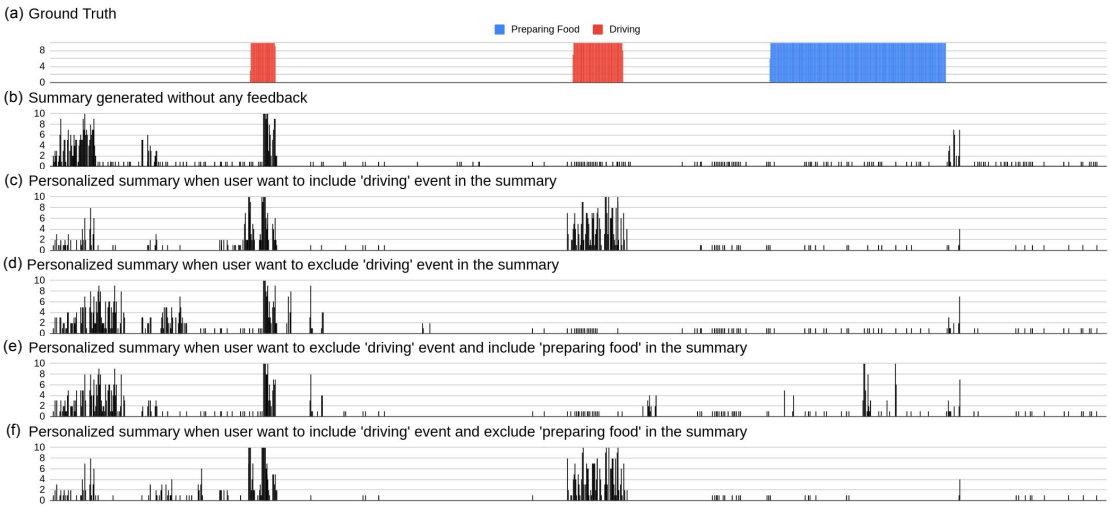

Figure A.10: The figure demonstrates the visualization of the interactive summarization of the 'P01' video sequence of the UTE dataset. Each bar represents 10 seconds of the time interval. (a)-(e) shows different summaries when two events, namely 'preparing food' and 'driving' are included/excluded in summary. We can observe that (c) has more driving sub-shots compared to (b), whereas in (d) the bars in the driving sub-shots are reduced considerably. Similarly, for (e) we get peaks in the 'preparing food' area, whereas the bars in the driving area are reduced. The opposite is seen in (d).

Algorithm 2 Proposed Framework

Input $F_{i=1}^{T}$**:** Video subshots
Output $P_{i=1}^{N}$**:** Probability scores

```
Freeze the C3D weights and randomly initialize weights of BiLSTM
for each epoch do
    for each video do
        for each pass do
            for each sliding window do
                Policy Gradient/Q Learning/ Actor-Critic
            end for
        end for
        if Policy Gradient then
            Update baseline B
        end if
    end for
end for
```

Subjects	Video -Dataset	Events Included	Excluded	LikertScore (1 to 5)	Conf.	Participant Feedback
S01-S1	Alin-Disney	Dinner	Dark scenes	3	4	*'Black part is not completely removed'*
S01-S1	P01-UTE	Driving	Social Int.	4.5	3	*'It accurately highligted part I liked and don't liked.'*
S02-S1	Alin-Disney	Dinner	Dark scenes	3	4	*'So many dark scenes'*
S02-S1	P01-UTE	lunch	Purchasing	3	4	*'Purchasing in store not removed completely'*
S03-S1	Alin-Disney	Dinner	Tram ride	5	4	*'Included really long dinner, Tram ride is mostly removed'*
S03-S1	P01-UTE	Social Int.	Driving	4	3	*'Detailed conversation, could exclude some more driving shots'*
S04-S1	Alin-Disney	Shopping	Escalator	4.5	4	*'Shopping is taken for little long, escalator is removed'*
S04-S1	P01-UTE	Driving	Writing	5	5	*'Majority of summary was driving, no writting event'*
S05-S2	Alin-Disney	Tram ride	Dinner	4	4	*'Dinner is almost removed'*
S05-S2	P02-UTE	Playing Lego	Eating Pizza	4	4	*'Eating is removed entirely and lego is included for more time'*
S06-S1	Alin-Disney	Dark room	Travel	4	4	*'Accurately included the suggested feedback'*
S06-S1	P02-UTE	Having pizza	Driving	2	4	*'Driving is not removed'*
S07-S1	Alin-Disney	Castle	Travel in bus	3.5	4	*'Overall its good, still there were some bus travel events'*
S07-S1	P01-UTE	Marketing	Driving	2.5	5	*'Lots of instances of driving which could have been reduced'*
S08-S1	Alin-Disney	Indoor	Outdoor	4	4	*'Most of video is outdoor based'*
S08-S1	P02-UTE	Ice Cream	Walking	3.5	4	*'Excluding is correct, inclusion is not very good'*
S09-S2	Alin-Disney	Tram ride	In bus, Dark	2	5	*'Tram ride is missing, rest is fine'*
S09-S2	P03-UTE	lunch, Payment	Purchasing	4.5	3	*'Summary is very nice'*
S10-S1	Alin-Disney	carousel	Dark scenes	2	4	*'Many dark scenes, poor summary'*
S10-S1	P03-UTE	Cooking	Drive, Wash	4	5	*'washing is removed, driving is not'*

Table A.2: The table shows the Likert score of 1 (Extremely dissatisfied) to 5 (Extremely satisfied) given by the participants when specific events are included or excluded in the summary with user comments on the personalized summary. S0X-SY represents subject 'X' in scenario 'Y'. It is observed that sometimes the user sees the excluded part in the personalized summary. This is because the interactive reward personalized the summary but at the same time distinctiveness-indicative reward that tries to maintain the global context. This can be handled by fine-tuning the weights of A and B discussed in interactive reward.

Algorithm 3 Policy Gradient Framework

Initialize θ and learning rate α.
for For each sliding window **do**
  Calculate S_{p} and S_{f} according to the position of W_{s}
  Get M probability scores from the neural network
  for For each episode **do**
    Sample M actions from probability scores
    Compute cost and reward

$$cost+ = \sum_{m=1}^{M} R(S)\nabla_{\theta} log \pi_{\theta}(a_m|h_m)]$$

  end for
  Compute episodic cost and episodic reward
  if episodic cost improves **then**
    update summary by picking top $|S|$ sub-shots
  end if
  if For each mini batch **then**
    Back-propagate pseudo batch cost
  end if
end for

Participant	Stream	Qualification	Gender	Professional Recording
S1	CSE	Ph.D.	Female	No
S2	CSE	Ph.D.	Female	No
S3	IT	Ph.D.	Male	Yes
S4	IT	Ph.D.	Female	No
S5	ECE	Undergrad	Male	No
S6	ECE	Undergrad	Male	No
S7	ECE	Undergrad	Male	No
S8	IT	Undergrad	Male	No
S9	IT	Undergrad	Male	Yes
S10	CSE	Undergrad	Male	Yes

Table A.3: Demographic Information of subjects for AHR. Three out of ten participants have professional video recording experience.

Algorithm 4 Q Learning Framework

Initialize θ, γ and learning rate α.
for For each sliding window **do**
  Calculate S_{p} and S_{f} according to the position of W_{s}
  Get M Q values from the Q value network
  Get M Q values from the target Q value network
  for For each episode **do**
    Sample M actions from probability scores
    Compute correction (TD error) for actions

$$\delta_m = R(S) + \gamma \sum_{m=1}^{M-1} Q_{\theta^-}(s_{m+1}, a_{m+1}) - \sum_{m=1}^{M-1} Q_\theta(s_m, a_m)$$

    Compute cost and reward R(S)

$$cost+ = \delta_m \sum_{m=1, a \in A}^{M} \nabla_\theta Q_\theta(s_m, a_m)$$

  end for
  Compute episodic cost and episodic reward
  if episodic reward improves **then**
    update summary by picking top $|S|$ subshots
  end if
  if For each mini batch **then**
    Back-propagate pseudo batch cost
  end if
end for

Algorithm 5 Actor Critic Framework

Initialize θ, w, γ and learning rates α_a, α_c.
for For each sliding window **do**
  Calculate S_p and S_f according to the position of W_s
  Get Q values from the Critic Network
  Get Policy distribution from Actor network
  Get Q values from the target Critic network
  for For each episode **do**
    Sample M actions from Policy distribution
    Actor cost calculation

$$cost_{ac} += \sum_{m=1}^{M} Q_c(s_m, a_m) \nabla_\theta log(\pi_a(s_m, a_m))$$

    Compute correction (TD error) for actions

$$\delta_m = R(S) + \gamma \sum_{m=1}^{M-1} Q_{w^-}(s_{m+1}, a_{m+1}) - \sum_{m=1}^{M-1} Q_w(s_m, a_m)$$

    Compute cost and reward R(S)

$$cost_{cri} += \delta_m \sum_{\substack{m=1, \\ a \in A}}^{M} \nabla_w Q_w(s_m, a_m)$$

  end for
  Compute episodic cost_{ac}, cost_{cri} and episodic reward of actor and critic
  if episodic reward improves **then**
    update summary by picking top $|S|$ subshots
  end if
  if For each mini batch **then**
    Back-propagate pseudo batch cost_{ac}, and cost_{cri}
  end if
end for

Bibliography

[1] S. N. Aakur and S. Sarkar, *A perceptual prediction framework for self supervised event segmentation*, in CVPR, 2019.

[2] M. Aghaei, *Social signal extraction from egocentric photo-streams*, in Proceedings of the 19th ACM International Conference on Multimodal Interaction, 2017, pp. 656–659.

[3] M. Aghaei, M. Dimiccoli, C. C. Ferrer, and P. Radeva, *Towards social pattern characterization in egocentric photo-streams*, Computer Vision and Image Understanding, 171 (2018), pp. 104–117.

[4] M. Aghaei, M. Dimiccoli, and P. Radeva, *All the people around me: face discovery in egocentric photo-streams*, in 2017 IEEE International Conference on Image Processing (ICIP), 2017, pp. 1342–1346.

[5] T. Ahmad, L. Jin, X. Zhang, L. Lin, and G. Tang, *Graph convolutional neural network for action recognition: A comprehensive survey*, IEEE Transactions on Artificial Intelligence, (2021).

[6] H. Alwassel, F. Caba Heilbron, and B. Ghanem, *Action search: Spotting actions in videos and its application to temporal action localization*, in Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 251–266.

[7] B. Amos, B. Ludwiczuk, M. Satyanarayanan, et al., *Openface: A general-purpose face recognition library with mobile applications*, CMU School of Computer Science, 6 (2016).

[8] L. Anne Hendricks, O. Wang, E. Shechtman, J. Sivic, T. Darrell, and B. Russell, *Localizing moments in video with natural language*, in ICCV, 2017.

[9] E. Apostolidis, E. Adamantidou, A. I. Metsai, V. Mezaris, and I. Patras, *Video summarization using deep neural networks: A survey*, Proceedings of the IEEE, 109 (2021), pp. 1838–1863.

[10] Y. M. Asano, C. Rupprecht, and A. Vedaldi, *Self-labelling via simultaneous clustering and representation learning*, in ICLR, 2020.

[11] N. Babaguchi, *Towards abstracting sports video by highlights*, in ICME, 2000.

[12] J. Bai, W. Wang, Y. Zhou, and C. Xiong, *Representation learning for sequence data with deep autoencoding predictive components*, in International Conference on Learning Representations, 2021.

[13] S. Bai, J. Z. Kolter, and V. Koltun, *An empirical evaluation of generic convolutional and recurrent networks for sequence modeling*, arXiv preprint arXiv:1803.01271, (2018).

[14] I. Beltagy, M. E. Peters, and A. Cohan, *Longformer: The long-document transformer*, arXiv preprint arXiv:2004.05150, (2020).

[15] B. L. Bhatnagar, S. Singh, C. Arora, C. Jawahar, and K. CVIT, *Unsupervised learning of deep feature representation for clustering egocentric actions.*, in IJCAI, 2017, pp. 1447–1453.

[16] A. Bifet and R. Gavalda, *Learning from time-changing data with adaptive windowing*, in Proceedings of SIAM international conference on data mining, 2007, pp. 443–448.

[17] M. Bolanos, M. Dimiccoli, and P. Radeva, *Toward storytelling from visual lifelogging: An overview*, IEEE Transactions on Human-Machine Systems, 47 (2016), pp. 77–90.

[18] S. Buch, V. Escorcia, C. Shen, B. Ghanem, and J. Carlos Niebles, *Sst: Single-stream temporal action proposals*, in Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2017, pp. 2911–2920.

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

[20] A. Cartas, M. Dimiccoli, and P. Radeva, *Batch-based activity recognition from egocentric photo-streams*, in Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2347–2354.

[21] P. Chang, M. Han, and Y. Gong, *Extract highlights from baseball game video with hidden markov models*, in Proceedings. International Conference on Image Processing, vol. 1, 2002, pp. I–I.

[22] Y.-W. Chao, S. Vijayanarasimhan, B. Seybold, D. A. Ross, J. Deng, and R. Sukthankar, *Rethinking the faster r-cnn architecture for temporal action localization*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 1130–1139.

[23] H.-W. Chen, J.-H. Kuo, W.-T. Chu, and J.-L. Wu, *Action movies segmentation and summarization based on tempo analysis*, in Proceedings of the 6th ACM SIGMM international workshop on Multimedia information retrieval, 2004, pp. 251–258.

[24] M. Chen, F. Wei, C. Li, and D. Cai, *Frame-wise action representations for long videos via sequence contrastive learning*, in CVPR, 2022.

[25] K. Choromanski, V. Likhosherstov, D. Dohan, X. Song, A. Gane, T. Sarlos, P. Hawkins, J. Davis, A. Mohiuddin, L. Kaiser, et al., *Rethinking attention with performers*, arXiv preprint arXiv:2009.14794, (2020).

[26] K. M. Choromanski, M. Rowland, and A. Weller, *The unreasonable effectiveness of structured random orthogonal embeddings*, Advances in neural information processing systems, 30 (2017).

[27] C. Cuevas, D. Quilon, and N. García, *Techniques and applications for soccer video analysis: A survey*, Multimedia Tools and Applications, 79 (2020), pp. 29685–29721.

[28] D. Damen, H. Doughty, G. M. Farinella, A. Furnari, E. Kazakos, J. Ma, D. Moltisanti, J. Munro, T. Perrett, W. Price, et al., *Rescaling egocentric vision: collection, pipeline and challenges for epic-kitchens-100*, International Journal of Computer Vision, 130 (2022), pp. 33–55.

[29] U. Damnjanovic, V. Fernandez, E. Izquierdo, and J. M. Martinez, *Event detection and clustering for surveillance video summarization*, in Image Analysis for Multimedia Interactive Services, 2008, 2008.

[30] K. Darabi and G. Ghinea, *Personalized video summarization using sift*, in ACM Symposium on Applied Computing, 2015.

[31] S. E. F. De Avila, A. P. B. Lopes, A. da Luz Jr, and A. de Albuquerque Araújo, *Vsumm: A mechanism designed to produce static video summaries and a novel evaluation method*, Pattern Recognition Letters, (2011).

[32] A. G. del Molino, X. Boix, J.-H. Lim, and A.-H. Tan, *Active video summarization: Customized summaries via on-line interaction with the user*, in AAAI, 2017.

[33] A. G. del Molino, J.-H. Lim, and A.-H. Tan, *Predicting visual context for unsupervised event segmentation in continuous photo-streams*, arXiv preprint arXiv:1808.02289, (2018).

[34] D. DeMenthon, V. Kobla, and D. Doermann, *Video summarization by curve simplification*, in ACMMM, 1998.

[35] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, *Bert: Pre-training of deep bidirectional transformers for language understanding*, arXiv preprint arXiv:1810.04805, (2018).

[36] C. Dias and M. Dimiccoli, *Learning event representations by encoding the temporal context*, in European Conference on Computer Vision, 2018, pp. 587–596.

[37] M. Dimiccoli, M. Bolaños, E. Talavera, M. Aghaei, S. G. Nikolov, and P. Radeva, *Sr-clustering: Semantic regularized clustering for egocentric photo streams segmentation*, Computer Vision and Image Understanding, 155 (2017), pp. 55–69.

[38] C. Ding, X. He, and H. D. Simon, *On the equivalence of nonnegative matrix factorization and spectral clustering*, in Proceedings of the 2005 SIAM international conference on data mining, SIAM, 2005, pp. 606–610.

[39] L. Ding and C. Xu, *Tricornet: A hybrid temporal convolutional and recurrent network for video action segmentation*, arXiv preprint arXiv:1705.07818, (2017).

[40] ——, *Video action segmentation with hybrid temporal networks*, (2018).

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

[42] F. Dirfaux, *Key frame selection to represent a video*, in ICIP, 2000.

[43] A. R. Doherty, S. E. Hodges, A. C. King, A. F. Smeaton, E. Berry, C. J. Moulin, S. Lindley, P. Kelly, and C. Foster, *Wearable cameras in health: the state of the art and future possibilities*, American journal of preventive medicine, (2013).

[44] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al., *An image is worth 16x16 words: Transformers for image recognition at scale*, arXiv preprint arXiv:2010.11929, (2020).

[45] D. K. Duvenaud, D. Maclaurin, J. Iparraguirre, R. Bombarell, T. Hirzel, A. Aspuru-Guzik, and R. P. Adams, *Convolutional networks on graphs for learning molecular fingerprints*, in Advances in Neural Information Processing Systems, C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, eds., vol. 28, Curran Associates, Inc., 2015.

[46] A. Ekin, A. M. Tekalp, and R. Mehrotra, *Automatic soccer video analysis and summarization*, IEEE Transactions on Image processing, (2003).

[47] E. Elhamifar, G. Sapiro, and R. Vidal, *See all by looking at a few: Sparse modeling for finding representative objects*, in CVPR, 2012.

[48] V. Escorcia, F. C. Heilbron, J. C. Niebles, and B. Ghanem, *Daps: Deep action proposals for action understanding*, in European Conference on Computer Vision, 2016, pp. 768–784.

[49] H. Fan, B. Xiong, K. Mangalam, Y. Li, Z. Yan, J. Malik, and C. Feichtenhofer, *Multiscale vision transformers*, in ICCV, 2021.

[50] A. Fathi, A. Farhadi, and J. M. Rehg, *Understanding egocentric activities*, in 2011 international conference on computer vision, IEEE, 2011, pp. 407–414.

[51] A. Fathi, J. K. Hodgins, and J. M. Rehg, *Social interactions: A first-person perspective*, in 2012 IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 1226–1233.

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

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

[54] A. Furnari, S. Battiato, and G. M. Farinella, *Personal-location-based temporal segmentation of egocentric videos for lifelogging applications*, Journal of Visual Communication and Image Representation, 52 (2018), pp. 1–12.

[55] J. Gao, C. Sun, Z. Yang, and R. Nevatia, *Tall: Temporal activity localization via language query*, in ICCV, 2017.

[56] Z. Gao, L. Wang, Q. Zhang, Z. Niu, N. Zheng, and G. Hua, *Video imprint segmentation for temporal action detection in untrimmed videos*, in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, 2019, pp. 8328–8335.

[57] A. Garcia del Molino, J.-H. Lim, and A.-H. Tan, *Predicting visual context for unsupervised event segmentation in continuous photo-streams*, in ACMMM, 2018.

[58] R. Girdhar, J. Carreira, C. Doersch, and A. Zisserman, *Video action transformer network*, in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 244–253.

[59] R. Girdhar and K. Grauman, *Anticipative video transformer*, in ICCV, 2021.

[60] GoPro. www.gopro.com. Accessed: 2018-09-03.

[61] M. Gygli, H. Grabner, H. Riemenschneider, and L. Van Gool, *Creating summaries from user videos*, in ECCV, 2014.

[62] M. Gygli, H. Grabner, and L. Van Gool, *Video summarization by learning submodular mixtures of objectives*, in CVPR, 2015.

[63] B. Han, J. Hamm, and J. Sim, *Personalized video summarization with human in the loop*, in WACV, 2011.

[64] I. U. Haq, K. Muhammad, T. Hussain, J. Del Ser, M. Sajjad, and S. W. Baik, *Quicklook: Movie summarization using scene-based leading characters with psychological cues fusion*, Information Fusion, (2021).

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

[66] P. Herruzo, L. Portell, A. Soto, and B. Remeseiro, *Analyzing first-person stories based on socializing, eating and sedentary patterns*, in International Conference on Image Analysis and Processing, 2017, pp. 109–119.

[67] H.-I. Ho, W.-C. Chiu, and Y.-C. Frank Wang, *Summarizing first-person videos from third persons' points of view*, in ECCV, 2018.

[68] M. Hoai and F. De la Torre, *Max-margin early event detectors*, International Journal of Computer Vision, 107 (2014), pp. 191–202.

[69] S. Hochreiter and J. Schmidhuber, *Long short-term memory*, Neural computation, (1997).

[70] R. Hou, R. Sukthankar, and M. Shah, *Real-time temporal action localization in untrimmed videos by sub-action discovery.*, in BMVC, vol. 2, 2017, p. 7.

[71] S. Huang, W. Wang, S. He, and R. W.H. Lau, *Egocentric temporal action proposals*, IEEE Transactions on Image Processing, (2017), pp. 1–1.

[72] iNFO Community, *Social Media Statistics and Facts in 2020*. https://www.youtube.com/watch?v=E5z8mOScYes. Accessed: 2021-11-29.

[73] M. Jain, J. Van Gemert, H. Jégou, P. Bouthemy, and C. G. Snoek, *Action localization with tubelets from motion*, in Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 740–747.

[74] M. Jain, J. C. Van Gemert, T. Mensink, and C. G. Snoek, *Objects2action: Classifying and localizing actions without any video example*, in Proceedings of the IEEE international conference on computer vision, 2015, pp. 4588–4596.

[75] S. Ji, W. Xu, M. Yang, and K. Yu, *3d convolutional neural networks for human action recognition*, IEEE PAMI, (2012).

[76] Z. Ji, K. Xiong, Y. Pang, and X. Li, *Video summarization with attention-based encoder-decoder networks*, IEEE TCSVT, (2019).

[77] H. Jin, Y. Song, and K. Yatani, *Elasticplay: Interactive video summarization with dynamic time budgets*, in ACMMM, 2017.

[78] T. Kanade and M. Hebert, *First-person vision*, Proceedings of the IEEE, (2012).

[79] H.-W. Kang and X.-S. Hua, *To learn representativeness of video frames*, in ACMMM, 2005.

[80] T. N. Kipf and M. Welling, *Variational graph auto-encoders*, arXiv preprint arXiv:1611.07308, (2016).

[81] N. Kitaev, Ł. Kaiser, and A. Levskaya, *Reformer: The efficient transformer*, arXiv preprint arXiv:2001.04451, (2020).

[82] K. M. Kitani, T. Okabe, Y. Sato, and A. Sugimoto, *Fast unsupervised ego-action learning for first-person sports videos*, in Proceedings of the IEEEconference on computer vision and pattern recognition, 2011, pp. 3241–3248.

[83] A. Kowdle and T. Chen, *Learning to segment a video to clips based on scene and camera motion*, in ECCV, 2012.

[84] R. Krishna, K. Hata, F. Ren, L. Fei-Fei, and J. Carlos Niebles, *Dense-captioning events in videos*, in Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 706–715.

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

[86] H. W. Kuhn, *The hungarian method for the assignment problem*, Naval research logistics quarterly, 2 (1955), pp. 83–97.

[87] S. Lan, R. Panda, Q. Zhu, and A. K. Roy-Chowdhury, *Ffnet: Video fast-forwarding via reinforcement learning*, in CVPR, 2018.

[88] L. J. Latecki, D. de Wildt, and J. Hu, *Extraction of key frames from videos by optimal color composition matching and polygon simplification*, in IEEE Workshop on Multimedia Signal Processing, 2001.

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

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

[91] Y. J. Lee and K. Grauman, *Predicting important objects for egocentric video summarization*, IJCV, (2015).

[92] J. Li, B. Li, and Y. Lu, *Hybrid spatial-temporal entropy modelling for neural video compression*, in ACMMM, 2022.

[93] S. Li, W. Li, C. Cook, C. Zhu, and Y. Gao, *Independently recurrent neural network (indrnn): Building a longer and deeper rnn*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 5457–5466.

[94] X. Li, B. Zhao, and X. Lu, *A general framework for edited video and raw video summarization*, IEEE TIP, (2017).

[95] Y.-L. Lin, V. I. Morariu, and W. Hsu, *Summarizing while recording: Context-based highlight detection for egocentric videos*, in ICCVW, 2015.

[96] Q. Liu and Z. Wang, *Progressive boundary refinement network for temporal action detection.*

[97] T. Liu and J. R. Kender, *An efficient error-minimizing algorithm for variable-rate temporal video sampling*, in Proceedings. IEEE International Conference on Multimedia and Expo, 2002.

[98] X. Liu, S. L. Pintea, F. K. Nejadasl, O. Booij, and J. C. van Gemert, *No frame left behind: Full video action recognition*, in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 14892–14901.

[99] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, *Roberta: A robustly optimized bert pretraining approach*, arXiv preprint arXiv:1907.11692, (2019).

[100] Z. Liu, Y. Nie, C. Long, Q. Zhang, and G. Li, *A hybrid video anomaly detection framework via memory-augmented flow reconstruction and flow-guided frame prediction*, in ICCV, 2021.

[101] G. Lu, W. Ouyang, D. Xu, X. Zhang, C. Cai, and Z. Gao, *Dvc: An end-to-end deep video compression framework*, in CVPR, 2019.

[102] M. Lu, Z.-N. Li, Y. Wang, and G. Pan, *Deep attention network for egocentric action recognition*, IEEE Transactions on Image Processing, 28 (2019), pp. 3703–3713.

[103] M. Lu, D. Liao, and Z.-N. Li, *Learning spatiotemporal attention for egocentric action recognition*, in Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, 2019, pp. 0–0.

[104] Z. Lu and K. Grauman, *Story-driven summarization for egocentric video*, in IEEE Conference on Computer Vision and Pattern Recognition, 2013.

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

[106] Y.-F. Ma, X.-S. Hua, L. Lu, and H.-J. Zhang, *A generic framework of user attention model and its application in video summarization*, IEEE TMM, (2005).

[107] Y.-F. Ma, L. Lu, H.-J. Zhang, and M. Li, *A user attention model for video summarization*, in ACMMM, 2002.

[108] L. Mackey, M. I. Jordan, R. Y. Chen, B. Farrell, J. A. Tropp, et al., *Matrix concentration inequalities via the method of exchangeable pairs*, The Annals of Probability, (2014), pp. 906–945.

[109] B. Mahasseni, M. Lam, and S. Todorovic, *Unsupervised video summarization with adversarial LSTM networks*, in CVPR, 2017.

[110] B. Mahasseni, X. Yang, P. Molchanov, and J. Kautz, *Budget-aware activity detection with a recurrent policy network*, arXiv preprint arXiv:1712.00097, (2017).

[111] J. Meng, H. Wang, J. Yuan, and Y.-P. Tan, *From keyframes to key objects: Video summarization by representative object proposal selection*, in CVPR, 2016.

[112] P. Mettes, J. C. v. Gemert, and C. G. Snoek, *Spot on: Action localization from pointly-supervised proposals*, in European conference on computer vision, Springer, 2016, pp. 437–453.

[113] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, *Asynchronous methods for deep reinforcement learning*, in ICML, 2016.

[114] P. Nagar, A. Rathore, C. Jawahar, and C. Arora, *Generating personalized summaries of day long egocentric videos*, IEEE PAMI, (2021).

[115] A. Y. Ng, M. I. Jordan, and Y. Weiss, *On spectral clustering: Analysis and an algorithm*, in Advances in neural information processing systems, 2002, pp. 849–856.

[116] C.-W. Ngo, Y.-F. Ma, and H.-J. Zhang, *Automatic video summarization by graph modeling*, in ICCV, 2003.

[117] X. V. Nguyen, J. Epps, and J. Bailey, *Information theoretic measures for clusterings comparison: is a correction for chance necessary?*, in ICML, 2009.

[118] M. Noroozi, A. Vinjimoor, P. Favaro, and H. Pirsiavash, *Boosting self-supervised learning via knowledge transfer*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 9359–9367.

[119] F. Paci, L. Baraldi, G. Serra, R. Cucchiara, and L. Benini, *Context change detection for an ultra-low power low-resolution ego-vision imager*, in European Conference on Computer Vision, 2016, pp. 589–602.

[120] J. Park, M. Lee, H. J. Chang, K. Lee, and J. Y. Choi, *Symmetric graph convolutional autoencoder for unsupervised graph representation learning*, in CVPR, 2019.

[121] N. Peyrard and P. Bouthemy, *Motion-based selection of relevant video segments for video summarization*, Multimedia Tools and Applications, (2005).

[122] H. Pirsiavash and D. Ramanan, *Detecting activities of daily living in first-person camera views*, in 2012 IEEE conference on computer vision and pattern recognition, IEEE, 2012, pp. 2847–2854.

[123] Pivothead. www.pivothead.com. Accessed: 2018-09-03.

[124] Y. Poleg, C. Arora, and S. Peleg, *Temporal segmentation of egocentric videos*, in CVPR, 2014.

[125] Y. Poleg, C. Arora, and S. Peleg, *Temporal segmentation of egocentric videos*, in CVPR, 2014.

[126] Y. Poleg, A. Ephrat, S. Peleg, and C. Arora, *Compact cnn for indexing egocentric videos*, in WACV, 2016.

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

[128] S. Pouyanfar, Y. Yang, S.-C. Chen, M.-L. Shyu, and S. Iyengar, *Multimedia big data analytics: A survey*, ACM computing surveys (CSUR), 51 (2018), pp. 1–34.

[129] A. Rahimi and B. Recht, *Random features for large-scale kernel machines*, Advances in neural information processing systems, 20 (2007).

[130] A. Rathore, P. Nagar, C. Arora, and C. Jawahar, *Generating 1 minute summaries of day long egocentric videos*, in ACMMM, 2019.

[131] S. Ren, K. He, R. Girshick, and J. Sun, *Faster R-CNN: towards real-time object detection with region proposal networks*, PAMI, (2017).

[132] A. Roy, M. Saffar, A. Vaswani, and D. Grangier, *Efficient content-based sparse attention with routing transformers*, Transactions of the Association for Computational Linguistics, 9 (2021), pp. 53–68.

[133] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, *Learning internal representations by error propagation*, tech. rep., California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

[134] S. Sarfraz, N. Murray, V. Sharma, A. Diba, L. Van Gool, and R. Stiefelhagen, *Temporally-weighted hierarchical clustering for unsupervised action segmentation*, in CVPR, 2021.

[135] S. Sarfraz, V. Sharma, and R. Stiefelhagen, *Efficient parameter-free clustering using first neighbor relations*, in CVPR, 2019.

[136] M. M. K. Sarker, H. A. Rashwan, E. Talavera, S. F. Banu, P. Radeva, and D. Puig, *Macnet: Multi-scale atrous convolution networks for food places classification in egocentric photo-streams*, in European Conference on Computer Vision, 2018, pp. 423–433.

[137] O. Sener and S. Savarese, *Active learning for convolutional neural networks: A core-set approach*, arXiv preprint arXiv:1708.00489, (2017).

[138] SenseCam. www.microsoft.com/microsoft-hololens Accessed: 2018-09-03.

[139] H. Seong, J. Hyun, and E. Kim, *Video multitask transformer network*, in Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, 2019, pp. 0–0.

[140] A. Sharaf, M. Torki, M. E. Hussein, and M. El-Saban, *Real-time multi-scale action detection from 3d skeleton data*, in 2015 IEEE Winter Conference on Applications of Computer Vision, 2015, pp. 998–1005.

[141] S. Sharma, L. E. Asri, H. Schulz, and J. Zumer, *Relevance of unsupervised metrics in task-oriented dialogue for evaluating natural language generation*, arXiv:1706.09799, (2017).

[142] Z. Shen, M. Zhang, H. Zhao, S. Yi, and H. Li, *Efficient attention: Attention with linear complexities*, in WACV, 2021.

[143] L. Shi, I. King, and M. R. Lyu, *Video summarization using greedy method in a constraint satisfaction framework*, in ICDS, 2003.

[144] Z. Shou, H. Gao, L. Zhang, K. Miyazawa, and S.-F. Chang, *Autoloc: Weakly-supervised temporal action localization in untrimmed videos*, in Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 154–171.

[145] Z. Shou, D. Wang, and S.-F. Chang, *Temporal action localization in untrimmed videos via multi-stage cnns*, in CVPR, 2016.

[146] P. Shukla, H. Sadana, A. Bansal, D. Verma, C. Elmadjian, B. Raman, and M. Turk, *Automatic cricket highlight generation using event-driven and excitement-based features*, in CVPR, 2018.

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

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

[149] ——, *Trajectory aligned features for first person action recognition*, Pattern Recognition, 62 (2017), pp. 45–55.

[150] H. S. Sokeh, V. Argyriou, D. Monekosso, and P. Remagnino, *Superframes, a temporal video segmentation*, 2018 24th International Conference on Pattern Recognition (ICPR), (2018), pp. 566–571.

[151] X. Song, K. Chen, J. Lei, L. Sun, Z. Wang, L. Xie, and M. Song, *Category driven deep recurrent neural network for video summarization*, in Multimedia & Expo Workshops, 2016.

[152] X. Song, L. Sun, J. Lei, D. Tao, G. Yuan, and M. Song, *Event-based large scale surveillance video summarization*, Neurocomputing, (2016).

[153] Y. Song, J. Vallmitjana, A. Stent, and A. Jaimes, *TVsum: Summarizing web videos using titles*, in CVPR, 2015.

[154] K. Soomro, H. Idrees, and M. Shah, *Action localization in videos through context walk*, in Proceedings of the IEEE international conference on computer vision, 2015, pp. 3280–3288.

[155] G. W. Stewart, *Matrix perturbation theory*, (1990).

[156] H. Sundaram and S.-F. Chang, *Video skims: Taxonomies and an optimal generation framework*, in Proceedings. International Conference on Image Processing, vol. 2, 2002, pp. II–II.

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

[158] C. Szepesvári, *Algorithms for reinforcement learning*, Morgan and Claypool, (2009).

[159] E. Talavera, M. Dimiccoli, M. Bolanos, M. Aghaei, and P. Radeva, *R-clustering for egocentric video segmentation*, in Iberian Conference on Pattern Recognition and Image Analysis, Springer, 2015, pp. 327–336.

[160] E. Talavera, C. Wuerich, N. Petkov, and P. Radeva, *Topic modelling for routine discovery from egocentric photo-streams*, Pattern Recognition, 104 (2020), p. 107330.

[161] H. Tang, V. Kwatra, M. E. Sargin, and U. Gargi, *Detecting highlights in sports videos: Cricket as a test case*, in 2011 IEEE International Conference on Multimedia and Expo, 2011.

[162] A. Tejero-de Pablos, Y. Nakashima, T. Sato, N. Yokoya, M. Linna, and E. Rahtu, *Summarization of user-generated sports video by using deep action recognition features*, IEEE TMM, (2018).

[163] M. D. C. Tongco, *Purposive sampling as a tool for informant selection*, Ethnobotany Research and applications, 5 (2007), pp. 147–158.

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

[165] G. Varol, I. Laptev, and C. Schmid, *Long-term temporal convolutions for action recognition*, PAMI, (2017).

[166] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, *Attention is all you need*, in Advances in neural information processing systems, 2017, pp. 5998–6008.

[167] A. Vyas, A. Katharopoulos, and F. Fleuret, *Fast transformers with clustered attention*, NeurIPS, (2020).

[168] C. Wang, S. Pan, G. Long, X. Zhu, and J. Jiang, *Mgae: Marginalized graph autoencoder for graph clustering*, in Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, 2017, pp. 889–898.

[169] L. Wang, Y. Qiao, and X. Tang, *Action recognition and detection by combining motion and appearance features*, THUMOS14 Action Recognition Challenge, 1 (2014), p. 2.

[170] P. Weinzaepfel, Z. Harchaoui, and C. Schmid, *Learning to track for spatio-temporal action localization*, in Proceedings of the IEEE international conference on computer vision, 2015, pp. 3164–3172.

[171] C.-Y. Wu, N. Singhal, and P. Krahenbuhl, *Video compression through image interpolation*, in ECCV, 2018.

[172] B. Xiong, G. Kim, and L. Sigal, *Storyline representation of egocentric videos with an applications to story-based search*, in Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 4525–4533.

[173] J. Xu, L. Mukherjee, Y. Li, J. Warner, J. M. Rehg, and V. Singh, *Gaze-enabled egocentric video summarization via constrained submodular maximization*, in CVPR, 2015.

[174] M. Xu, J.-M. Pérez-Rúa, V. Escorcia, B. Martinez, X. Zhu, L. Zhang, B. Ghanem, and T. Xiang, *Boundary-sensitive pre-training for temporal localization in videos*, in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 7220–7230.

[175] S. Yan, Y. Xiong, and D. Lin, *Spatial temporal graph convolutional networks for skeleton-based action recognition*, in AAAI, 2018.

[176] Y. Yan, E. Ricci, G. Liu, and N. Sebe, *Egocentric daily activity recognition via multitask clustering*, IEEE Transactions on Image Processing, 24 (2015), pp. 2984–2995.

[177] J. Yang, D. Parikh, and D. Batra, *Joint unsupervised learning of deep representations and image clusters*, in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 5147–5156.

[178] B. Yao, X. Jiang, A. Khosla, A. L. Lin, L. Guibas, and L. Fei-Fei, *Human action recognition by learning bases of action attributes and parts*, in ICCV, 2011.

[179] T. Yao, T. Mei, and Y. Rui, *Highlight detection with pairwise deep ranking for first-person video summarization*, in CVPR, 2016.

[180] S. Yeung, A. Fathi, and L. Fei-Fei, *Videoset: Video summary evaluation through text*, arXiv:1406.5824, (2014).

[181] P. Yousefi and L. I. Kuncheva, *Selective keyframe summarisation for egocentric videos based on semantic concept search*, in IPAS, 2018.

[182] G. Yu and J. Yuan, *Fast action proposals for human action detection and search*, in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 1302–1311.

[183] M. Zaheer, G. Guruganesh, K. A. Dubey, J. Ainslie, C. Alberti, S. Ontanon, P. Pham, A. Ravula, Q. Wang, L. Yang, et al., *Big bird: Transformers for longer sequences*, Advances in Neural Information Processing Systems, 33 (2020), pp. 17283–17297.

[184] M. Z. Zaheer, A. Mahmood, M. H. Khan, M. Segu, F. Yu, and S.-I. Lee, *Generative cooperative learning for unsupervised video anomaly detection*, in CVPR, 2022.

[185] Y. Zhai and M. Shah, *A general framework for temporal video scene segmentation*, in ICCV, 2005.

[186] K. Zhan, S. Faux, and F. Ramos, *Multi-scale conditional random fields for first-person activity recognition*, in 2014 IEEE international conference on pervasive computing and communications (PerCom), IEEE, 2014, pp. 51–59.

[187] X. Zhan, J. Xie, Z. Liu, Y.-S. Ong, and C. C. Loy, *Online deep clustering for unsupervised representation learning*, in CVPR, 2020.

[188] K. Zhang, W.-L. Chao, F. Sha, and K. Grauman, *Video summarization with long short-term memory*, in ECCV, 2016.

[189] S. Zhang, Y. Zhu, and A. K. Roy-Chowdhury, *Context-aware surveillance video summarization.*, IEEE TIP, (2016).

[190] X. Zhang, C. Xu, and D. Tao, *Context aware graph convolution for skeleton-based action recognition*, in CVPR, 2020.

[191] X.-D. Zhang, T.-Y. Liu, K.-T. Lo, and J. Feng, *Dynamic selection and effective compression of key frames for video abstraction*, Pattern recognition letters, (2003).

[192] Y. Zhang, M. Kampffmeyer, X. Zhao, and M. Tan, *Deep reinforcement learning for query-conditioned video summarization*, Applied Sciences, (2019).

[193] Y. Zhang, X. Liang, D. Zhang, M. Tan, and E. P. Xing, *Unsupervised object-level video summarization with online motion auto-encoder*, Pattern Recognition Letters, (2018).

[194] ——, *Unsupervised object-level video summarization with online motion auto-encoder*, Pattern Recognition Letters, (2020).

[195] B. Zhao, X. Li, and X. Lu, *Hierarchical recurrent neural network for video summarization*, in ACMMM, 2017.

[196] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, *Learning deep features for scene recognition using places database*, in NIPS, 2014.

[197] J. Zhou, K.-Y. Lin, H. Li, and W.-S. Zheng, *Graph-based high-order relation modeling for long-term action recognition*, in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 8984–8993.

[198] J. T. Zhou, J. Du, H. Zhu, X. Peng, Y. Liu, and R. S. M. Goh, *Anomalynet: An anomaly detection network for video surveillance*, IEEE Transactions on Information Forensics and Security, (2019).

[199] K. Zhou, Y. Qiao, and T. Xiang, *Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward*, in AAAI, 2018.

[200] C. Zhu, W. Ping, C. Xiao, M. Shoeybi, T. Goldstein, A. Anandkumar, and B. Catanzaro, *Long-short transformer: Efficient transformers for language and vision*, NeurIPS, (2021).