

Bangor University

DOCTOR OF PHILOSOPHY

Keyframe Summarisation of Egocentric Video

Yousefi, Paria

Award date:
2019

Awarding institution:
Bangor University

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 25. Apr. 2024

Bangor University

DOCTOR OF PHILOSOPHY

Keyframe Summarisation of Egocentric Video

Yousefi, Paria

Award date:
2019

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 29. Oct. 2019



PRIFYSGOL
BANGOR
UNIVERSITY

School of Computer Science and Electronic Engineering
College of Environmental Sciences and Engineering

Keyframe Summarisation of Egocentric Video

Paria Yousefi

Submitted in partial satisfaction of the requirements for the
Degree of Doctor of Philosophy
in Computer Science

Supervisor Professor Ludmila I. Kuncheva

February 2019

Declaration and Consent

Details of the Work

I hereby agree to deposit the following item in the digital repository maintained by Bangor University and/or in any other repository authorized for use by Bangor University.

Author Name: Paria Yousefi

Title: Keyframe Summarisation of Egocentric Video

Supervisor/Department: Professor Ludmila I. Kuncheva/ Computer Science

Funding body (if any): Project RPG-2015-188 funded by The Leverhulme Trust, UK.

Qualification/Degree obtained: PhD

This item is a product of my own research endeavours and is covered by the agreement below in which the item is referred to as “the Work”. It is identical in content to that deposited in the Library, subject to point 4 below.

Non-exclusive Rights

Rights granted to the digital repository through this agreement are entirely non-exclusive. I am free to publish the Work in its present version or future versions elsewhere.

I agree that Bangor University may electronically store, copy or translate the Work to any approved medium or format for the purpose of future preservation and accessibility. Bangor University is not under any obligation to reproduce or display the Work in the same formats or resolutions in which it was originally deposited.


Bangor University Digital Repository

I understand that work deposited in the digital repository will be accessible to a wide variety of people and institutions, including automated agents and search engines via the World Wide Web.

I understand that once the Work is deposited, the item and its metadata may be incorporated into public access catalogues or services, national databases of electronic theses and dissertations such as the British Library’s EThOS or any service provided by the National Library of Wales.

I understand that the Work may be made available via the National Library of Wales Online Electronic Theses Service under the declared terms and conditions of use (<http://www.llgc.org.uk/index.php?id=4676>). I agree that as part of this service the National Library of Wales may electronically store, copy or convert the Work to any approved medium or format for the purpose of future preservation and accessibility. The National Library of Wales is not under any obligation to reproduce or display the Work in the same formats or resolutions in which it was originally deposited.

Acknowledgements

 *Understanding vision and building visual systems is really understanding intelligence.*

— **Fei-Fei Li**

I would like to express my sincere gratitude and appreciation to Professor Ludmila (Lucy) I. Kuncheva, for the continuous support of my PhD study and related research, for her constant presence to give me advice, and share her immense knowledge with me.

I am also immensely grateful to The Leverhulme Trust, UK, who funded this research under project RPG-2015-188. This work would not have been possible without their financial support.

A very special gratitude goes out to my life-coach parents, and my beloved brother who have provided me with moral and emotional support all the way through.

Last but by no means least, I would like to extend my deepest gratitude to the most enthusiastic cheerleader, my darling husband for his enduring love and support, for keeping things going when I was down, and for always showing how proud he is of me.

Thank you all for being the important part of my journey, your presence is the solution!

Paria Yousefi

Statement of Originality

The work presented in this thesis/dissertation is entirely from the studies of the individual student, except where otherwise stated. Where derivations are presented and the origin of the work is either wholly or in part from other sources, then full reference is given to the original author. This work has not been presented previously for any degree, nor is it at present under consideration by any other degree awarding body.

Student:

Paria Yousefi

Statement of Availability

I hereby acknowledge the availability of any part of this thesis/dissertation for viewing, photocopying or incorporation into future studies, providing that full reference is given to the origins of any information contained herein. I further give permission for a copy of this work to be deposited with the Bangor University Institutional Digital Repository, the British Library ETHOS system, and/or in any other repository authorised for use by Bangor University and where necessary have gained the required permissions for the use of third party material. I acknowledge that Bangor University may make the title and a summary of this thesis/dissertation freely available.

Student:

Paria Yousefi

Abstract

Egocentric data refers to collections of images by a user wearing a camera over a period of time. The pictures taken provide considerable potential for knowledge mining related to the user's life, and consequently open up a wide range of opportunities for new applications on health-care, protection and security, law enforcement and training, leisure, and self-monitoring. As a result, large volumes of egocentric data are being continually collected every day, which highlights the importance of developing video analysis techniques to facilitate browsing the created video data. Generating condensed yet informative version from the original unstructured egocentric frame stream eases comprehending content, and browsing the narratives.

Given the great interest in creating keyframe summaries from video, it is surprising how little has been done to formalise their evaluation and comparison. The thesis first carries out a series of investigations related to automatic evaluation of video summaries, and their comparisons. A discrimination capacity measure is proposed as a formal way to quantify the improvement over the uniform baseline, assuming that one or more ground truth summaries are available. Subsequently, a formal protocol for comparing summaries when ground truth is available is proposed.

We noticed the mostly used benchmark summarisation methods: random, uniform, and mid-event selections, are weak competitors. Therefore, we propose a new benchmark method for creating a keyframe summary, called "closest-to-centroid". We examined the presented baseline method on 20 different image descriptors to demonstrate its performance against the typical choices of baseline methods.

Thereafter, the problem of selecting a keyframe summary is addressed as a problem of prototype (instance) selection for the nearest neighbour classifier (1-nn). Assuming that the video is already segmented into events of interest (classes), and represented as a data set in some feature space, we propose a Greedy Tabu Selector algorithm which picks one frame to represent each class. Summaries generated by the algorithm are evaluated on a widely-used egocentric video database, and compared against the proposed baseline (closest-to-centroid). The Greedy Tabu Selector algorithm leads to an improved match to the user ground truth, compared to the closest-to-centroid baseline summarisation method.

Next, a method for selective video summarisation of egocentric video is introduced. It extracts multiple summaries from the same stream based upon different user queries. The result is a time-tagged summary of keyframes related to the query concept. The method is evaluated on two commonly used egocentric and lifelog databases.

Further to this, it is noted that despite the existence of a large number of approaches for generating summaries from egocentric video, on-line video summarisation has not been fully explored yet. This type of summary can be useful where memory constraints mean it is not practical to wait for the full video to be available for processing. We propose a classification (taxonomy) for on-line video summarisation methods based upon their descriptive and distinguishing properties. Afterwards, we develop an on-line video summarisation algorithm to generate keyframe summaries during video capture. Results are evaluated on an egocentric database. The summaries generated by the proposed method outperform those generated by the two competitors.

Contents

| | |
|--|-----------|
| List of Abbreviations | xv |
| 1 Introduction | 1 |
| 1.1 Motivation | 1 |
| 1.2 Aims | 2 |
| 1.3 Objectives | 2 |
| 1.4 Contributions | 3 |
| 1.5 Publications Related to the Thesis | 4 |
| 1.6 Thesis Overview | 6 |
| 2 Background | 7 |
| 2.1 What is Video Summarisation? | 7 |
| 2.2 What is Egocentric Vision? | 7 |
| 2.3 First Person View Paradigm | 8 |
| 2.4 Challenges in Egocentric Vision | 10 |
| 2.5 A Review on Video Summarisation | 12 |
| 2.6 Conclusion | 19 |
| 3 Automatic Evaluation Protocol for Visual Comparison of Keyframe Summaries | 21 |
| 3.1 A Taxonomy of Evaluation Strategies | 21 |
| 3.2 A Review on Automatic Evaluation Frameworks | 24 |
| 3.3 Components of the Evaluation Protocol | 29 |
| 3.3.1 Feature Representation | 29 |
| 3.3.2 Similarity Metrics | 31 |
| 3.3.3 Matching Strategies | 32 |
| 3.3.4 Accuracy Metrics | 38 |
| 3.4 Evaluation Protocol | 39 |
| 3.4.1 What is a Good Evaluation Protocol? | 39 |
| 3.4.2 Data Set | 40 |
| 3.4.3 Discrimination Capacity | 40 |
| 3.4.4 Identifying the Protocol Components | 43 |
| Description of the Experiment | 43 |
| Evaluation of Distance Metrics and Thresholds | 44 |
| Evaluation of Feature Representation | 46 |

| | |
|--|------------|
| Evaluation of Matching Algorithms | 46 |
| 3.4.5 The Proposed Protocol | 49 |
| 3.5 An Example | 50 |
| 3.6 Conclusion | 54 |
| 4 Closest-to-Centroid Baseline Method | 55 |
| 4.1 Motivation | 55 |
| 4.2 Story-Line of Evaluating Keyframe Summarisations | 55 |
| 4.3 Closest-to-Centroid Baseline | 58 |
| 4.4 Feature Representations | 59 |
| 4.5 An Experiment with an Egocentric Video Database | 66 |
| 4.5.1 Data Set | 66 |
| 4.5.2 Matching Procedure | 67 |
| 4.5.3 Results | 68 |
| 4.6 Conclusion | 70 |
| 5 A Prototype Selection Technique for Video Summarisation | 73 |
| 5.1 Generic Summary | 73 |
| 5.2 Edited Nearest Neighbour Approach for Keyframe Selection . . | 76 |
| 5.2.1 Motivation | 77 |
| 5.2.2 Problem Statement | 78 |
| 5.3 Greedy Tabu Selector (One-per-Class) | 80 |
| 5.3.1 The Algorithm Details | 80 |
| 5.3.2 Greedy Tabu Selector for the Cartoon Example | 82 |
| 5.3.3 An Example with Generated Data | 83 |
| 5.4 Experimental Evaluation | 84 |
| 5.4.1 Feature Representations | 84 |
| 5.4.2 The Challenge of Egocentric Video Data | 86 |
| 5.4.3 Experimental Protocol | 91 |
| 5.4.4 Results | 92 |
| 5.5 Conclusion | 98 |
| 6 Selective Search for Producing Query-Based Summary | 101 |
| 6.1 Query-Based Video Summary | 101 |
| 6.2 Problem Statement | 102 |
| 6.3 Methodology | 103 |
| 6.3.1 Description of the Proposed Process | 103 |
| 6.3.2 Semantic Concept Search | 104 |
| 6.3.3 Occurrence-led Event Segmentation | 106 |
| 6.3.4 Keyframe Selection | 108 |
| 6.3.5 The Compass Summary Visualisation | 108 |
| 6.4 Experimental Results | 109 |
| 6.4.1 Data Sets | 110 |

| | | |
|----------|---|------------|
| 6.4.2 | Effectiveness of the Semantic Search Algorithm | 110 |
| 6.4.3 | Effectiveness of the Selective Summarisation Method | 111 |
| 6.5 | Summarisation Examples | 113 |
| 6.6 | Conclusion | 114 |
| 7 | On-Line Video Summarisation | 117 |
| 7.1 | Motivation | 117 |
| 7.2 | Problem Statement | 117 |
| 7.3 | A Classification of On-line Summarisation Methods | 118 |
| 7.4 | Methods Included in the Comparison Study | 121 |
| 7.4.1 | Shot Boundary Detection (SBD) | 121 |
| 7.4.2 | Zero-mean Normalised Cross-Correlation (ZNCC) | 121 |
| 7.4.3 | Diversity Promotion (DIV) | 121 |
| 7.4.4 | Submodular Convex Optimisation (SCX) | 122 |
| 7.4.5 | Minimum Sparse Reconstruction (MSR) | 122 |
| 7.4.6 | Gaussian Mixture Model (GMM) | 123 |
| 7.4.7 | Histogram Intersection (HIST) | 123 |
| 7.4.8 | Merged Gaussian Mixture Models (MGMM) | 123 |
| 7.4.9 | Sufficient Content Change (SCC) | 124 |
| 7.5 | Control-Charts Method for On-line Video Summarisation | 124 |
| 7.5.1 | Control-Charts Method (CCS) | 124 |
| 7.5.2 | Feature Representation | 130 |
| 7.6 | Experiments on Comparing Nine On-line Methods | 131 |
| 7.6.1 | Data | 131 |
| 7.6.2 | Evaluation Metrics | 133 |
| 7.6.3 | Experimental Protocol | 134 |
| 7.6.4 | Results | 137 |
| 7.7 | Experiments on the Proposed Method | 142 |
| 7.7.1 | Results on Synthetic Data | 142 |
| 7.7.2 | Results on VSUMM Videos | 144 |
| 7.8 | Experiments on Comparing the Descriptors | 145 |
| 7.8.1 | Extraction Time | 146 |
| 7.8.2 | Performance Measure | 146 |
| 7.8.3 | Quality of the Keyframe Summary | 147 |
| 7.9 | Conclusions | 147 |
| 8 | Control-Charts for Generating Budget-Constrained On-line Summary | 149 |
| 8.1 | Problem Statement | 149 |
| 8.2 | On-line Video Summarisation | 150 |
| 8.2.1 | Budget-Constrained On-line Video Summarisation | 150 |
| 8.2.2 | Choosing Parameter Values | 153 |

| | | |
|----------|---|------------|
| 8.2.3 | Feature Representation | 154 |
| 8.3 | Experimental Results | 154 |
| 8.3.1 | Data Set | 154 |
| 8.3.2 | Annotation Strategy | 154 |
| 8.3.3 | Rival On-line Video Summarisation Methods | 155 |
| 8.3.4 | Keyframe Selection Results | 155 |
| 8.4 | Conclusion | 157 |
| 9 | Conclusions and Future Work | 159 |
| 9.1 | Conclusions | 159 |
| 9.2 | Future Work | 161 |
| | References | 163 |

List of Figures

| | | |
|------|---|-----|
| 1.1 | A diagram of collaborative contributions to video summarisation. | 3 |
| 2.1 | Illustration of video summarisation types. | 7 |
| 2.2 | A classification of various topics related to video summarisations. | 13 |
| 3.1 | A taxonomy of evaluation strategies. | 22 |
| 3.2 | Example of a small bipartite graph. | 36 |
| 3.3 | An example of calculating Discrimination Capacity C_U | 42 |
| 3.4 | Discrimination capacity as a function of the threshold: 3 distances. | 45 |
| 3.5 | Discrimination capacity as a function of the threshold: Manhattan. | 47 |
| 3.6 | Visualisation of the C_U for the 6 matching methods. | 48 |
| 3.7 | Proposed protocol for Video #22, <i>DT</i> | 51 |
| 3.8 | Proposed protocol for Video #22, <i>OV</i> | 51 |
| 3.9 | Proposed protocol for Video #22, <i>STIMO</i> | 52 |
| 3.10 | Proposed protocol for Video #22, <i>VSUMM1</i> | 52 |
| 3.11 | Proposed protocol for Video #22, <i>VSUMM2</i> | 53 |
| 4.1 | Illustration of the results from the matching process, for video P03. | 68 |
| 4.2 | Averaged F-values for the proposed baseline method (20 features). | 70 |
| 5.1 | A classification of generic video summarisation methods. | 73 |
| 5.2 | Example: A day with 4 events (each row shows an event). | 78 |
| 5.3 | Two keyframe summaries in the example in Figure 5.2. | 78 |
| 5.4 | An example of 2D data labelled in three classes. | 84 |
| 5.5 | Keyframe selection for educational video. | 87 |
| 5.6 | Keyframe selection for Third Person Video. | 88 |
| 5.7 | Keyframe selection for egocentric video. | 89 |
| 5.8 | Improvement ΔF for three parameters and the 7 features. | 95 |
| 5.9 | Video P01 summaries: GT, CC and GTS. | 96 |
| 5.10 | Video P02 summaries: GT, CC and GTS. | 96 |
| 5.11 | Video P03 summaries: GT, CC and GTS. | 97 |
| 5.12 | Video P04 summaries: GT, CC and GTS. | 97 |
| 6.1 | A classification of query-based video summarisation methods. | 101 |
| 6.2 | Diagram of the proposed method. | 104 |
| 6.3 | Flowchart of the proposed method. | 104 |
| 6.4 | Illustration of the semantic search process. | 106 |

| | | |
|-----|---|-----|
| 6.5 | Mislabelled frames selected from videos P02 and P01. | 106 |
| 6.6 | An example of a compass summary for query ‘phone’. | 109 |
| 6.7 | A summary example for video P03, and query ‘food’. | 113 |
| 6.8 | A summary example for Subject 1-2, and query ‘coffee’. | 114 |
| 7.1 | A classification of on-line video summarisation methods. | 119 |
| 7.2 | Synthetic Data set#1. | 132 |
| 7.3 | Synthetic Data set#2. | 132 |
| 7.4 | Average ranks of the methods on the synthetic data sets. | 137 |
| 7.5 | Average F -measure per method for Video #21. | 139 |
| 7.6 | Comparison of ground-truth #3 and the SCX summaries (Video #29). | 141 |
| 7.7 | Synthetic Data sets #1 - #5. | 143 |
| 7.8 | F -measure versus the cardinality for CCS, MGMM, and SCX methods | 145 |
| 7.9 | Comparison of ground-truth #1 and the CC summaries (Video #47) | 145 |
| 8.1 | A sketch of the proposed on-line video summarisation method. . | 149 |
| 8.2 | An example summaries obtained by the BCC, SCX and UE methods | 157 |

List of Tables

| | | |
|-----|---|-----|
| 3.1 | Overview of existing automatic evaluation frameworks. | 26 |
| 3.2 | An example of the calculation of C_U for video #22. | 43 |
| 3.3 | Description of the proposed framework. | 49 |
| 3.4 | Calculation of the F -values and C_U for the 5 summarisation methods. | 53 |
| 4.1 | An overview of summarisation methods (rivals, and proposed). . | 56 |
| 4.2 | The main characteristics of the evaluated feature representations. | 61 |
| 4.3 | F-values for the 4 videos for the U, ME and CC summaries. . . . | 69 |
| 5.1 | Description of our method in terms of the spider diagram. | 80 |
| 5.2 | Cartoon example data | 83 |
| 5.3 | Feature descriptors | 85 |
| 5.4 | F -values and classification error for the UTEgo videos. | 93 |
| 5.5 | Correlation coefficients between F -values and the error rate. . . | 97 |
| 6.1 | Description of our method in terms of the spider diagram. | 103 |
| 6.2 | Result of the concept search algorithm. | 111 |
| 6.3 | Results of the Selective Summary process. | 112 |
| 7.1 | Descriptive classification terms of the comparative methods. . . | 125 |
| 7.2 | Parameters for the nine on-line methods. | 135 |
| 7.3 | The Pareto sets for the SCX method on Data set #1. | 136 |
| 7.4 | Method parameters tuned on VSUMM video #21. | 138 |
| 7.5 | F-values for the comparative on-line methods on VSUMM videos. | 140 |
| 7.6 | Results of paired-sample t-tests. | 143 |
| 7.7 | Comparison of extraction time and performance of features. . . | 146 |
| 8.1 | F-values for the comparative methods of BCC, SXC, and UE. . . . | 156 |

List of Abbreviations

| | |
|------------------------------------|--------------------------------------|
| A | Accuracy. |
| B | Initial buffer size. |
| $F = \{f_1, \dots, f_N\}$ | Data stream. |
| FN | False negative. |
| FP | False positive. |
| F | F-measure (F-value or F-score). |
| GT | Ground truth summary. |
| J | Approximation error. |
| S | Computer-generated summary. |
| TN | True negative. |
| TP | True positive. |
| U | Uniform summary. |
| γ | Similarity measure between two sets. |
| μ | Mean. |
| σ | Standard deviations. |
| θ | Threshold for keyframe similarity. |
| c_U | Discrimination capacity. |
| c | Class label. |
| d | Distance metric. |
| ms | Minimum shot length. |
| m | Number of matches. |
| t | Tabu parameter. |
| $\mathbf{V} = \{f_1, \dots, f_N\}$ | Video. |

| | |
|-------|--|
| AC | Agglomerative Clustering. |
| CC | Closest-to-centroid baseline method. |
| CCS | Control-charts method. |
| CNN | Convolutional Neural Network. |
| CUS | Comparison of User Summary. |
| DIV | Diversity promotion method. |
| FPV | First Person View. |
| GMM | Gaussian mixture model method. |
| GTS | Greedy Tabu Selector. |
| HIST | Histogram intersection method. |
| MGMM | Merged gaussian mixture models method. |
| MSR | Minimum sparse reconstruction method. |
| SBD | Shot boundary detection method. |
| SCC | Sufficient content change method. |
| SCX | Submodular convex optimisation method. |
| SIFT | Scale-Invariant Feature Transform. |
| SURF | Speeded Up Robust Features. |
| TPV | Third Person View. |
| UE | Uniform Events baseline method. |
| UTEgo | University of Texas Egocentric. |
| VSUMM | Video SUMMarization. |
| ZNCC | Zero-mean normalised cross-correlation method. |

Chapter 1

Introduction

1.1 Motivation

Wearable camcorders provide consumers with the ability to record their daily activities all day long. A large amount of research has demonstrated the potential use of the captured data for monitoring health-related behaviours, retrieving memory, remembrance cognitive training, preventing functional declines in elderly people, and navigation for blind people. The applications are not limited to research purposes anymore, as the affordability of such devices has grown rapidly in recent years. Mass-market consumers show a growing interest in recording and sharing every aspect of their lives, despite the fact that the recorded visual memories may have never be revisited. As a consequence, the volume of video information stored in on-line or off-line repositories is increasing. Having a voluminous and at the same time largely redundant stream of frames makes browsing the videos a disagreeable task. Therefore, in the past years, there has been a demand to enable automatic processing, browsing and retrieving of egocentric videos [118, 115, 93].

Thus far, the issue has been addressed in the literature in many aspects: from indexing and retrieval to summarising the content of the video. While the literature abounds with methods for summarisation, surprisingly little has been done towards developing a formal evaluation protocol. Moreover, even though the goal is to facilitate user's experience on extracting meaningful information from the recorded memory, the summarisation methods are often blind to the user's interests and preferences.

This thesis proposes solutions for the aforementioned issues in egocentric video streams. Due to the variety of subjects in this thesis, related works are reviewed separately for each subject.

1.2 Aims

The aim of this project is to address deficiencies in the state-of-the-art egocentric video summarisation evaluation frameworks and user requirements, by proposing new frameworks, methods and approaches.

1.3 Objectives

To accomplish the aim, we identify the following objectives:

1. Our first objective is to investigate the current approaches for evaluating and comparing video summarisation methods. Subsequently, we will aim to propose a new automatic evaluation protocol for video summarisation.
2. Currently, novel video summarisation methods are compared to simple (and weak) baseline methods such as: uniform, random, and mid-event selection. Our second objective is to propose a stronger baseline method for this collection.
3. Our third objective is to propose a new keyframe summarisation method which enforces coverage, diversity, and video story-telling. Unlike the existing methods, the new method should allow for distinguishing between similar *events* happening at different times not only between the selected keyframes.
4. We noticed that the overwhelming majority of keyframe summarisation methods offer a general summary. Arguing that such a summary would be of limited use, our fourth objective is to propose a new query-tailored video summarisation method.
5. Most video summarisation methods work off-line, after the whole video is available. Our fifth objective is to explore the current state-of-the-art

in on-line video summarisation, and contribute our own method that improves on the quality of the existing methods.

1.4 Contributions

The collaborative contributions presented in this thesis can be visualised as a mind map (Figure 1.1), and listed as follows:

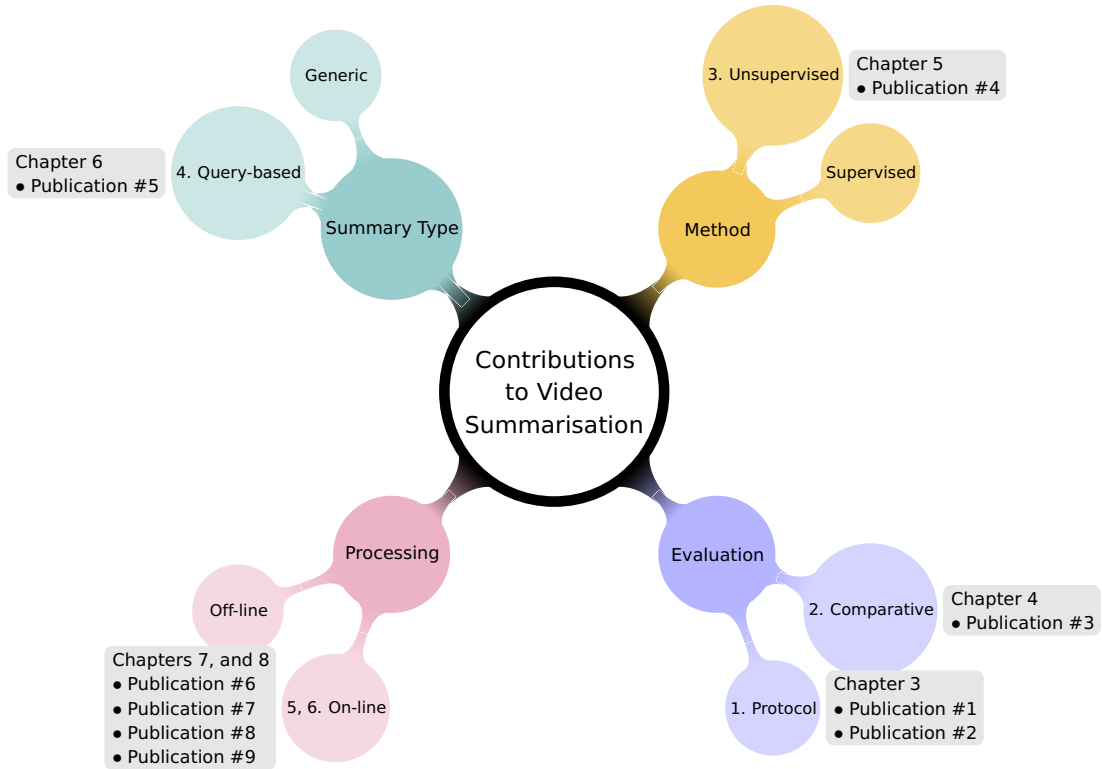


Figure 1.1: A mind map diagram of collaborative contributions to video summarisation.

1. We propose a generic evaluation protocol for objective comparison of an automatic keyframe summary, and a set of ground truth summaries. The development of the protocol was based upon selecting appropriate visual descriptors, a distance metric, and a matching strategy for pairing two summaries. Experiments were carried out on a real collection of video data (Objective #1- Publications #1, and #2).
2. Concerned by the lack of benchmark summarisation methods, we reinstated an old favourite, which we called “closest-to-centroid”. The presented baseline was empirically proven to be stronger than other typical choices of baseline methods such as uniform, random, and

- mid-event selection, tested on egocentric video data (Objective #2- Publication #3).
3. We developed a method for extracting a keyframe summary from a video using prototype selection for nearest neighbour classifiers. An edited nearest neighbour method was designed to ensure coverage, diversity of events, and video story-telling. The method was demonstrated on a cartoon example, and tested on egocentric videos (Objective #3- Publication #4).
 4. Acknowledging the limited value of an all-purpose keyframe summary, we proposed a pipeline enabling users to extract multiple keyframe summaries from the same stream based upon different queries. The presented selective summary system acquires a user's query, carries out a semantic concept search using a pre-trained Convolutional Neural Network and visualises the summary as a "compass". The system was evaluated on both egocentric videos and lifelogging photo-stream data (Objective #4- Publication #5).
 5. We offer an experimental comparison of on-line video summarisation methods. Subsequently, we propose a new generic on-line keyframe summarisation method. The method's performance was demonstrated on synthetic and real data (Objective #5- Publications #6, and #7).
 6. We contributed a budget-constrained on-line video summarisation algorithm for egocentric videos. The algorithm is based on control charts for change detection. Among its main assets are its low computational complexity, robustness with respect to the feature representation, and the accessibility of the keyframe summary at any moment of the recording. Suitable feature descriptors were selected through an empirical study on egocentric videos (Objective #5- Publications #8, and #9).

1.5 Publications Related to the Thesis

1. L. I. Kuncheva, P. Yousefi, and I. A. D. Gunn, On the Evaluation of Video Keyframe Summaries using User Ground Truth, *arXiv:1712.06899*, 2017.

2. I. A. D. Gunn, L. I. Kuncheva, and P. Yousefi, Bipartite Graph Matching for Keyframe Summary Evaluation, *arXiv:1712.06914*, 2017.
3. L. I. Kuncheva, P. Yousefi, and J. Almeida, Comparing Keyframe Summaries of Egocentric Videos: Closest-to-Centroid Baseline, *Proceedings of The Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA 2017)*, pages 1–6, Montreal, Canada, 2017. DOI: [10.1109/IPTA.2017.8310123](https://doi.org/10.1109/IPTA.2017.8310123).
4. L. I. Kuncheva, P. Yousefi, and J. Almeida, Edited nearest neighbour for selecting keyframe summaries of egocentric videos, *Journal of Visual Communication and Image Representation*, 52: 118–130, 2018. DOI: [10.1016/j.jvcir.2018.02.010](https://doi.org/10.1016/j.jvcir.2018.02.010).
5. P. Yousefi, and L. I. Kuncheva, Selective keyframe summarisation for egocentric videos based on semantic concept search, *Proceedings of the International Image Processing Applications and Systems Conference (IPAS 2018)*, pages 19–24, Sophia Antipolis, France, 2018. DOI: [10.1109/IPAS.2018.8708887](https://doi.org/10.1109/IPAS.2018.8708887).
6. C. E. Matthews, L. I. Kuncheva, and P. Yousefi, Classification and comparison of on-line video summarisation methods, *Machine vision and applications*, 30(3): 507–518, 2019. DOI: [10.1007/s00138-019-01007-x](https://doi.org/10.1007/s00138-019-01007-x).
7. C. E. Matthews, P. Yousefi, and L. I. Kuncheva, Using control charts for online video summarisation, *Proceedings of the International Joint Conference on Metallurgical and Materials Engineering (JCMME 2018)*, vol. 277, page 01012, Wellington, New Zealand, 2018. DOI: [10.1051/matecconf/201927701012](https://doi.org/10.1051/matecconf/201927701012).
8. P. Yousefi, L. I. Kuncheva, and C. E. Matthews, Selecting feature representation for online summarisation of egocentric videos, *Poster session presented at the International Conference on Computer Graphics and Visual Computing (CGVC 2018)*, Swansea, UK, 2018. DOI: [10.5281/zenodo.1475097](https://doi.org/10.5281/zenodo.1475097).
9. P. Yousefi, C. E. Matthews, and L. I. Kuncheva, Budget-constrained online video summarisation of egocentric video using control charts, *Proceedings of the International Symposium on Visual Computing (ISVC*

2018), pages 640–649. Springer, Cham, Las Vegas, USA, 2018. DOI: 10.1007/978-3-030-03801-4_56.

1.6 Thesis Overview

To achieve the aims outlined above, the thesis is organised as follows:

- Chapter 2 presents a background study upon which our research was drawn.
- Chapter 3 provides an empirical analysis on different components of a protocol for evaluating the outputs of keyframe summarisation algorithms.
- Chapter 4 introduces an empirically proven stronger baseline model for the comparative evaluation of keyframe summaries, instead of the widely used Uniform and Mid-event selections.
- Chapter 5 targets an instance selection for the nearest neighbour classifier to generate a keyframe summary, and proposes a generic video summarisation method.
- Chapter 6 proposes a method to extract a selective, time-aware keyframe summary for an egocentric video.
- Chapter 7 examines the performance of nine on-line video summarisation methods using synthetic data and real short videos.
- Chapter 8 proposes a fast and effective on-line summarisation method for egocentric videos.
- Chapter 9 gives the conclusions drawn overall from this study. It also indicates the possibility of future work in this area.

Chapter 2

Background

2.1 What is Video Summarisation?

Video summarisation is a compact way of representing a video by its key components, converting hours of video into limited series of keyframes or subshots. This makes video summary a subset of the original video, which may or may not be presented in a temporal order. Figure 2.1 illustrates two types of video summaries.

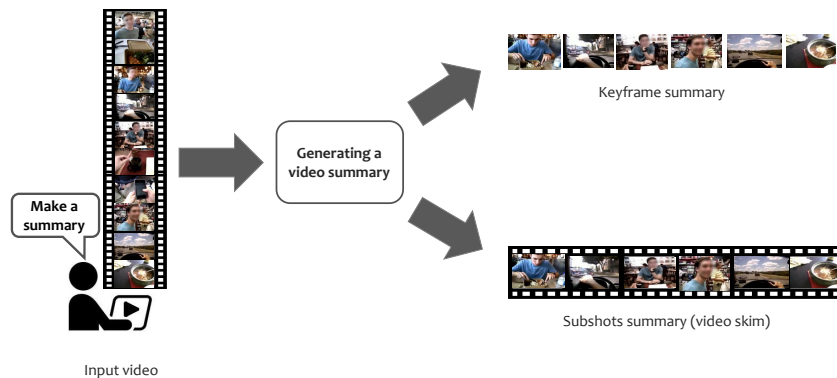


Figure 2.1: Illustration of video summarisation types.

2.2 What is Egocentric Vision?

Egocentric vision, also known as first person vision, First Person View (FPV), refers to video material captured by wearable cameras. Wearable cameras are small electronic devices that can be placed on the head of the user,

clipped to the user's clothes, or worn as accessories. Images are automatically recorded from the first person perspective without user intervention.

Lifelogging is the process of continuously recording the user's everyday experiences, via wearable cameras. Data acquired over a period of time provide knowledge on the wearer's life, and consequently enable many applications. Microsoft's SenseCam is one of the lifelogging devices, which has been commonly employed in health related research for several years [68, 13, 26, 130, 27, 132, 83]. Widespread use of wearable devices for health related research is limited, due to obstacles such as: ethical issues, privacy concerns, difficulty in collecting a large number of samples, the time-consuming process of manual analysis, and incorrect positioning of the camera [45]. Other examples of wearable camcorders are: Narrative Clip (as photographic cameras); and GoPro, Google Glass, MeCam, and Looxcie (as video cameras). Photographic cameras have a low temporal resolution which allows for acquiring images over a long period of time without the need to recharge the battery. However, motion features cannot be reliably estimated because of abrupt appearance changes. On the other hand, video cameras have relatively high temporal resolutions, which allow for capturing the fine temporal details of interactions. However, due to the abrupt head movements of the camera wearer, assessing the global motion of the wearer is difficult.

2.3 First Person View Paradigm

Some applications of FPV cameras are listed below:

1. *Summarising a person's life; applications for memory retrieval.* To support a person with dementia, egocentric images captured by an individual can be used to enhance memory of their recent activities or forgotten events. The narratives were employed by Piasek et al. [132] to improve Cognitive Stimulation Therapy (CST) [151] for patients in the early stage of dementia and carer, to engage them in meaningful discussions (on images of patient's life).

Hodges et al. and Browne et al. [68, 27] conducted single-case studies to show using wearable cameras would improve the autobiographical memory¹ in comparison with reviewing written diaries for patients in the early stage of Alzheimers disease (with minor problems with cognition) and amnesia. The effectiveness of using video summarisation as a memory aid to enhance users experience has been demonstrated empirically [87].

2. *Extracting the nutritional information about the user's diet.* Being able to automatically record and analyse visual diaries, the nutritional data can be created based on: types of nutritious substances; locations of use; and the conditions of usage by the camera wearer. Using wearable cameras to capture an individual's nutritional data seems convincing, as the study [95] shows people tend to under-report their daily caloric consumption when they document them manually. Therefore, Bolaños and Radeva [23] proposed a method to localise and recognise food in egocentric narratives. Bolaños et al. [20] further explored recognising food ingredients in conventional images, which makes it possible to calculate the overall amount and variety of nutrients. However, this seems to be a challenge using low quality egocentric images.
3. *Monitoring diets of athletes.* Wearing egocentric cameras assisted athletes and sport dietitians in increasing the accuracy of dietary reports and consequently their assessments, by reducing the likelihood of misreporting the energy intake [122]. The narratives also provide beneficial information on dietary intake patterns and emphasise a significant under-reporting of calories consumed by athletes (e.g. eating leftover food).
4. *Acquiring information on sedentary behaviours.* Issues such as: memory recall errors, identification accuracy errors, or impracticality in free-living, are common when estimating sedentary behaviours based on self-report tools or hip-worn accelerometers [140, 110, 41]. Because of these difficulties, lifelog devices are deployed (either along with accelerometer or by themselves) to obtain direct observations of a person's stationary-

¹Autobiographical memory is a person's recollection of past incidents and events [27].

behaviour [83, 46, 65]. Acquired images give evidence on the type, context, and duration of sedentary behaviours.

5. *Discovering social interactions and relations.* Exploring the social life of an individual can provide useful information about the person's mental and physical health [159, 86]. In addition to detecting and generating a diary of interactions, wearable cameras can be used for memory reinforcement. Wearable cameras provide a personal platform to observe each person's social interactions (from their point of view), and therefore attracted many researchers to this area [6, 29, 3, 4, 2].
6. *Daily use of body-worn cameras related to law enforcement and security.* Currently, many police officers in the United States, Canada, and parts of Asia and Europe (including the United Kingdom) are recording their day to day operations using body-worn cameras. There are many advantages reported from using body-cams in which police officers are involved [36]. The narratives are real-life situations, which can be employed for training purposes. Employing body-worn cameras can protect officers from false allegations, and also influence good behaviours for both parties (police and those being recorded). The recorded events or crimes can be documented and recalled during investigation and prosecution periods.
7. *Creating photo albums of authentic memory of holidays.* Egocentric narratives recorded on vacation are structurally different from daily living videos. The video mainly contains images of picturesque scenes, locations, and landmarks of historical significance. These narratives can be used for personal reasons, such as sharing with family, friends, and the general public [14].

2.4 Challenges in Egocentric Vision

In recording daily life, a camera wearer has no specific intention on capturing every single frame. As a result, the acquired frames may be blurred; underexposed or overexposed; containing tilted, occluded, or off-centred objects; holding poor composition; or covering non-informative content (e.g. sky, wall, or ground). In order to clear out frames of low visual

quality, some authors edited the frames-stream before any further structural processing. The discrimination can be content blind or based on the visual information. The problems are solved either by a binary semi-supervised technique (e.g. training an ‘informative vs non-informative’ classifier [96]), or by an unsupervised technique (e.g. kernel domain adaptation [172]). The former requires the frames to be manually labelled into informative and non-informative for training a discriminative binary classifier [96]. While the latter attempts to approximate the distribution gap between third-person-taken images collected from the web, and egocentric frames. It estimates the likelihood of an egocentric frame being under the distribution of web images based on the nearest neighbours’ distances [172].

FPV streams are characterised by a smooth transition across scenes and continuous changes of camera wearer’s focus points. Some researchers initially partitioned the unconstrained frame stream into events. An event or segment is defined as a section of the video enclosed between two time stamps. The advantage of adding event segmentation into video processing is that the summary can include multiple occurrences of an object or person, interacted with the camera wearer in the different sequence of the video. Having that, the summarisation will cover similar events happened in different time stamps.

The most conventional way to solve the problem of event segmentation, is to group contiguous frames with similar global appearance together (even with few unrelated frames in between) [92, 21]. The similarity of pairwise frames is calculated using their visual feature distances. The visual representations of frames can be based on colour (e.g. colour histogram); or a Convolutional Neural Network (CNN, or ConvNet). After similarities are measured, visually related frames are grouped together using clustering methods such as Agglomerative Clustering (AC). The grouping strategy is always accompanied by a temporal analysis to prevent time-related inconsistency of the segmentation.

An example of event segmentation method is SR-Clustering [42]. The SR-Clustering method automatically determines the events using contextual and semantic information of frames. To Obtain semantic information, similar concepts are integrated leveraging the posterior temporal segmentation. Using contextual attributes with temporal coherence, frames with similar visual and semantic features are clustered together. AC is combined with the concept drift change detection technique, called ADaptive WINdowing (ADWIN) [17] to group frames. The ADWIN technique uses sliding windows whose size is recomputed directed by the rate of change in the data. The Graph-Cut algorithm is applied to obtain a trade off between these two approaches, which requires initialising the minimum number of frames in events. The R-Clustering method has a similar process where the semantic information is eliminated [155].

Other ways of partitioning videos into events (or subshots) are to detect generic patterns of camera wearers: activities (this is also called ‘ego-activity’) [103], or behaviours [161]. For the former category, ego-activities are classified into: *static*, meaning body or head are not undergoing a significant motion; *transit*, meaning a physical movement from one point to another; and *moving the head*, meaning wearer attention is changed to different part of the scene. For the latter, behaviour patterns are classified into: *body motion*, including *walking*, *running*, *on transit*, and *wandering*; and *body still*, containing *static*, and *looking around*. The approaches proposed to analyse camera wearer’s motion patterns assess: a histogram of motion features (e.g. dense optical flow [98]) and a blurriness score [37] of frames. Thereafter, it is followed by training a classifier (e.g. Support Vector Machine (SVM)). This strategy is also associated with temporal analysis to connect neighbour frames.

2.5 A Review on Video Summarisation

Truong and Venkatesh [158] described and categorised existing solutions for non-FPV² video summarisation. Comprehensive surveys also exist for

²The non-FPV data stream includes a Third Person View (TPV)

application- or approach-specific solutions e.g. egocentric videos [115] and lifelogging [19], as well as context-based summaries [80].

Figure 2.2 shows a spider diagram of various topics related to video summarisation. From now on, in all diagrams, we use colour-coded nodes from Figure 1.1 to signal transition between subjects. Information associated with the various topics related to video summarisation methods are displayed in golden nodes fading to grey. Transitions to the collaborative contributions are changed to: blue-green in colour for summary type, blush colour for processing, and violent in colour for evaluation.

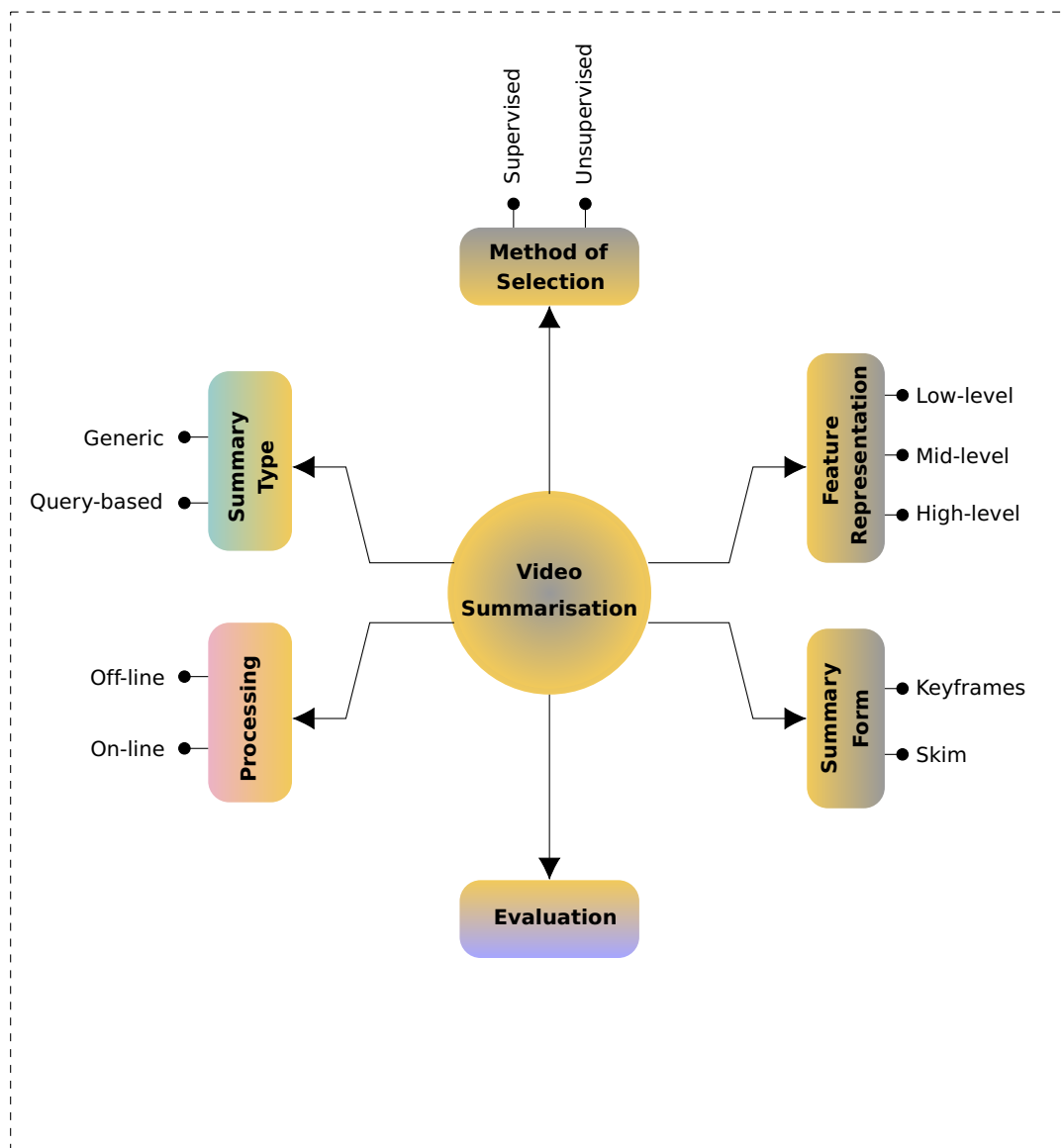


Figure 2.2: A classification of various topics related to video summarisation methods.

Below we explain each leaf of the diagram:

- *Summary form.* Generated summary can be represented as a series of still images (static keyframe summary) [1, 137, 76, 84, 9, 92, 10, 96, 51, 21, 112, 14], or a sequence/collection of subshots (dynamic video skims) [160, 103, 9, 60, 61, 174, 128, 175]. The appropriate choice for the form of the summary depends on the application. This review is primarily focused on methods that generate static keyframe sets.

- *Feature representations.* In most works, video frames are represented as vectors in an n -dimensional feature space, $\mathbf{x} \in \mathbb{R}^n$. The choice of feature space can be a combination of low-level features (e.g. moments and histograms of colour spaces) [103, 60, 173]; mid-level features (e.g. complex CNN) [174, 61, 21, 175, 96, 162, 10, 51]; or high-level features [103, 92, 144, 145]. High-level features can be related to semantic information [60]; can be region features describing the objects and faces which the camera wearer interacts with [103, 92]; or can be concept-oriented features with visual descriptors [144, 145]. Examples of low-level visual features used in the literature are RGB histograms [173]; HSV histograms [1, 9, 137]; texture descriptors such as Improved Fisher Vector (IFV) [173], Gist [173, 144, 145], Local Binary Patterns (LBP) [144, 145], dense Scale-Invariant Feature Transform (SIFT) [173]; CENTRIST feature space [112]; and dense optical flow as motion descriptors [173].

Appropriate features are selected based on the application of use. For instance, in an application of on-line video summarisation, features that are less computationally expensive and require less memory are preferable. Various choices of feature representations suitable for the different parts of our research will be discussed in more detail in the respective chapters.

- *Method of selection.* A summary consists of a concise number or sequences of frames selected to represent a video. The selection can be executed through an unsupervised or a supervised method.

In the unsupervised group, the keyframe/shot selection is not guided by previously available data with examples of good summaries. Many such

methods were developed in the past [158]. Assuming that the frames are represented by their visual information, the criterion for selecting keyframes can be based on different ideas, possibly overlapping, as detailed below:

- *Grouping strategy*: For this category [63, 119, 62, 53, 40, 106], frames are presented by feature vectors, and the similar ones are grouped together to form clusters. Choices of the clustering algorithm vary according to authors' preferences. The number of clusters can be: set *a priori* [178, 67, 127]; *computed*, as a number of sufficient content changes³ in sequences of frames [40]; or *determined automatically by the clustering algorithm* [106].
- *Sufficient content change*: Similar to the grouping strategy, close-content frames can also be identified by detecting significant changes in the content information [176, 180, 75, 76, 8, 49, 59]; cumulative curve of frame differences [55]; or motion activities [43].

Finally, for both categories above, keyframes are produced either through a naive selection related to the frame location within a shot or based on a predefined objective. The former selection group is as simple as choosing the first frame [120, 1], the middle frame [137, 55, 8, 106], or the last frame of each shot. While this approach is fast, the extracted keyframes may not capture the informative visual content. Moreover, the frames located at the beginning and the end of the shot are often not stable.

For the sufficient content change method, the objectives are to optimise some distribution or characteristic function such as: a similarity distance [184, 178, 119, 40, 84, 21, 127]; a discriminative distance⁴ [35]; the entropy of colour distribution [164]; a ranking score [43, 59]; random walk [21]; or Markov Random Field (MRF) [173]. For application on

³This is obtained by calculating the pairwise Euclidean distances between consecutive frames. The number of clusters is incremented, when the pairwise distance is positioned above a threshold.

⁴The discriminative distance is calculated as a ratio (or subtractive) value to be maximised. Assuming to have a video segmented into shots, the ratio is computed for every frame to measure a degree of its similarity with the other frames in that segment, to a degree of dissimilarity with the other segments (discrimination).

egocentric data set, a distribution model is adopted in most works [84, 21].

- *Cross-Correlation*: The criterion in this category is to minimise the cross-correlations among selected keyframes, which ultimately increases the dissimilarity of the extracted keyframes from one another [11, 47].
- *Reconstruction error*: This category formulates video summarisation as a minimum reconstruction error problem, which maximises the capability of reconstructing the original frames from as few as possible selected keyframes [88, 89, 112].
- *Ranking strategy*: The Ranking strategy optimises an energy function, which comprises of objective functions (metrics). The energy function sequentially scores frames or subshots based on how well they present the predefined objectives. This category mainly used for egocentric summarisation, e.g. optimising energy function [103, 60, 92, 96, 162], frames-ranked maximisation with heuristic parameters [14, 175].
- *Motion pattern*: For this method, the video frames are described by their motion features, e.g. optical flow [169]. A motion metric based on the feature representation is computed. Then motion levels of consecutive frames are analysed, and the frames with minimum motion (local minima) are extracted as keyframes.

Supervised methods, on the other hand, train a classifier with human-edited summary instances, either at frame or video levels, to learn how to produce video summaries related to their predefined criteria. In short, supervised summarisation acts as a structured binary prediction vector to indicate whether a frame is to be selected or not.

- *Category-Specific training*: Knowing the category of a video (related to the visual content, e.g. birthday party), Potapov et al. [134] trained a linear SVM classifier with positive video samples from the same category, and negatives from the other categories (binary for each category).
- *Sequential diversity model*: Recent approaches such as the ones proposed by Zhang et al. [181], and Gong et al. [56] model sequential diversity by probabilistic distributions. Zhang et al. [181] used Long

Short Term Memory (LSTM) to model the sequential dependency of video summary, followed by Determinantal Point Process (DPP) to collect diverse frames. Sharghi et al. [144, 145] combined Sequential and Hierarchical Determinantal Point Process (SH-DPP) to select diverse shots related to the user's query. Both LSTM and DPP are trained by summaries based on human annotations. The main drawback of applying DPP to produce a video summary is its high computational cost, particularly for a long video [144].

- Submodular maximisation: This method formulates video summarisation task as a subset selection problem. The final summary is selected to maximise the predefined objective functions. Using submodular optimisation, summary is generated by maximising multiple objective functions at the same time. Objective functions are often computed by calculating mutual information, or learning objective weights (learn to rank). The method used in [61, 174] to generate summary for egocentric video data. Computational cost for generating a video summary using submodular functions is also high when the video is long [144].

- *Summary type.* Summary can be either a generic one, where a single summary is produced by an automatic method or a query-based one, where multiple summaries can be extracted based upon different user's queries. Chapters 5, and 6 study the properties of each type in details.

- *Processing.* Video can be analysed and summarised after recording the entire data set in an off-line setting or during the recording in an on-line setting. For the traditional video data (TPV), on-line term may also refer to producing a 'good quality' summary within a reasonable time, which allows for on-line usage [8, 9]. Properties of on-line video summarisation methods will be discussed in Chapter 7.

- *Evaluation.* After proposing a new video summarisation method, its performance must be evaluated. This topic will be discussed in details in Chapters 3, and 4.

Researchers may focus on: using more sophisticated selection methods relying on commonly used features; or designing more complicated features to encode the semantic contents of frames (in the videos), and instead proposed relatively simpler methods for summarisation. For instance, Otani et al. [127] stacked visual representations of frames which were uniformly sampled from a video, and combined those with the corresponding sentence representations related to the description of that segment of the video. They trained a deep neural network (using positive and negative examples) to map frames into a similar semantic space including objects, actions and scenes. To generate a generic summary, a given video is uniformly sampled creating video segments. For each segment, deep features are extracted, mapped into a semantic space, and clustered. The summary is the selection of frames corresponding to cluster centres.

A summarisation method can be tested on collections of videos to analyse its performance. The collection can be obtained from third person perspectives (denoted as TPV), e.g. Disneyland data set collected from YouTube [173]; or recorded by first person (wearable) camera (denoted as FPV), e.g. cultural heritage [162], or University of Texas Egocentric (UTEgo⁵) [145]. Studying on first person video summarisation, an acceptable collection must contain long videos recorded in unconstrained environments by people with wearable cameras preferably attached to their head [115]. Alternatively, the camera may be clipped to their clothes, hung around their neck, or attached in another way to the upper part of the wearer's chest. Such a collection have advantages over others obtained by mobile telephones and hand-held cameras.

It is important to note that videos are recorded for different purposes, which may require different setup for summarisation in order to extract relevant data. For instance, the summarisation approach proposed by Xiong et al. [173] requires a prior knowledge about attraction locations and events, to collect training sets of images and videos from Google, Flickers and YouTube.

⁵<http://vision.cs.utexas.edu/projects/egocentric/>

These applications are either related to tourist attractions [173, 162] or daily living [145].

2.6 Conclusion

In this chapter, we classified video summarisation methods based on their main topics. We also illustrated the wide range of applications of FPV cameras, and then described challenges of working with egocentric data streams.

Even though a growing amount of literature is dedicated to non-FPV video summarisation, the proposed TPV summarisation approaches for traditional videos may not be suitable for FPV [115, 174]. In TPV, a stream of frames are captured from a stable point of view, while in FPV the task becomes a lot more complicated as explained in Section 2.4.

Video summarisation of egocentric data is a relatively new subject of research, therefore there are several areas that can be improved. First, evaluation of keyframe summaries extracted by different algorithms is an important but a little-discussed problem. A further investigation on this area is recommended. Second, current video summarisation methods may not be able to enforce diversity of the summary between events. So far, selection methods related to diversity are concerned with visual differences among the selected keyframes. A good summarisation method should emphasises the difference between the events being represented within the summary, which is not necessarily equivalent to visual differences between the selected frames. Third, reviewing the literature, we came to the conclusion that generic summaries may not be very useful, hence selective summaries should be developed instead. Finally, we identified on-line video summarisation as an interesting direction which is likely to grow in the near future.

The above areas of improvement were addressed in the following chapters starting with evaluation of keyframe summaries.

Chapter 3

Automatic Evaluation Protocol for Visual Comparison of Keyframe Summaries

3.1 A Taxonomy of Evaluation Strategies

The success of a system that automatically summarises a video must be demonstrated by evaluating its results. Thus far, video summarisation research offers three types of evaluation (Figure 3.1):

- *Descriptive*. Typically, the proposed summarisation method is implemented on a few videos and the created summaries are either displayed or described, leaving the judgement to the readers. Some authors may explain the advantages of their proposed summarisation method/algorithm. This simple form of evaluation was a popular model for the past decade. Nowadays, it is usually followed by other forms of evaluation. This form of evaluation is insufficient in most cases, and gives little ground to generalise the performance of the proposed method outside of those few videos.
- *Quantitative metrics*. This form of evaluation uses a predefined fidelity criterion that is computed from the extracted keyframes, and the original set of frames. The fidelity metric is often linked to the proposed method [158].
- *User-involved strategy*. This evaluation measures whether the obtained summary maps well to user judgement. Users can be included to create a

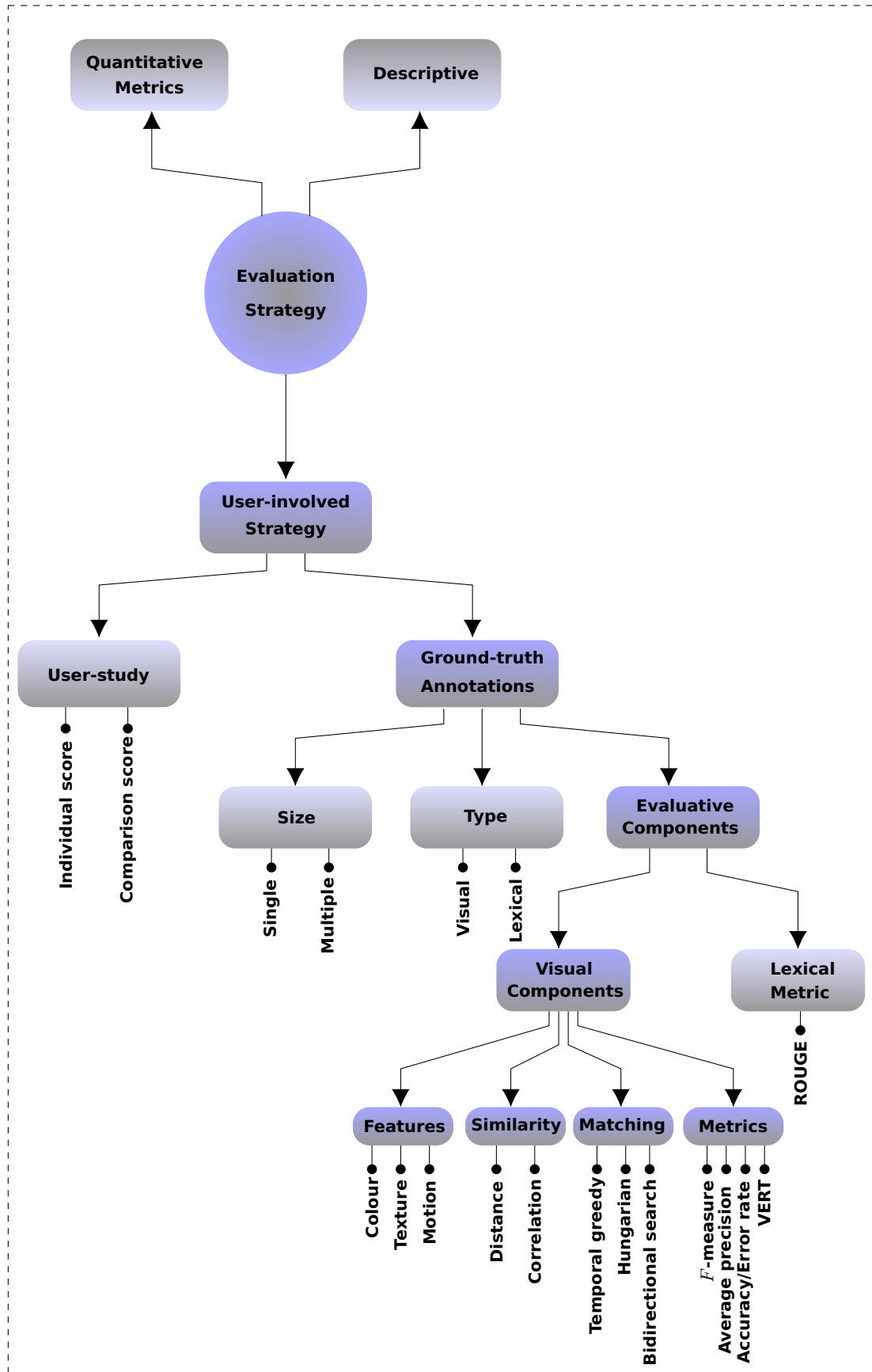


Figure 3.1: A taxonomy of evaluation strategies in the video summarisation literature.

summary prior to producing a computer-generated summary (*Ground-truth annotations*), or to judge a computer-generated summary (*User-study*). This form of evaluation is the most reasonable type.

- *User-study*. After video summaries are generated, independent users are employed to judge the quality of the automatic summaries. The original videos are usually presented to the users in a speed-up version. The judgements can be expressed by satisfactory scores of the summary, either in a numerical grading form [49, 84, 96] or a lexical grading form [92]. Each summary can be rated individually [96] or in comparison against: other existing summarisation techniques [49, 103, 92, 175, 96]; or baselines [84, 161, 21, 14], in blind-taste tests¹. Querying user’s opinions can also be in the form of selecting the best summary between two summaries (based on a criterion given to the users) [103, 161, 21, 14, 175]. This type of evaluation is subjective, difficult to replicate, and time-consuming. It also fails to show what a ‘good’ summary is.

- *Ground-truth annotations*. This term refers to creating a ground-truth keyframe summary manually. Evaluation based on ground-truth annotations has the advantage of being efficient and repeatable.

The number of annotators can vary from a *single* user [77] to *multiple* users [40, 107, 81]; termed as *size* in the diagram. Having multiple ground truth summaries is a typical annotation choice in summarisation, due to the discrepancies among users on selecting a unique summary from the same video stream. Consequently, the comparison strategies between the computer-generated frames, and the multiple user-generated frames are divided into two groups [70]. The first group compares every individual ground truth summary with the computer-generated summary, and then computes the average of the overall performance scores [40, 49, 112]. The second group aggregates the multiple summaries to produce a final ground

¹Blind-taste test is an evaluation procedure where a user selects a preferred summary among two summaries produced by different methods from the same video, without having information on the sources. Typically the proposed summary is placed randomly along with its competitor summaries.

truth summary that has the maximum agreement with all users. The final ground truth summary is then compared against the computer-generated summary [56].

Ground-truth annotations can be collected either as: *visual*, in which human annotators directly select frames to make the video summary [40, 84, 60, 174, 175] or *lexical*, in which the annotators (indirectly) summarise the video by writing on what the summary should cover [61]. The former is also called pixel-based or direct ground truth, and the latter is called text-based or indirect ground truth. The term is indicated as *type* in the taxonomy.

The automatic comparison involving ground truth can include different components associated with the type of the ground-truth annotations; termed *evaluative components*. The pixel-based ground truth comparison is carried out by evaluating the degree of match between the ground truth summary (reference summary), and the summary of interest (candidate summary). The term is indicated as *visual components*. This process goes through several steps, and relies on choices made at each step. The text-based ground truth comparison measures the overlapping words between the reference and the candidate summaries, using a *lexical metric*. Details are given in subsequent sections.

In this thesis we are interested in proposing a unified protocol for automatic comparison of two video summaries. To this end, we identify the necessary components (stages), and carry out a comprehensive study to establish which of the many available algorithms or approaches fits each component to the highest degree.

3.2 A Review on Automatic Evaluation Frameworks

While there is a multitude of works on video summarisation, surprisingly little has been done toward developing a comprehensive objective evaluation protocol. The need for such a protocol is widely acknowledged [158, 53, 49, 84,

135, 96, 115]. Existence of a standard and consistent evaluation framework is essential in order to determine to what extent the candidate summary matches the ground truth summary. Such a framework will also allow for a fair comparison among the increasing amount of novel summarisation methods.

Evaluation of video summaries is difficult partly due to the following reasons [158]: 1) Unlike other research areas such as object recognition, the ground truth is not consistent across human evaluators. Studies show that an ‘ideal’ summary does not exist [60]. Therefore, evaluation of video summary is not a straightforward task; 2) Summarisation perspective is application-dependent. For this reason, to compare two summarisation techniques, their application aspect resemblances must be taken into account; 3) Previous works are often unavailable for comparative evaluations, or require a certain setting or format for use; 4) Evaluating a large number of existing summarisation techniques on a commonly accessible, large, and diverse video data set with long hours is important but very difficult to accomplish.

Our study is devoted to visual-based keyframe summarisation. Table 3.1 summarises some literature sources which propose new evaluation frameworks. A standard evaluation protocol could be based on the following components:

(1) A feature representation. The first step is representing the reference summary, and the candidate summary as a collection of vectors in some feature space. Ideally, this should be done using a simple and effective feature extraction algorithm. Feature representations can be based on colour, texture, or motion (displacement). Thus each frame is represented as a point in some metric space \mathbb{R}^n .

Proposing an automatic evaluation framework, authors typically presented frames using a colour histogram, calculated in the Hue-Saturation-Value (HSV) colour space but quantised into different number of bins. Avila et al. [40] only employed the hue component with 16 colour bin quantisation,

while Mahmoud et al. [107] and Kannappan et al. [81] both used all three components, respectively quantised into 32, 4, 2 and 32, 4, 4 bins of hue, saturation and value. Mahmoud et al. [107] applied texture features (Discrete Haar Wavelet Transforms) once after validating the colour similarity of two frames. Jinda et al. [77] employed colour histogram of the hue channel after computing the number of the matching points, and number of matching errors using Speeded Up Robust Features (SURF) features. Khosla et al. [84] computed the scale invariant feature transform flow [99] features between two frames selected from the automatic set and the ground truth set.

Table 3.1: Overview of existing automatic evaluation frameworks classified based on the taxonomy components.

| | Feature representation | | | Similarity metric | | | | Matching strategy | | | Accuracy metric | | Data set | | | |
|----------------------------|------------------------|---------|--------|-------------------|------------|------------|---------------------|-------------------|-----------|----------------------|-----------------|---------------------|-------------------|------------------------------|-----------|---------------|
| | colour histogram | texture | motion | Manhattan | Bhattaryya | Pixel-wise | Pearson Correlation | Temporal greedy | Hungarian | Bidirectional search | Ratio/Threshold | Accuracy/Error rate | Average precision | Precision/ <i>F</i> -measure | Benchmark | Non benchmark |
| Avila et al.,2011 [40] | ✓ | | | ✓ | | | | ✓ | | | ✓ | | | | ✓ | |
| Jinda et al., 2013 [77] | ✓ | ✓ | | | | ✓ | | | | | ✓ | | | ✓ | | ✓ |
| Khosla et al.,2013 [84] | | | ✓ | | | ✓ | | | ✓ | | | | ✓ | | | ✓ |
| Mahmoud et al., 2013 [107] | ✓ | ✓ | | | ✓ | | | ✓ | | | | | | ✓ | ✓ | |
| Kannappan et al.,2016 [81] | ✓ | | | | | | ✓ | | | ✓ | | | | ✓ | ✓ | |

(2) *A similarity metric.* This metric evaluates the degree of match between two given frames. If the frames are represented in \mathbb{R}^n , the similarity can be calculated by a linear correlation metric, or a distance metric.

In the next step, distance metrics including Manhattan², Bhattacharyya³, and sum-squared pixel-wise distances are used respectively in [40], [107], and [84] to inspect similarity of two frames. Jinda et al. [77] also calculated the intersections between the two histogram sets, which is categorised as pixel-wise similarity metrics, after matching a certain number of keypoints. Seen against the distance metrics, Kannappan et al. [81] employed correlation metric, e.g. Pearson Correlation Coefficient to measure the similarity.

²Manhattan distance between two points calculates the sum of the absolute differences of their Cartesian coordinates.

³In statistics, the Bhattacharyya distance measures the similarity of two probability distributions.

(3) *A matching strategy.* The matching strategy is used to pair the frames between the two summaries. A temporally-ordered greedy matching algorithm is used in [40, 107]. Once a match is detected, the matched cases (frames) are removed from both sets. A bidirectional search, and the Hungarian matching algorithms are employed by Kannappan, Liu and Tiddeman [81] and by Khosla, Hamid, Lin and Sundaresan [84], respectively.

(4) *An accuracy metric.* This metric is applied to calculate the overall similarity of the two summary sets based on the individual matching scores between the paired frames.

Typically, the quality of the candidate summary is evaluated by the F -measure (F -value or F -score), calculated from the number of matched frames and the cardinalities of the two summaries [107, 49, 60, 56, 61, 174, 112, 77, 81]. Authors also made use of Average Precision (AP) metric [84, 175], by averaging the precision scores of a ranked-ordered summary. Average precision score in [84] is obtained by finding the area under the precision-recall curve, where the curve is plotted by iterative evaluation of precision and recall values in ascending orders of number of matches. To do so, the summary obtained by an annotator is arranged from high to low rank (score), and the number of matches with the automatic summary set is detected. Subsequently, precision and recall values for each step⁴, are calculated by summing the matched (pairwise) distances, and plotted to create the precision-recall curve.

Another measure of quality is the Comparison of User Summary (CUS). It consists of Accuracy rate (CUS_A) and Error rates (CUS_E). They are calculated as the ratio of the number of matched and non-matched frames with the ground truth summary [40]. Observing the discrepancy between user summaries for the same video stream, Li et al. [94] adapted evaluation metrics from information retrieval to be used as a unified metric in evaluating

⁴Step values start from one for number of matches to the total number of frames in the ground truth

video summaries, termed *VERT* (Video Evaluation by Relevant Threshold) in the diagram (Figure 3.1).

(5) *A data set.* It is important to have access to a data set with its ground truth allowing researchers to compare their proposed methods against previous ones.

Majority of the automatic evaluation studies [40, 107, 81] conducted their experiments on a benchmark set, typically VSUMM (Video SUMMarization) video collection⁵. The video collection offers two data bases of videos acquired from different sources: 1) a video set containing 50 videos in MPEG-1 format assembled from the open video project⁶. Video content include several genres: educational, ephemeral, documentary, historical, and lectures; 2) a video set collected from YouTube containing 50 videos of several genres including cartoons, news, sports, commercials, tv-shows, and home videos.

Khosla et al. [84] obtained their video collection by searching the YouTube for certain title descriptions, while Jinda et al. [77] recorded their own lifelog data set. Both are referred to as non benchmark in the Table 3.1, indicating that the data set is not available.

If the ground truth is provided in the form of text (the *lexical* approach), the evaluation protocol should follow a different pattern. According to the text-based evaluation approach [177], the original video is annotated based on its semantic content. To achieve that, the video is uniformly segmented into short subshots and one sentence description of each subshot is manually obtained. A ground truth text summary is also generated by asking a human annotator to write a text summary of what happened in that video. Once the candidate summary is generated using a summarisation algorithm, the associated text annotations of frames are extracted, then concatenated, and compared against the ground truth text summary. This can be done by applying the ROUGE metric. ROUGE, Recall-Oriented Understudy for Gisting

⁵<https://sites.google.com/site/vsummsite/home>

⁶<https://open-video.org/>

Evaluation, is a standard text summary evaluation algorithm to automatically determine the quality of a summary by comparing it to another set of ‘ideal’ summaries typically created by humans. The measure counts the number of overlapping words or n-grams between the computer-generated summary and the ground truth summary [97].

3.3 Components of the Evaluation Protocol

Our aim is to propose a protocol for visual-based evaluation of a candidate keyframe summary with respect to a ground truth summary. This section details our choice of methods and algorithms for every component of the protocol. We list many alternatives which we will subsequently use in an empirical study in order to select the best combination.

3.3.1 Feature Representation

Here, we detail feature descriptors with different properties to investigate their representation influence on judging the similarity between two sets of keyframes. In the comparison, we include colour based representations described in the Red-Green-Blue (RGB), HSV, and other standard, though less popular, colour spaces: Chrominance components (CHR) and Ohta (OHT). Texture based descriptors and CNN features are also considered.

For the colour-based descriptors, the original image is decomposed into two channels (CHR) or three channels (RGB, HSV, OHT). Thereafter, each image channel is uniformly divided into $n \times n$ blocks, termed the sub-images. The feature descriptors considered in our analyses are described bellow:

1. *RGB_9blocks* : The image is represented in the RGB colour space, and each image channel split into a 3-by-3 grid. The mean and the standard deviation of each channel for each sub-image are calculated, and the values are concatenated to generate a feature vector of 54 dimensions.

2. *CHR_9blocks* : The chrominance components for a given pixel are calculated as [164]:

$$C_1 = \frac{R}{q}, \quad C_2 = \frac{G}{q}, \quad q = \sqrt{R^2 + G^2 + B^2},$$

where R , G , and B are the red, green and blue intensities of the pixel, respectively.

The mean and standard deviation of the components C_1 and C_2 are computed, and the values are concatenated, producing a 36 dimensional feature vector.

3. *OHT_9blocks* : Image is transformed into Ohta space as follows [123]:

$$\begin{aligned} I1 &= \frac{1}{3}(R + G + B) \\ I2' &= R - B \\ I3' &= \frac{1}{2}(2G - R - B) \end{aligned}$$

The mean and standard deviation of each channel are calculated. A feature vector of 54 dimensions is created concatenating the computed values.

4. *HSV_9blocks* : The image is converted into HSV colour space, and the mean and standard deviation of each sub-image for all colour channels are computed. Concatenating the values, a feature vector of 54 dimensions is obtained.
5. *H8_9blocks* : Using the HSV colour space, only the values obtained from the hue (H) channel are used. For each sub-image of the 3-by-3 grid, the values are quantised into 8 bins producing the H-histogram. The feature vector obtained from this descriptor has 72 dimensions.
6. *H16_9blocks* : A histogram of only the hue channel is computed. For each sub-image of the 3-by-3 grid, the values are quantised into 16 bins producing a 144 dimensional feature vector.
7. *H16_4blocks* : H-histogram for sub-images of a 2-by-2 grid is computed, quantising the hue channel values into 16 bins. This generates a feature vector with 64 dimensions.

8. *H16_1blocks* : H-histogram is calculated by quantising hue values into 16 bins, without splitting the image into grids. A feature vector of 16 dimensions is obtained.
9. *H32_1block* : H-histogram with 32 bins is computed, producing a 32 dimensional feature vector.

Beyond colour based features, the following feature descriptors were added to the list:

10. *SURF* : These features are used to match relevant points between two frames.
11. *CNN* : The last fully connected layer of a pre-trained CNN was used as a 4096-dimensional feature space [147].

3.3.2 Similarity Metrics

There are many ways to calculate similarity of two frames. A popular approach is calculating pairwise distance between the point representations of the two frames in the n -dimensional feature space. Point-wise distance will be our approach for the feature descriptors specified above, excluding the SURF features. Here we used the Manhattan and the Euclidean distances.

For the Manhattan distance, we transform both frame representations into probability distributions so that the values for each frame sum up to 1. For this choice, the Manhattan distance varies between 0 (identical distributions) and 2 (completely non-intersecting distributions). This gives us the ground for selecting the span of possible threshold values for our experiment.

For the Euclidean distance, we took a different approach. This time we did not normalise the data into two distributions but used the original features. This is why the thresholds for the Manhattan distance and the Euclidean distance in our experiments have different scales.

Another way to calculate the similarity between two frames is keypoint matching which evaluates the proportion of matched local features such as SIFT [100], or SURF [77] keypoints.

When employing SURF descriptors, we implement keypoint matching approach as an alternative example that does not attempt to embed the frames in \mathbb{R}^n .

The SURF features are applied to find similar points (matching keypoints), therefore number of matches are computed by counting the number of those keypoints. Different number of keypoints will be identified for each pair of compared frames. For comparing frames a and b , the following procedure is applied:

1. Identify the total number of keypoints in frame a (denoted as n_a), and assign the number of keypoint matches with frame b to k_a ;
2. Identify the total number of keypoints in frame b (n_b) and assign the number of matched keypoints with frame a to k_b ;
3. Calculate the similarity between the two frames as the following proportion: $\frac{2 \min\{k_a, k_b\}}{n_a + n_b}$. For consistency with other distance metrics we introduced a distance between two frames a and b as:

$$d(a, b) = 1 - \frac{2 \min\{k_a, k_b\}}{n_a + n_b}. \quad (3.1)$$

3.3.3 Matching Strategies

In order to evaluate a candidate summary, the number of matches with a ground truth summary is determined while accounting for the total number of frames in both summaries [40, 84, 49, 107, 56, 112].

It is assumed that a distance metric d between two frames has been already chosen, as discussed in Section 3.3.2. Two frames a and b are sufficiently similar to be called a match if $d(a, b) < \theta$, where θ is a chosen threshold.

Let S and GT represent two sets of keyframes, one obtained by an automatic summarisation approach (the candidate summary) while the other acquired from a user’s annotation (the ground truth). We are interested in measuring the degree of similarity between the two sets. The objective is to find a suitable process of pairing frames, one from each set, so that the number of matches m between S and GT is accurately calculated.

The cardinalities of the two summary sets are denoted as $N_1 = |S|$ and $N_2 = |GT|$, where S is the computer-generated set of frames (the automatic summary or candidate summary), and GT is the user selection of frames (ground truth). A distance matrix D is constructed where element $d_{i,j}$ is calculated as a distance value between frames $i \in S$ and $j \in GT$, generating the total dimension of $(N_1 \times N_2)$. This matrix will be referred to as a ‘pairwise distance matrix’ in this chapter. The number of matches returned by a matching algorithm is denoted by m . Six matching algorithms are examined, as detailed below.

(1) *Naïve Matching* (no elimination). The inspection for matching goes through each item (keyframe) in the computer-generated summary, and looks for a match in the ground truth summary. If a match is found, the match counter is incremented, and the next element of the computer-generated summary is processed. No frame is removed from the ground truth summary. Not eliminating matched frames from the ground truth summary causes an obvious problem. If the computer-generated summary set S consists of almost identical frames which happen to be close to one frame from the ground truth summary GT , then all frames in the computer-generated summary will be matched with this one frame. Despite generating a perfect matched number of $m = N_1$, for an arbitrary N_1 , such a candidate summary is quite inadequate. It is neither concise nor representative. Algorithm 1 relies on the presumption that S is a reasonable summary containing diverse frames.

(2) *Greedy Matching*. This algorithm is widely used to match keyframe summaries despite the fact that it is quite conservative. Initially, the pairwise distance matrix D is calculated. As long as the minimum distance is below the

Algorithm 1: Naïve Matching.

```
1  $m \leftarrow 0$ .
2 for  $i = 1, \dots, N_1$  do
3   if any  $d_{i,j} < \theta$ ,  $j = 1, \dots, N_2$  then
4      $\quad$  increment the number of matches,  $m \leftarrow m + 1$ .
```

threshold value, the corresponding frames from the two sets are identified as a matching pair. Once the match is detected, the items are removed from both sets, and the iteration passes to the next minimum distance value, as detailed in Algorithm 2.

Algorithm 2: Greedy Matching.

```
1  $m \leftarrow 0$ .
2 Find the smallest distance  $d_{\min} = \min D$ .
3 while  $d_{\min} < \theta$  do
4   Increment the number of matches,  $m \leftarrow m + 1$ .
5   Remove the row and the column of the matched elements from  $D$ .
6   Find the smallest distance from the remaining matrix  $d_{\min} = \min D$ .
```

(3) *Hungarian Matching*. The Hungarian algorithm is a common bipartite graph matching algorithm, used by Khosla et al. [84] (Algorithm 3). After calculating the pairwise distance matrix D , the algorithm identifies the pairs such that the sum of the distances of the paired frames is minimum. A thresholded matching can be naïvely formed from this minimal complete matching by simply removing all pairings over the threshold distance θ . Thus, close matches could be missed in an attempt to minimise the total distance.

Algorithm 3: Hungarian Matching.

```
1 Apply the Hungarian assignment algorithm to  $D$ .
2 Identify the matched pairs of frames  $(i, j)$ , and retrieve the distances  $d_{i,j}$  from  $D$ .
3 Assign to  $m$  the number of these distances which are smaller than  $\theta$ .
```

(4) *Temporally-ordered Greedy algorithm*. This algorithm is used in [40, 107, 105]. The frames in both summaries are arranged in a temporal order. For each frame in the ground truth set, the nearest match from the computer-generated set is detected, and eliminated accordingly from both sets (Algorithm 4). Apart from the temporal ordering, the algorithm is identical to the Greedy Matching.

Algorithm 4: Temporally-ordered Greedy Algorithm.

Input: Keyframe summaries S and GT arranged in temporal order, and threshold θ .

Output: Number of matches m .

```
1  $m \leftarrow 0$ .
2 for  $j \in GT$  do
3   for  $i \in S$  do
4     if  $d_{i,j} < \theta$  then
5       Increment the number of pairings,  $m \leftarrow m + 1$ .
6       Remove  $i$  from  $S$  and  $j$  from  $GT$ .
7       Break.
```

(5) *Bidirectional Search Algorithm*. An interesting alternative approach to the matching problem is put forward by Kannappan et al. [81]. In their approach, a keyframe from the candidate set and a keyframe from the ground truth are matched only if each is the other's best possible match (Algorithm 5). We have modified this procedure to make the thresholding equivalent to that of the temporally-ordered greedy algorithm.

Algorithm 5: Bidirectional Search Algorithm.

```
1 Initialise a set of pairings  $M \leftarrow \emptyset$ .
2 for each frame  $i \in S$  do
3   for each frame  $j \in GT$  do
4     if  $j' = \arg \min_{s \in GT} d(i, s)$  and  $i' = \arg \min_{s \in S} d(s, j)$  then
5       Add the pair to the matching set  $M \leftarrow M \cup \{(i', j')\}$ .
6 Remove  $M$  from all pairs for which  $d(i', j') \geq \theta$ .
7  $m \leftarrow |M|$ .
```

(6) *Maximal Matching*. The greatest possible value of m is given by a maximal *unweighted* matching in which frames less than distance θ apart can be paired. Such a matching is given by the Hopcroft-Karp algorithm [166]. We will use instead the convenient alternative Algorithm 6, in which we find the lowest-weight complete matching on a binary matrix D' obtained by thresholding D . Entry $d'_{i,j}$ in D' has value 0 if $d_{i,j} < \theta$, and 1 otherwise. After the optimal assignment is found through the Hungarian algorithm, the number of matches is determined by counting how many of the matched pairs are at distance less than θ .

Algorithm 6: Maximal Matching algorithm.

- 1 Construct matrix D' of the same size as D such that $d'_{i,j} = 0$ if $d_{i,j} < \theta$, and $d'_{i,j} = 1$, otherwise.
 - 2 Apply the Hungarian assignment algorithm to D' .
 - 3 Identify the matched pairs of frames (i, j) , and retrieve the distances $d_{i,j}$ from D .
 - 4 Assign to m the number of these distances which are smaller than θ .
-

To illustrate the different behaviours of these algorithms, a small bipartite graph is presented in Figure 3.2. This example shows that even when using the same distance metric and threshold value, algorithms may return different matching items and/or different cardinalities. This case also demonstrates the importance of selecting an appropriate matching algorithm.

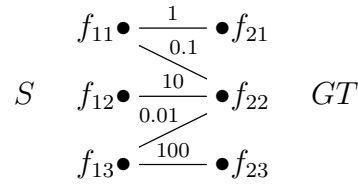


Figure 3.2: Example of a small bipartite graph to illustrate behaviour of matching algorithms. Each of the algorithms will return a different matching for this graph. The numbers in the figure give the weights of the five edges.

Assume S is the computer-generated summary holding $\{f_{11}, f_{12}, f_{13}\}$, while GT is the ground truth summary containing $\{f_{21}, f_{22}, f_{23}\}$, where f indicates the frame. Degree of similarity between two frames are given as a weight number of the edges at the bipartite graph. We assume all weights are below the threshold.

Starting with the *Naïve Matching* algorithm, the algorithm first selects the edge with weight 1 as a match. Next it moves to the second node of f_{12} without any elimination, where it finds edge with weight 10 as its second match. It continues with the third vertex of f_{13} where the edge of weight 0.01 is selected as a third match. In overall, a match set of cardinality 3 is identified using the Naïve algorithm.

Considering the *Greedy* algorithm, first the edge with the lowest weight, 0.01, is detected as a match. The pair of frames (f_{13}, f_{22}) is eliminated from the further similarity detection process. The algorithm continues with the second

lowest weight of 1 and the only remaining possible pair. At this point, no more weights are available, therefore the matching algorithm stops and a matching of cardinality 2 is returned.

The *Hungarian Matching* algorithm will return the only possible one-to-one matching of cardinality 3 with weights 1, 10 and 100.

The output of the *Temporally-ordered Greedy* algorithm depends on the order of vertices submitted to the algorithm. Therefore, the edge with weight 1 is detected first as a match, followed by the edges with weight 10 and 100. The algorithm finds a matching of cardinality 3.

Moving to the *Bidirectional Search* algorithm, a matching of cardinality 1 is found consisting of frames (f_{12}, f_{22}) joined by an edge with weight 0.01.

Finally, a *Maximal Matching* algorithm returns a maximal unweighted matching of 3 given by the set of edges of weights 1, 10, and 100.

Note that other matching algorithms such as dynamic time wrapping [139] used in speech recognition can be added into the experiment. However, this algorithm has the potential to find many matches for a single item in the ground truth set. Therefore, it is quite naive to use such an algorithm in detecting matches between two sets.

Beside the impact of choosing any of the above-mentioned algorithms for detecting an accurate number of matches, the selected threshold value (parameter) also has an immediate effect on identifying similarity of the pairs. The threshold may vary based on different choices of feature descriptor and similarity metric. So far, this value has been either intuitively selected [81] or empirically found [40, 107]. However, to the best of our knowledge, there is no study which examines the effect of choosing among various values of this parameter combined with different selection of features and similarity metrics on estimating the summarisation accuracy.

For our experiments, we will use a range of thresholds from 0.01 up to 0.7 for the Manhattan metric. For the Euclidean metric, we will scale the threshold relative to the distribution of all pairwise distances between frames in the video. The thresholds will be percentiles of this distribution, from the 0.01th up to the 3rd percentile. For the SURF metric, we will vary the threshold between 0.01 and 0.4.

3.3.4 Accuracy Metrics

Once the number of matches m has been found, the value is used to quantify the effectiveness of the computer-generated summary against the ground truth summary. We introduce γ to measure how close two sets of keyframes are together.

Avila et al. [40] propose a pair of measures called Accuracy/Error rate (CUS_A/CUS_E). Having S , GT and m respectively for the computer-generated summary, the ground truth summary, and the number of matches, the two metrics are defined as:

$$CUS_A = \frac{m}{|GT|}, \text{ and}$$

$$CUS_E = \frac{|S| - m}{|GT|}.$$

The values of CUS_A range from 0 (when none of the frames from the computer-generated set matches any frame in the ground truth set) to 1 (when all frames in the computer-generated set match with the ground truth summary). Note that value 1 of CUS_A does not indicate one-to-one correspondence between the frames of S_1 and S_2 because the ground truth summary can be bigger than the computer-generated summary. On the other hand, the values of CUS_E vary from 0 (when all frames of the computer-generated set match the ground truth) to N_1/N_2 (when no frame of the computer-generated summary matches the ground truth set). The problem with these measures is that the upper limit of CUS_E depends on $|S|$.

Alternatively, making use of precision and recall is very popular as shown in the Table 3.1. Given a number of matches m , the similarity between S and GT can be quantified using the F -measure, whose advantage is that it is symmetric on its two arguments [56]. Without loss of generality, choose S for calculating the Precision, and GT for calculating the Recall. Then

$$\begin{aligned}
\text{Recall} &= \frac{m}{|GT|} \\
\text{Precision} &= \frac{m}{|S|} \\
F(S, GT) &= \frac{2(\text{Recall} \times \text{Precision})}{\text{Recall} + \text{Precision}} \\
&= \frac{2m}{|S| + |GT|}
\end{aligned} \tag{3.2}$$

We have chosen to use this F -measure as our $\gamma(S, GT)$ because, unlike CUS_A and CUS_E , it is symmetric, limited between 0 and 1, and interpretable.

We note that there is a potential problem when using the F -measure with the *Naïve Matching* algorithm and the *bidirectional search* algorithm because they do not guard against $m > N_2$, which may lead to $F > 1$. In such cases we clipped the value of F to 1.

3.4 Evaluation Protocol

3.4.1 What is a Good Evaluation Protocol?

The fundamental idea for our experiments is that a good measure of similarity between two summary sets should distinguish as clearly as possible between content-blind baseline methods, such as uniform summaries, on the one hand, and a sophisticated algorithmic summary, on the other hand.

Denote by Θ the set of possible choices of feature descriptor, similarity metric, matching algorithm, and threshold value. As a baseline model we consider a summarisation method termed *Uniform* summarisation. This method requires a single parameter: the desired number of frames K . The video is split into K equal time-contingent segments, and the middle frame of each segment is

taken in order to make the summary. The best choice of Θ will ensure that the difference between computer-generated summaries and uniform summaries is the largest possible.

3.4.2 Data Set

Operating on the same data set as the state-of-the-art methods, permits to assess our evaluation protocol on their results, along with the ground truth sets. For this experiment we used the VSUMM video collection including 50 videos recorded in 30 fps in 352×240 pixels. The videos duration varies from 1 to 4 minutes which provide the approximate of 75 minutes in total. Each video has been manually summarised by 5 different users providing the total of 250 user summaries.

The five algorithmic summarisation methods are provided in the VSUMM video database are:

- Delaunay Triangulation (DT) [119],
- Open Video Project (OV)⁷,
- STill and MOving Video Storyboard (STIMO) [53],
- Video SUMMarization1 (VSUMM₁) [40], and
- Video SUMMarization2 (VSUMM₂) [40].

3.4.3 Discrimination Capacity

To estimate how well a measure distinguishes between baseline designs and bespoke selection methods, we propose the quantity which we call *discrimination capacity* as the difference:

$$c_U \triangleq c_U(S, U, GT) = \gamma(S, GT) - \gamma(U, GT), \quad (3.3)$$

where GT is a ground truth summary set, S indicates a computer-generated summary set obtained by an algorithmic method, and U is a baseline summary, which in our case will be the Uniform summary of the same cardinality as S . From now on, the accuracy metric of γ is being measured by

⁷<https://www.open-video.org>.

the F -measure (F -score). Large values of c_U will signify good choices of the properties Θ , which could be recommended for the practical implementation of the proposed protocol as a tool for evaluation of future video summarisation algorithms.

We stipulated that the evaluation protocol must be independent of the video summarisation method. Thus, the computer-generated summary can be obtained using any automatic summarisation approach. For instance, video can be either divided into shots/events or used in its entirety; representation of frames can be based on low-level colour, or complex visual features; frame selection can be supervised or unsupervised.

We also assume that the frames in both sets of computer-generated summary and the ground truth are not ranked by degree of importance, nor are they arranged in a temporal order.

Despite some automatic evaluation approaches that fuse multiple annotated summaries into a single ground truth summary [70, 56], to maintain simplicity and transferability we decided to use each ground truth separately.

Let $GT = \{S_{gt1}, \dots, S_{gtL}\}$ be a set of ground truth summaries obtained from L users. Let $U(s)$ be a uniform summary with s number of keyframes. We calculate C_U , the average of c_U for S and GT , as:

$$\begin{aligned} C_U &= \frac{1}{L} \sum_{i=1}^L c_U(S, U, S_{gti}) \\ &= \frac{1}{L} \sum_{i=1}^L (F(S, S_{gti}) - F(U(|S|), S_{gti})) , \end{aligned} \quad (3.4)$$

where F is the F -measure. C_U measures how well S performs compared to a uniform keyframe summary with the same cardinality. To make a fair comparison, we set the cardinality of U as $|U| = |S|$. As the value for C_U depends on the choices of the properties Θ , we look for a set which maximises

the discrimination capacity C_U across a range of videos and summarisation algorithms.

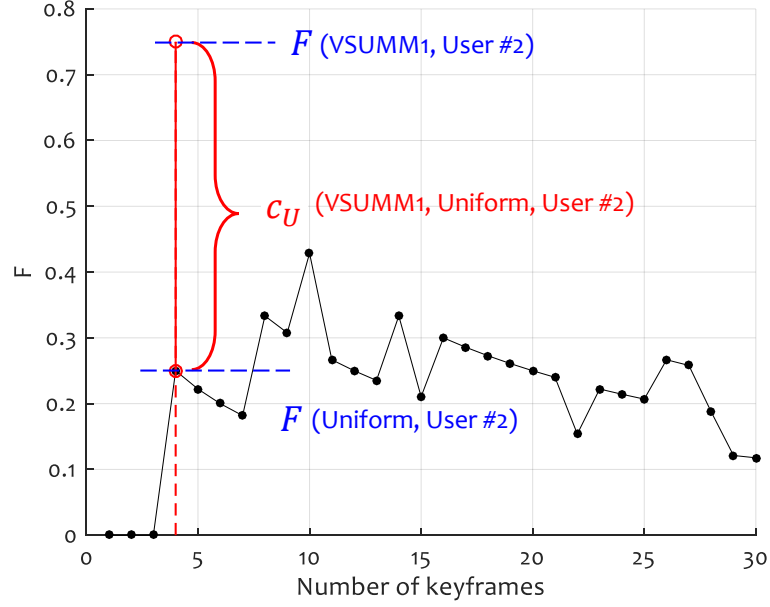


Figure 3.3: An example of calculating C_U for the summary generated by VSUMM₁ method for Video #22, feature space #8 (H16_1block), Manhattan distance, the Hungarian Matching method, and threshold 0.5. c_U is the difference between the F -value for matching candidate summary VSUMM₁ to User #2 (ground truth #2) and the F -value matching a uniform summary of the same cardinality as VSUMM₁ (4 in this case) and User #2. C_U is the average of the 5 such c_U terms in Eq.3.4.

The calculation of C_U in Eq.3.4 is graphically illustrated in Figure 3.3 for the automatic summary obtained from the VSUMM₁ approach, and the uniform summary of the same cardinality. Both summaries are compared with just one of the user-generated summaries. The value of C_U is a measure of how much closer the automatic summary is to a ground truth summary compared with a uniform summary of the same size.

The full calculation of C_U value for the remaining four user annotations is shown in Table 3.2. Each entry represents the F -measure value between a user ground truth, and either the automatic or the uniform summary. The overall value C_U is obtained by averaging the difference between the F -values across the same user.

Note that the advantage of setting the same cardinality of U and S is to avoid having a bias toward either of these competitors. Taking into the account that

in Figure 3.3, graph $F(\text{Uniform}, \text{User}\#2)$ shows an irregular behaviour when cardinality values are increased, for a fair comparison it is the best to keep the cardinality of U the same as S . Doing so, $F(\text{VSUMM}_1, \text{GT})$ and $F(\text{Uniform}_4, \text{GT})$ in Table 3.2 have the same cardinality (number of keyframes), and their values are calculated based on the number of matches found in each set with the ground truth set.

Table 3.2: An example of the calculation of C_U for Video #22, selected from the VSUMM data set. The automatic summary method is the VSUMM₁ keyframe selection method. Other properties are: feature space #8 (H16_1block), the Hungarian Matching method, Manhattan distance, and threshold 0.5. The F -values are shown in the table; the bottom row contains the terms in Eq.3.4; the values for User #2, marked with * are the ones in Figure 3.3.

| | user 1 | User 2* | User 3 | User 4 | User 5 | overall |
|----------------------------------|--------|---------|--------|--------|--------|---------|
| $F(\text{VSUMM}_1, \text{GT})$ | 0.5000 | 0.7500* | 0.6667 | 0.2857 | 0.4444 | |
| $F(\text{Uniform}_4, \text{GT})$ | 0.5000 | 0.2500* | 0.2222 | 0.2857 | 0.4444 | |
| C_U terms | 0 | 0.5000* | 0.4444 | 0 | 0 | 0.1889 |

3.4.4 Identifying the Protocol Components

Description of the Experiment

The purpose of the experiment is to determine which combination of values of Θ maximises C_U . The results may serve as a unified evaluation methodology for comparing a candidate summary with a ground truth summary. We considered: 11 feature spaces, 6 matching algorithms, 2 types of distance (Euclidean and Manhattan) for the metric spaces, a proportion-based distance for the SURF features, and a range of values of the threshold θ for each distance.

For the uniform baseline, for each video we generated 30 summaries with cardinalities from 1 to 30.

An overall value of C_U for each instance of Θ is calculated as an average across the values for all users videos and summarisation methods. In this experiment we will be looking for the best possible combination of properties Θ to maximise C_U .

In our experiments we calculated C_U for every choice of property settings and every video.

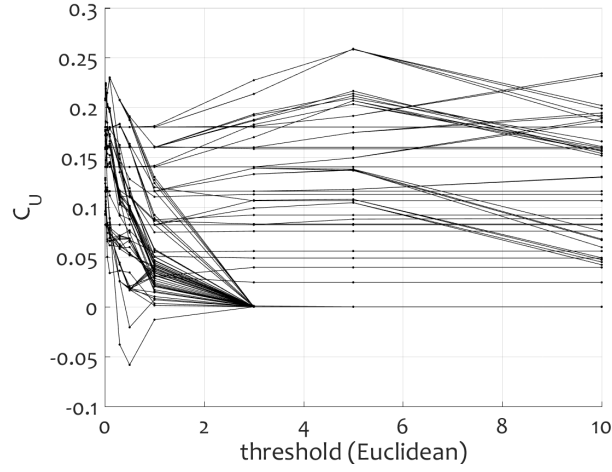
Evaluation of Distance Metrics and Thresholds

The aim of this evaluation is to choose a suitable distance metric along with its threshold value. The following threshold ranges were evaluated:

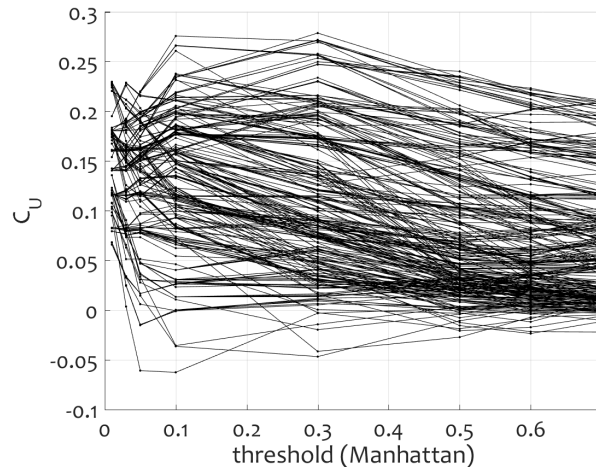
1. For the Euclidean distance: $\theta \in \{0.01, 0.02, 0.05, 0.1, 0.3, 0.5, 1, 3, 5, 10\}$,
2. For the Manhattan distance: $\theta \in \{0.01, 0.03, 0.05, 0.1, 0.3, 0.5, 0.6, 0.7\}$,
3. For SURF feature distance: $\theta \in \{0.01, 0.03, 0.05, \dots, 0.39\}$.

In Figure 3.4, we plot C_U versus the threshold θ for the 10 feature descriptors (SURF features is plotted separately), 6 matching strategies, and 5 automatic summary methods. Therefore, each line in the (θ, C_U) graphs corresponds to a specific combination of a feature descriptor, matching algorithm, and summarisation method. Note that C_U may be negative. This is the undesirable case where the uniform summary matches the user ground truth better than the algorithmic (candidate) summary.

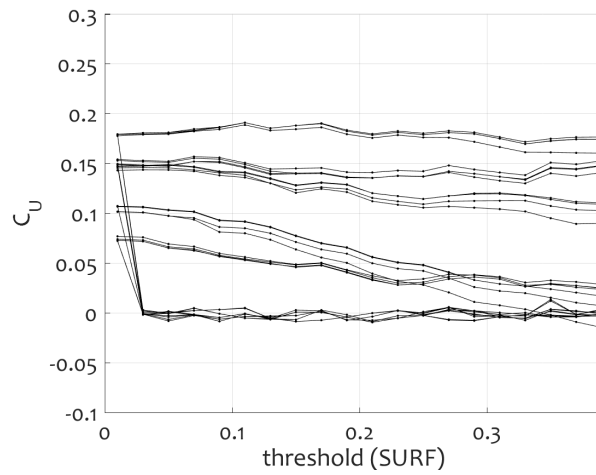
The shape of the line graph in relation to the threshold is expected to be convex with lower values for smaller and larger thresholds. For small thresholds, there will be very few matches, hence the F -values will be low for both the candidate summary and the uniform summary, hence the difference C_U will be small. For large values of the threshold, a large number of matches will be detected in both comparisons, both F -values will be high, and the difference C_U will be small again. The best results (larger C_U) are offered by the Manhattan distance. The peak for the Manhattan distance is between $\theta = 0.3$ and $\theta = 0.5$. For the Euclidean distance, there are two different types of curves. Some peak quite early, at θ between 0 and 0.5, while others stay stable. The SURF feature curves exhibit consistent and stable patterns, which will be analysed later.



(a) Euclidean distance



(b) Manhattan distance



(c) SURF feature distance

Figure 3.4: Discrimination capacity C_U as a function of the threshold for the three types of distances used. Each of plots (a) and (b) contains 300 line graphs (10 feature spaces, 6 matching methods, 5 summarisation methods). Plot (c) contains 30 lines (SURF space, 6 matching methods, 5 summarisation methods). Each line is the average across 50 videos and 5 users.

From these findings, we favour the Manhattan distance for our proposed protocol, and will use this distance for the following evaluation of the feature spaces.

Evaluation of Feature Representation

Considering the best results obtained by the Manhattan distance in the previous section, now we look for a feature representation which maximises C_U . Figure 3.5 shows the results for the 11 feature descriptors. Each sub-plot corresponds to one feature space. As in Figure 3.4 (b), the horizontal axis is the threshold used with the Manhattan distance, and the vertical axis is C_U . This time, all curves corresponding to the 10 feature descriptors are shown in each plot at the same time, which makes 30 curves obtained from the combination of 6 matching algorithms with 5 summarisation methods. In each graph, the respective feature descriptors are highlighted in black.

Our results show that the simple colour based features (1-4) are not useful in this context. The hue histograms, on the other hand, give the best results. The feature descriptor with the largest C_U is *H32_1block*. It is somewhat surprising to find that a colour based descriptor wins over the texture (SURF) or CNN feature descriptors. This result hints to the possibility that spending a lot of computational effort for calculating highly sophisticated properties of images may be unjustified in some cases. Thus, we propose to use *H32_1block* for the purposes of automatic evaluation of keyframe summaries when ground truth is available.

Evaluation of Matching Algorithms

The results for this part are shown in Figure 3.6. The lines plotted in black are the ones corresponding to the matching method in the title of the subplot.

It can be seen that, for Euclidean and Manhattan distance, the Naive matching is slightly inferior to the rest of the matching methods. This is to be expected, as the Naive labelling method may result in a large number of false positive matches for both the uniform summary and the summary of interest.

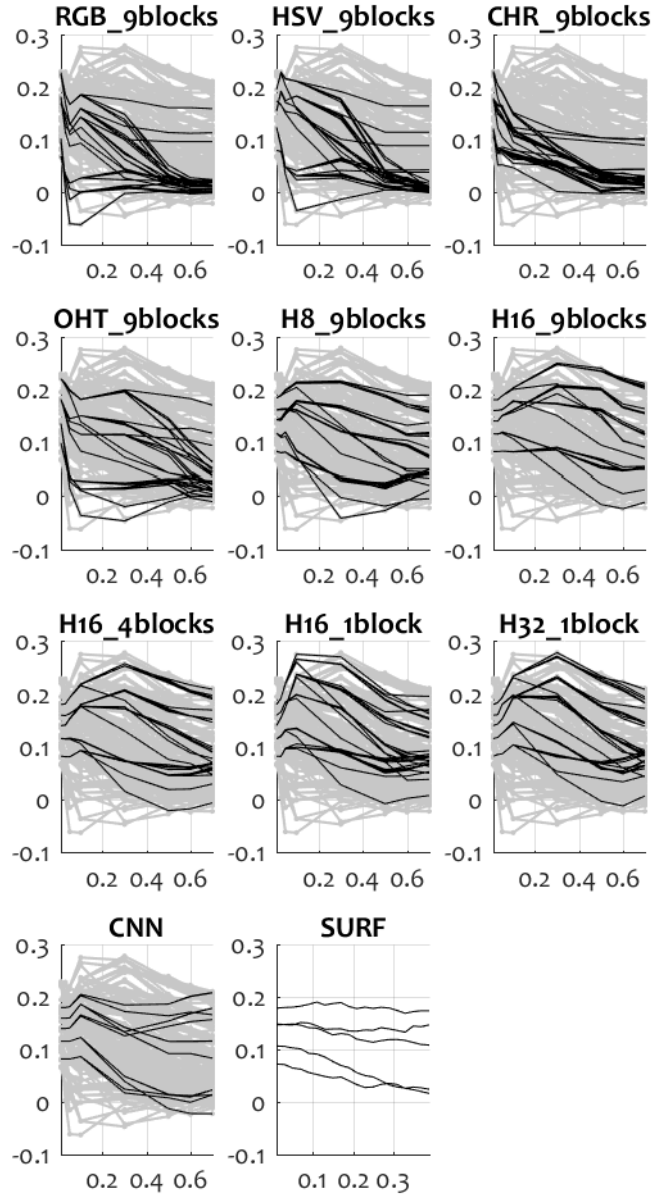
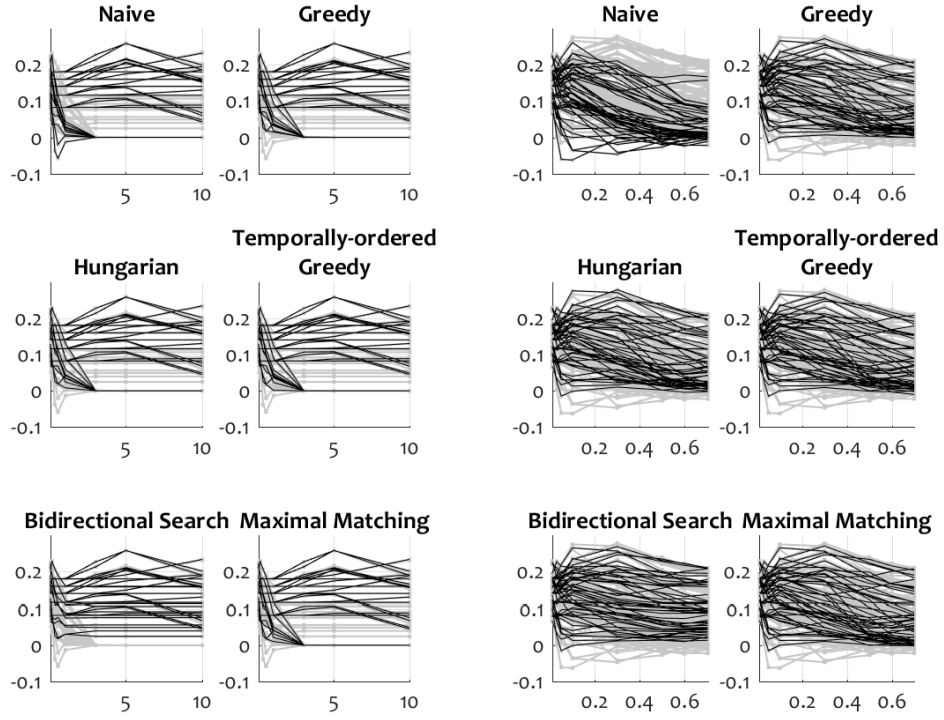


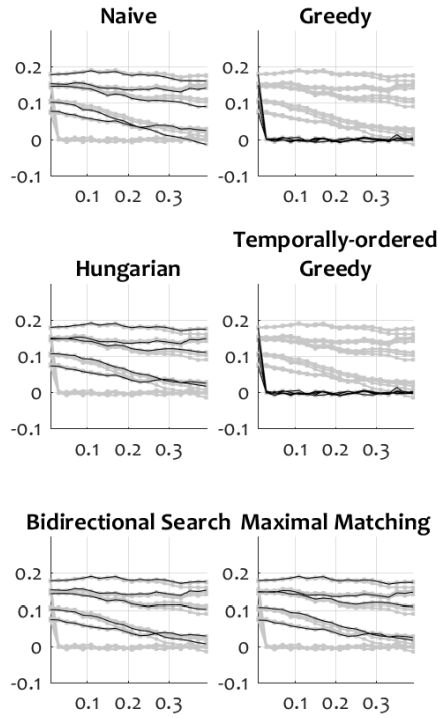
Figure 3.5: Discrimination capacity C_U as a function of the threshold (Manhattan distance) with the 11 feature spaces. For the first ten graphs, all curves corresponding to the 10 feature descriptors are shown in each graph at the same time. The respective feature descriptors are highlighted in black.

This will smear the difference between the F -values, leading to low C_U . The remaining 5 methods are not substantially different. Interestingly, the conservative matching methods - Greedy and temporally-order greedy, do not work well with the SURF features. Note that here we view all the results together, both good and bad. Further analyses show that the variability in the C_U for each matching method is not due to feature representations but to



(a) Euclidean distance

(b) Manhattan distance



(c) SURF feature distance

Figure 3.6: Visualisation of the C_U for the 6 matching methods.

summarisation method. The best such method, VSUMM₁, corresponds to the highest curves.

Based on these results, we can recommend any of the three matching methods: Hungarian (minimal-weight complete matching followed by thresholding); bidirectional search (The algorithm of Kannappan et al. [81]); and Maximal Matching (Hopcroft-Karp: The Hopcroft-Karp algorithm or any equivalent algorithm returning a maximal unweighted matching from the sub-threshold pairings). Of these, bidirectional search algorithm has the lowest computational complexity $O(n^2)$ compared with $O(n^3)$ for Hungarian, and with the maximal-matching method whose worst-case is $O(n^{2.5})$ if implemented as the Hopcroft-Karp algorithm, or $O(n^3)$ if implemented as algorithm 6. Hence we include the algorithm of Kannappan et al. in our proposed protocol.

3.4.5 The Proposed Protocol

Several authors (e.g. [28, 56, 112]) have followed the choice of feature descriptor, similarity metric, matching algorithm, and threshold pioneered by Avila et al. [40]. So far, there is no publication evidence of any theoretical or experimental basis for these choices. The choice of H16_1block feature representation, and threshold value $\theta = 0.5$ is reasonable, though the finer-grained H32_1block features outperforms it on average. The proposed evaluation framework is described in Table 3.3.

Table 3.3: Description of the proposed framework in terms of the classification taxonomy.

| Property | : Value |
|-------------------------------|---|
| Feature Representation | : <i>H32_1block</i> , 32-bin hue histogram (normalised to sum 1). |
| Similarity Metric | : Manhattan distance |
| Threshold | : $\theta = 0.3$ |
| Matching Strategy (Algorithm) | : Bidirectional search algorithm |
| Accuracy Metric | : <i>F</i> -measure |

Finally, in order to allow for a fair comparison between different summarisation algorithms, we propose the use of C_U as defined in equation (3.4). Suppose that there are two algorithmic methods giving summaries P and Q , respectively. One of them may have a larger F -value for its match to the ground truth (GT) only by virtue of the number of keyframes within. To guard

against this, C_U evaluates by how much an algorithm improves over a uniform summary of the same cardinality. Therefore, instead of comparing $F(P, GT)$ with $F(Q, GT)$, we propose to compare:

$$C_U(P) = F(P, GT) - F(U(|P|), GT)$$

with

$$C_U(Q) = F(Q, GT) - F(U(|Q|), GT),$$

where $U(s)$ is a uniform summary with s frames.

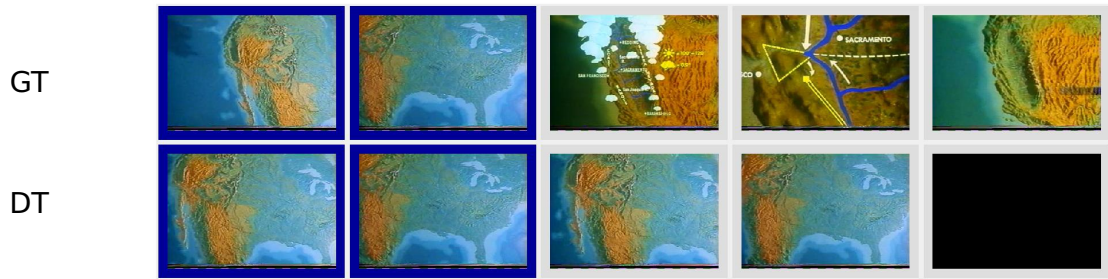
If the two rival keyframe summaries P and Q are of the same cardinality, their relative merit can be evaluated by $F(P, GT)$ and $F(Q, GT)$, but the question will remain whether they improve at all on a uniform (or another) baseline.

3.5 An Example

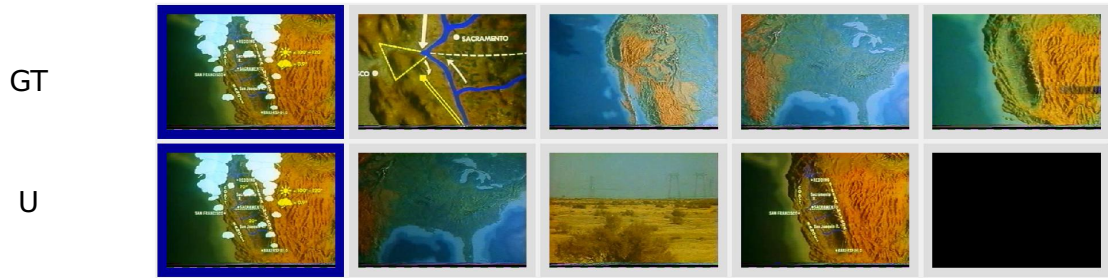
We now illustrate how the protocol can be used in practice⁸. Figures 3.7 to 3.11 show the summaries by the 5 algorithmic methods: DT, OV, STIMO, VSUMM₁, and VSUMM₂, together with the corresponding uniform summary of the same cardinality (the (b) plots). The matches are highlighted with a dark-blue frame. The images in the summaries are arranged so that the matching ones are on the left (recall that we treat the summary as a set, and not as a time sequence). The matches are calculated using the choices of methods and parameters of our proposed protocol. Table 3.4 shows the numerical results for the five methods, assuming that the only available ground truth is the summary of User #3. (Both the video and the user were chosen at random.)

While in this example the overall ranking of the five summarisation methods is the same according to $F(K, GT)$ and C_U , this will not in general be the case. Methods with higher C_U should be preferred. The F -value alone may lead to false claim of matching the ground truth, especially if $F(U(|K|), GT)$ happens

⁸MATLAB code is provided at: <https://github.com/LucyKuncheva/1-nn-editing>

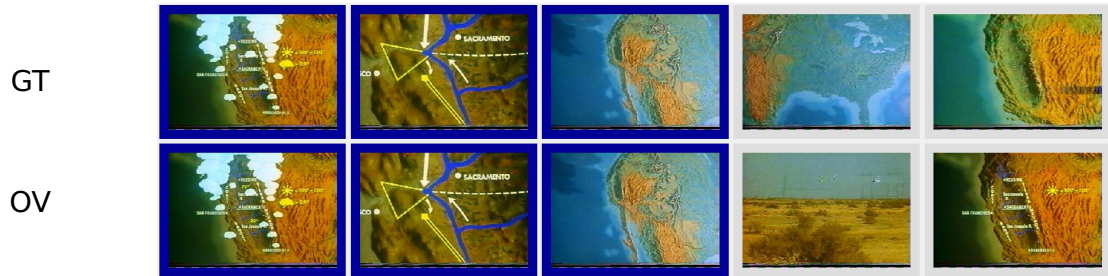


(a) DT summary: 2 matches

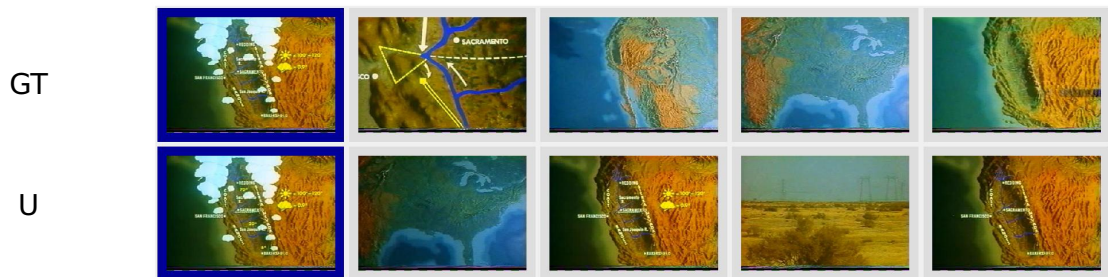


(b) Uniform summary $U(4)$: one match

Figure 3.7: Proposed protocol for Video #22, DT summarisation method, User #3 as a single ground truth. The matches are highlighted with a dark blue frame.

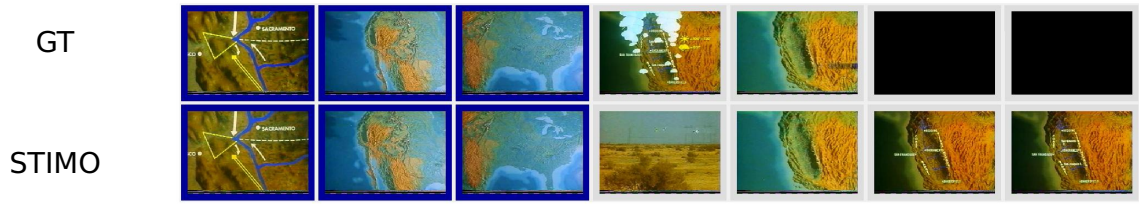


(a) OV summary: 3 matches

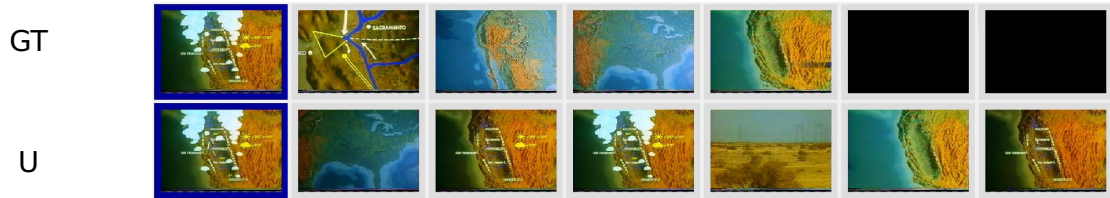


(b) Uniform summary $U(5)$: one match

Figure 3.8: Proposed protocol for Video #22, OV summarisation method, User #3 as a single ground truth.

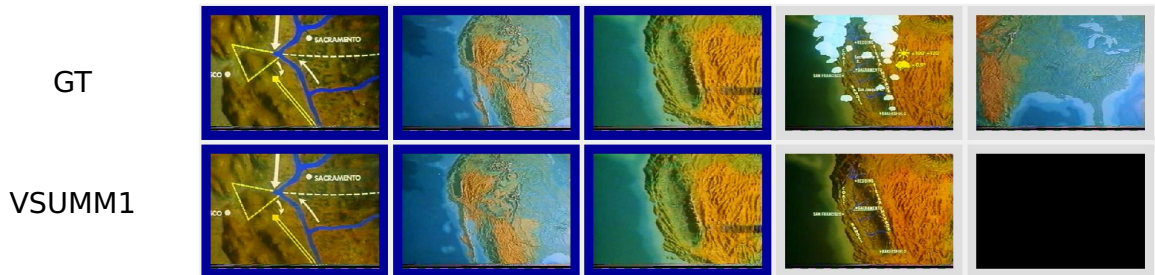


(a) STIMO summary: 3 matches

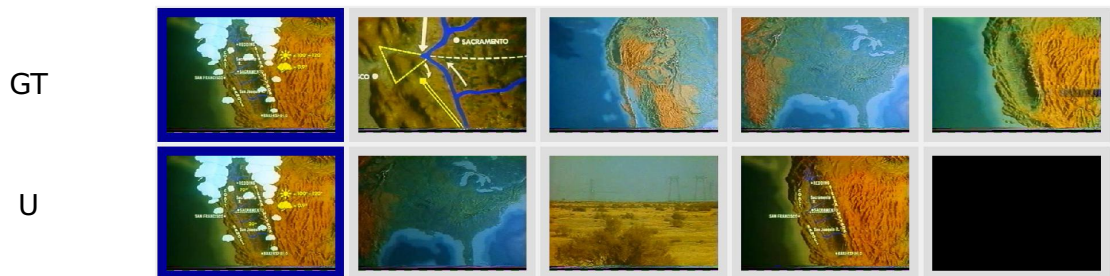


(b) Uniform summary $U(7)$: one match

Figure 3.9: Proposed protocol for Video #22, *STIMO summarisation method*, User #3 as a single ground truth.

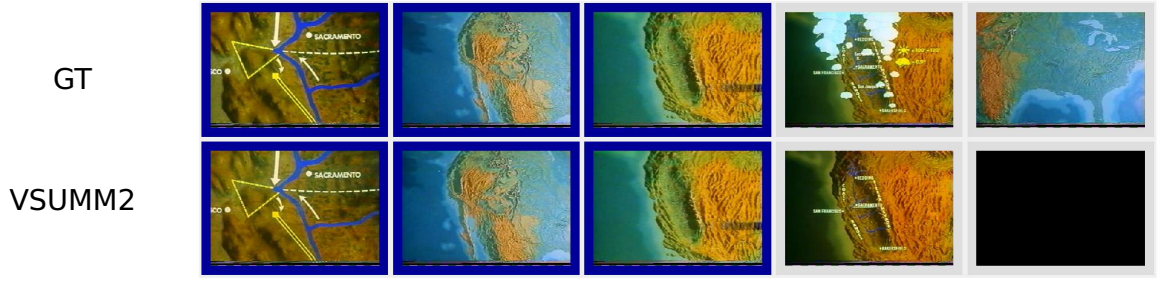


(a) VSUMM1 summary: 3 matches

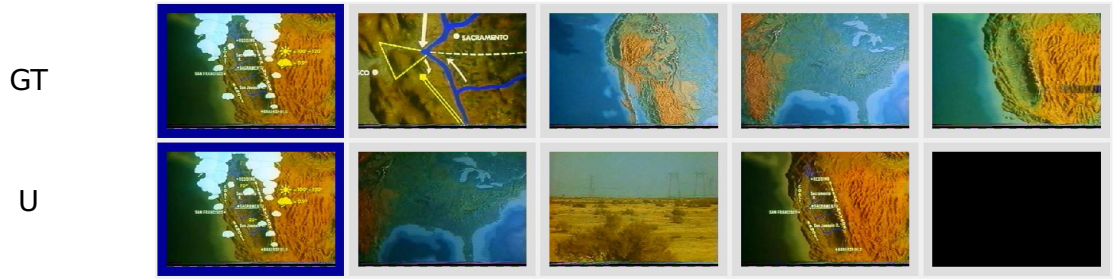


(b) Uniform summary $U(4)$: one match

Figure 3.10: Proposed protocol for Video #22, *VSUMM1 summarisation method*, User #3 as a single ground truth.



(a) VSUMM2 summary: 3 matches



(b) Uniform summary $U(4)$: one match

Figure 3.11: Proposed protocol for Video #22, *VSUMM2 summarisation method*, User #3 as a single ground truth.

Table 3.4: Calculation of the F -values and C_U for the 5 summarisation methods, based on the matches identified by the proposed protocol and illustrated in Figures 3.7–3.11.

| | $F(S, GT)$ | $F(U(S), GT)$ | C_U |
|-----------|---------------------------------|---------------------------------|-------|
| DT | $\frac{2 \times 2}{5+4} = 0.44$ | $\frac{2 \times 1}{5+4} = 0.22$ | 0.22 |
| OV | $\frac{2 \times 3}{5+5} = 0.60$ | $\frac{2 \times 1}{5+5} = 0.20$ | 0.40 |
| STIMO | $\frac{2 \times 3}{5+7} = 0.50$ | $\frac{2 \times 1}{5+7} = 0.17$ | 0.33 |
| $VSUMM_1$ | $\frac{2 \times 3}{5+4} = 0.67$ | $\frac{2 \times 1}{5+4} = 0.22$ | 0.45 |
| $VSUMM_2$ | $\frac{2 \times 3}{5+4} = 0.67$ | $\frac{2 \times 1}{5+4} = 0.22$ | 0.45 |

to be high. In some cases C_U is negative, which casts a doubt on the validity of the algorithm producing the keyframe summary K .

3.6 Conclusion

We have experimentally investigated a range of choices for different components of a protocol for evaluating the outputs of keyframe-extraction algorithms. A new measure called “discrimination capacity” C_U is proposed, which evaluates by how much a given summary improves on the uniform keyframe summary of the same cardinality. Using C_U and the VSUMM video collection, we offer empirical recommendations, and propose a full protocol for comparison of keyframe summaries, listed at the start of sub-section 3.4.5. A 32-bin hue histogram feature space fared better than the high-level features. Our study also contains a comprehensive collection of algorithms for matching (pairing) two summaries of different cardinalities.

Our future work will include looking into semantic comparisons between frames and summaries in addition to matching based solely on visual appearance. Combinations thereof as well as incorporating the time tag in the comparisons will be explored.

As an alternative to the ground truth based evaluation discussed above, a comparative evaluation technique can be used. This will be discussed in the following chapter.

Chapter 4

Closest-to-Centroid Baseline Method

4.1 Motivation

Subsequent to evaluating a keyframe summarisation method against the ground truth summary, the method must be compared with existing rivals. Comparison with rival methods can be carried out by having access: to their implementations, or their summary results. The former is not always straightforward if the method is designed to use a particular type of data. The latter can be facilitated by using a benchmark video data.

In traditional video, Avila et al. [40] published the results of different keyframe summarisation methods along with the full video data set, and user-formulated ground truth (benchmark). Yet, this option does not exist in the egocentric type of videos. Due to the limitation of using alternative (rival) summarisation methods, baseline methods can be adopted for this comparison.

4.2 Story-Line of Evaluating Keyframe Summarisations

At present, authors often develop a bespoke experimental set-up in which their proposed method for keyframe selection compares favourably to one or two alternative baseline methods. Table 4.1 lists chronologically 29 publications on keyframe video summarisation along with their comparison choice of other methods. Different choices of rival methods are enumerated in columns 1-5. The column 'Rivals' shows the choices of rival methods while the column

Table 4.1: An overview of keyframe summarisation methods with respect to their choices for comparative evaluation (Rivals), and their proposed method (Proposed). Summarisation studies proposed for FPV data set are highlighted in grey colour. The level of interest in each method is accumulated at the usage count.

| | | Rivals | | | | | Proposed | |
|---------------------------|-------|----------|--------|-----------|-------------------|------------------|---------------------|-------|
| | | Baseline | | | Other | | Closest-to-Centroid | Other |
| | | Uniform | Random | Mid-Frame | Bespoke method(s) | State-of-the-art | | |
| Zhuang et al. 1998 | [184] | | | | | | ✓ | |
| Hanjalic et al. 1999 | [63] | | | | | | ✓ | |
| Gong et al. 2000 | [57] | | | | | | ✓ | |
| Zhu et al. 2004 | [183] | | | | | ✓ | ✓ | |
| Yu et al. 2004 | [178] | | | | | | ✓ | |
| Mundur et al. 2006 | [119] | | | | ✓ | ✓ | ✓ | |
| Doherty et al. 2008 | [44] | | | ✓ | ✓ | | | ✓ |
| Herranz et al. 2009 | [67] | | | | | | ✓ | |
| Spyrou et al. 2009 | [152] | | | | | | ✓ | |
| Furini et al. 2010 | [53] | | ✓ | | ✓ | ✓ | | ✓ |
| Jojić et al. 2010 | [78] | | | | ✓ | | | ✓ |
| Avila et al. 2011 | [40] | | | | | ✓ | ✓ | |
| Almeida et al. 2012 | [8] | | | | | ✓ | | ✓ |
| Ejaz et al. 2012 | [50] | | | | | ✓ | | ✓ |
| Ejaz et al. 2013 | [49] | | | | | ✓ | | ✓ |
| Guan et al. 2013 | [59] | | | | | ✓ | | ✓ |
| Jinda et al. 2013 | [77] | | | ✓ | | | | ✓ |
| Khosla et al. 2013 | [84] | ✓ | ✓ | | ✓ | | ✓ | |
| Gong et al. 2014 | [56] | | | | ✓ | ✓ | | ✓ |
| Lakshmi Priya et al. 2014 | [135] | | | | | ✓ | ✓ | |
| Mahmoud et al. 2014 | [105] | | | | | ✓ | | ✓ |
| Xiong et al. 2014 | [172] | | ✓ | | | | | ✓ |
| Lee et al. 2015 | [92] | ✓ | | | | ✓ | | ✓ |
| Lidon et al. 2015 | [96] | ✓ | | | | | | ✓ |
| Mei et al. 2015 | [112] | | | | | ✓ | | ✓ |
| Ratsamee et al. 2015 | [138] | | | ✓ | ✓ | | | ✓ |
| Bolãnos et al. 2015 | [21] | | ✓ | | | | ✓ | |
| Bettadapura et al. 2016 | [14] | ✓ | | | | | | ✓ |
| Otani et al. 2016 | [127] | ✓ | | | ✓ | ✓ | ✓ | |
| Usage count | | 5 | 4 | 3 | 8 | 14 | 13 | 16 |

‘Proposed’ shows the methods proposed within the respective publication. Since we are specifically interested in the Closest-to-Centroid method (CC), we identified studies where this method has been used as a significant part

of the proposed keyframe summarisation method. All these studies have a tick mark in the column ‘Proposed/Closest-to-Centroid’.

We also summarise studies where CC is not the proposed methods. This was done to explore the choices of rival methods across the whole field. For these studies, we placed a tick mark in column ‘Proposed/Other’.

In column ‘Rivals’, we distinguish between two types of methods: commonly accepted baseline methods (‘Rivals/Baseline’) and other methods (‘Rivals/Other’). State-of-the-art methods were previously published ones. Bespoke methods were unpublished alternatives proposed within the study as a weaker variants of the main method.

Summarisation studies proposed for FPV data set are highlighted in this table. Comparison strategies can be either based on user-study or on ground-truth annotations.

Typical choices for baseline methods are Random (R), Uniform (U), and Mid-Event (ME) selection. For R and U selection methods, the number of keyframes (K) is set in advance. For a fair comparison, the value K is usually equivalent to the number of keyframes generated by the proposed summary method. In R selection, K frames are randomly selected from the video frames regardless of their temporal positions. While for U selection, the video is uniformly divided into K equal segments, and the middle frame in each segment is taken for the summary. The ME summarisation method requires the video to be already segmented into temporally coherent units (events), either by an automatic event segmentation method, or manually by a user. Thereafter for each event, the middle frame (time-wise) is chosen to represent the summary of that event. The information required to implement R and U is only the number of keyframes, which makes the implementation task relatively easy. ME, on the other hand, requires prior segmentation of the video into units (events), which is a difficult task in its own right, even more so for egocentric and life-logging data.

As can be observed from column ‘Rivals/Other/State-of-the-art’ of this table, evaluating a new summarisation method against the other prior techniques is not generally practised until Avila et al. [40] organised the VSUMM data set with its annotations in 2011. While this study does not claim that all papers on the subject are included, we offer an interesting statistic. Out of the top 11 entries, three studies compared their results with the result of a state-of-the-art method at the time (27%). After that, 11 out of the 18 studies report comparisons with state-of-the-art (61%).

For FPV data, authors frequently used the baseline methods, or develop their bespoke methods alternative to their proposal. These baseline methods are arguably easy to beat.

The CC method has been often used in the past as witnessed by column ‘Closest-to-Centroid’ of Table 4.1. Using CC, some authors compared their summary results against U and/or R baselines [84, 21, 127], which suggests that CC was considered in the past a higher quality of summary compared to the typical baselines.

The popularity of the use of CC, encouraged us to develop CC into a baseline keyframe selection method. We choose a large variety of feature descriptors including: colour, texture, shape, motion, and complex features, and conduct the experiment to ensure the higher performance of CC compared with U and ME baselines. It is important to note that R is intentionally not taken forward because it is deemed to be the weakest baseline [179, 84]. Here the focus is on evaluation of keyframe video summaries exclusively for FPV data, which so far includes no consensus on protocols, benchmarks, and baseline models. Therefore, a generic matching protocol is additionally designed to evaluate the merit of the keyframe summaries.

4.3 Closest-to-Centroid Baseline

Let $V = \langle f_1, \dots, f_N \rangle$ be the video stream of N frames, where each frame is indexed by its time tag. Assume video frames are described in some feature

space, and each frame is represented by a feature vector in an n -dimensional space, $\mathbf{x}(f_i) \in \mathbb{R}^n$. Frames with similar content have smaller distance among each other than frames with different content. To simplify notation, we will use just \mathbf{x}_i to represent frame f_i .

Let $I_k \subset \{1, 2, \dots, N\}$ be the index set of consecutive time tags identifying event k from the total of K events, $k = 1, \dots, K$.

The proposed baseline model returns the frame closest to the centroid of each event. We refer to the events as “clusters” although they may not form a conventional cluster structure in \mathbb{R}^n . Formally, the summary is the collection of ordered indices $S = \langle s_1, \dots, s_K \rangle$ where

$$s_k = \arg \min_{m \in I_k} \{d(\mathbf{x}_m, \mathbf{c}_k)\}, \quad (4.1)$$

$d(., .)$ is a chosen distance metric in \mathbb{R}^n , and

$$\mathbf{c}_k = \frac{1}{|I_k|} \sum_{j \in I_k} \mathbf{x}_j$$

is the centroid of cluster (event) k .

4.4 Feature Representations

A crucial component of any keyframe selection method is the choice of feature representations. Following the literature, features are divided into two groups: features that describe the *content* of the frame; and features that evaluate the image *quality* or *aesthetics*¹ [44]. Note that the two groups are not completely non-intersecting; they likely share low-level features. Here we are interested in the former group.

The content type feature descriptors are further divided into three categories:

¹Image quality or aesthetic attributes are either combinations of low-level features [121, 15] such as: sharpness, colour harmony, noise, eye sensitivity, brightness, blurriness, dark channel, or generated by deep neural network [16, 142].

- Low-level (context-free) features such as colour, texture, shape, motion, and regions;
- Mid-level features extracted through deep learning; or
- High-level information (context-involved) such as face recognition descriptors, and semantic features.

Further, the original feature descriptors may be transformed through Principal Component Analysis (PCA), Discrete Cosine Transform (DCT) [35], Singular Value Decomposition (SVD) [58], or Kernel-based Principal Component Analysis (K-PCA) [178].

Typical choices in the low-level feature group are colour information² [35, 165, 50, 135], colour histograms [153, 164, 53, 40, 165, 50, 103, 173], edge feature³ [165, 135], MPEG-7 descriptors⁴ [152, 67], HOG pyramid [103], SURF [103], SIFT [100, 165, 150, 56, 173], Gist [103, 34, 145], and motions [101, 103, 173]. Using convolutional deep learning networks (CNN) is a leading feature extraction method for video summarisation due to its ability to learn an advanced set of features [21, 161, 127, 96, 162], and therefore we include CNN in our experiments. Specific domain high-level features (context-involved) used in video summarisation can be listed as follows: faces [30, 91, 103, 92, 173], objects [30, 91, 103, 92], famous landmarks [60], and visual thesaurus [152].

Feature representation for a baseline method must be easy to extract. In this study we chose the features shown in Table 4.2.

CNN features, although sometimes perceived as ‘high-level’, don’t carry by themselves semantic information. This is why they are classed as mid-level in our study.

To extract colour properties, following descriptors are calculated:

²The descriptors can include one or a combinations of colour moments, dominant colour, scalable colour, and colour correlogram.

³The descriptors can be contained one or a combinations of edge distribution histogram, and wavelet transform.

⁴The descriptors are contained of colour layout, colour structure, scalable colour, and edge histogram.

Table 4.2: The main characteristics of the evaluated feature representations.

| Feature Type | Visual Information | Acronym | Size |
|--------------|--------------------|-------------------------|------|
| Low-Level | Colour | 1. ACC | 1024 |
| | | 2. CEDD | 144 |
| | | 3. CLD | 118 |
| | | 4. FCTH | 192 |
| | | 5. FOH | 576 |
| | | 6. GIST | 960 |
| | | 7. HSV ^{ch} | 32 |
| | | 8. JCD | 168 |
| | | 9. RGB ^{ch} | 512 |
| | | 10. RGB ^{cm} | 54 |
| | | 11. SCD | 64 |
| | Texture | 12. EHD | 80 |
| | | 13. Gabor | 60 |
| | | 14. LBP | 256 |
| | | 15. LBP ^{riu2} | 36 |
| | | 16. Tamura | 18 |
| | Shape | 17. PHOG | 630 |
| | Motion | 18. HMP | 6075 |
| Mid-Level | Corners and edges | 19. FV | 4096 |
| | Objects | 20. CNN | 4096 |

1. *Auto Colour Correlogram* (ACC) [69] measures the spatial correlation of colour changes between different pixels in the image with respect to the changes in their distances. This descriptor differs from the colour histogram which only captures colour distributions. For a given image, colour values are quantised into bins. Then distances between each pair of pixels are calculated. In simple terms, a colour correlogram of an image is an indexed table of colour pairs, where each value of the matrix specifies the probability of finding the pixel pairs at that distance in the image.

2. *Colour and Edge Directivity Descriptor* (CEDD) [32] is a global descriptor. First, the image is divided into rectangular areas. For each block, colour and texture information are extracted, and represented by a quantised vector. After calculating the quantised vectors for all blocks, they are combined (fused) to generate a single feature vector. The final descriptor is generated after normalising and quantising the feature vector into predefined levels.

3. *Colour Layout Descriptor* (CLD) [108] is a resolution-invariant MPEG-7 visual descriptor, which represents the spatial distribution of colours in YC_bC_r colour space. Given an image, it is divided into 8×8 blocks and the average colour values of each block is calculated. Using DCT, the average values are quantised into three sets of 64 DCT coefficients, and a few low-frequency coefficients are chosen by zigzag scanning. Feature dimension is based on the number of coefficients for each component.

4. *Fuzzy Colour and Texture Histogram* (FCTH) [31] is calculated from the combination of three fuzzy units. A given image is initially segmented into a number of blocks, and all fuzzy units are passed from each block. The image is represented into HSV colour space. For the first unit, a Fuzzy-Linking histogram is extracted by applying a set of fuzzy rules to the image. This generates a 10-bin histogram. In the second unit, the 10-bin histograms are expanded into 24-bin histograms applying a two-input fuzzy system. As the third unit, each block is transformed using Haar Wavelet transform which generates a set of texture elements. The elements are used to convert the 24-bin histogram into 192-bin histogram.

5. *Fuzzy Opponent Histogram* (FOH) [141] includes shift-invariant colour models regarding light intensity. An RGB image is represented with two colour information channels and one intensity component. Histograms of each component are calculated, and combined based on the opponent colour space. A fuzzy system is applied on the histogram to generate the descriptor vector.

6. *GIST* [126] is a low dimensional representation of the scene. Initially, the image is convolved with a 32 Gabor filter of 4 scales and 8 orientations. This produces feature maps which are later divided into regions. The average feature values for each region are calculated and concatenated to produce the feature vector. To extract the GIST features, we used the Lear's GIST implementation⁵.

⁵The Lear's GIST implementation is available at: https://lear.inrialpes.fr/src/lear_gist-1.2.tgz (As of August 2019)

7. *HSV Colour Histogram* (HSV^{ch}) captures colour distributions. The HSV^{ch} features refer to a colour histogram computed only from the hue value (H) of the HSV colour space after its uniform quantisation into 32 colour bins.

8. *Joint Composite Descriptor* (JCD) [33] combines the two CEDD and FCTH descriptors into one histogram.

9. *RGB Colour Histogram* (RGB^{ch}) [154] is relatively invariant with translations and rotations about the viewing axis. This descriptor is calculated by combining three histogram of colour channels R, G, and B.

10. *RGB Colour Moments* (RGB^{cm}) measure colour distribution in an image. Image is divided uniformly into a 3-by-3 grid of blocks. Thereafter, the mean and the standard deviation for each block and each colour are computed (9 blocks \times 3 colour \times 2 statistics = 54 features).

11. *Scalable Colour Descriptor* (SCD) [108] is a Haar-transform based encoding and measures colour distribution over the entire image. A given image is typically converted into some colour space, and uniformly quantised to generate a histogram. Thereafter, the histogram values are normalised and non-linearly mapped into a four bits integer representation. Finally the histogram is encoded applying Haar-transform across the histogram bins. When the full resolution is not required, the representation size can be reduced by limiting the extracted number of Haar coefficients from the histogram bins of 128, 64 or 32.

The descriptors for encoding texture properties are calculated as follows:

12. *Edge Histogram Descriptor* (EHD) [108] represents spatial distributions of edges. An image is split into 4×4 blocks, and edge histograms are computed for each block. To calculate the edge histogram, in each block edges are quantised into 5 bins related to their directions: vertical, horizontal, 45° diagonal, 135° diagonal, and isotropic.

13. *Gabor* features [109] are used for texture representation and discrimination. A Gabor filter is a sinusoidal signal modulated by a Gaussian, with predefined frequency and orientation. The filter is passed through an image to generate a set of features.

14. *Local Binary Patterns* (LBP) [124] are a powerful descriptor to extract texture information. A given image is first divided into blocks. For each pixel in a block, it compares the pixel value with all of its 8 neighbour values. If the value is greater than the neighbour value, 0 is written otherwise 1 is written. Doing this, an 8-digit binary value is obtained, which can also be presented as a decimal number. For each block, the histogram is generated, normalised and then concatenated.

15. *Rotation Invariant Local Binary Patterns* (LBP^{riu2}) [125] measure spatial structure of image texture. This descriptor is similar to LBP with addition of circularly shifting the binary code (neighbours) by a predefined number of steps.

16. *Tamura* features [156] are motivated by the psychological studies on human visual perception of textures. Given an image, a feature vector is calculated by combining six basic texture features of coarseness, contrast, directionality, line likeness, regularity, and roughness.

For encoding shape information, we use:

17. *Pyramid of Histogram of Oriented Gradients* (PHOG) [24]. The image is decomposed into sequence of sub-regions at several pyramid levels where at each level of the pyramid, there are number of sub-regions. To form a pyramid at level l , the image is divided into finer spatial grids by doubling the number of divisions (total of 2^l) along each dimension. The feature vector is computed by combining the histograms of edge orientations gradients of each sub-regions.

Descriptors 1-17 (except 6) were extracted using the LIRE library⁶ [104].

18. *Histogram of Motion Patterns* (HMP) [7] was also considered as a spatio-temporal descriptor to encode motion information. Given a MPEG video, first intra-coded (I-frame) frames are extracted⁷. Each I-frame consists of small processing units called macro-blocks, which can be used to obtain motion information. For each I-frame, the I-frames positioned in both sides are scanned (previous-current-next), and their corresponding macro-blocks are analysed to form an ordinal matrix. The ordinal matrix later encoded into a histogram to represent a spatio-temporal descriptor.

Finally, we evaluated two mid-level representations. One is based on visual dictionaries, and the other is CNN, detailed as follows:

19. *Fisher Vectors* (FV) [73] encode local features as visual words. To create the visual dictionary, local patches were extracted with a Hessian-affine detector and described by SIFT descriptors [102], which were reduced using PCA and then used to create a codebook with 64 visual words learned by Gaussian Mixture Models (GMM). A global representation of a video frame is obtained by accumulating the residual vectors. The difference of each reduced SIFT descriptor and the mean vector of the Gaussian distribution assigned to each visual word was calculated. These differences were concatenated into a single feature vector, which was subsequently power normalised⁸ and then L_2 -normalised. The GMM computation and FV encoding were performed using the Yael library⁹ [48].

20. *CNN* are features extracted by a convolutional neural network. The 4096 deep features were extracted right before the classification (soft-max) layer, from the response of the Fully Connected layer (FC7) of the CNN. The runner-up in ILSVRC 2014, known as VGGNet architecture [147], was chosen to train

⁶The LIRE library is available at: <http://www.lire-project.net> (As of August 2019)

⁷A MPEG video is composed of mainly three types of frames that serve different purposes: intra-coded (I-frames), predicted (P-frames), and bidirectionally predicted (B-frames).

⁸This is obtained by applying the function $\text{sign}(z)|z|^\rho$ with $0 \leq \rho \leq 1$ to each dimension of the FV. It is also called signed square-rooting with $\rho = 0.5$.

⁹The Yael library is available at: <http://yael.gforge.inria.fr> (As of August 2019)

the network. This network contains 16 hidden (Conv/FC) layers. In order to extract the neural network features, we used MatConvNet [163].

4.5 An Experiment with an Egocentric Video Database

The purpose of this experiment is to identify a feature representation among the chosen 20 representations in Table 4.2, where CC is markedly better than U and ME. In doing so, we also contribute a method for comparing keyframe summaries based on the visual appearance of the frames. The assumptions in this experiments are:

1. The video has been already segmented into temporally coherent events.
2. One frame per event is selected in the summary.
3. There is a ground truth of representative frames (one per event).

4.5.1 Data Set

The UTEgo data set [91] contains 4 videos (each lasting about 3-4 hours) of subjects performing their daily activities such as driving, shopping, attending lectures and eating¹⁰. The data set is challenging because it contains frequent changes of the illumination and the camera position. The videos were recorded at 15 frames/second with 350×480 resolution per frame. We sub-sampled each video taking one frame per four seconds, thus reducing the number of frames as follows:

- P01 , 3464 frames, 14 events.
- P02 , 4566 frames, 19 events.
- P03 , 2696 frames, 10 events.
- P04 , 4446 frames, 16 events.

¹⁰This benchmark data set has been used as a sole experimental test bed in many studies on egocentric video summarisation.

Each video was segmented into events using Semantic Regularised Clustering (SR-Clustering) [42]¹¹. We only used the contextual information extracted from a pre-trained CNN (AlexNet [85]) as the CNN model, and run through the deep learning framework Caffe [74]).

A ground truth summary was constructed for each video. A user picked a frame for each event so that the events are faithfully represented and still discernible within the video.

4.5.2 Matching Procedure

Our matching procedure is intended to pair two frames recorded by egocentric camera *for the same event* with respect to their visual appearance. While there are many possibilities, we chose SURF features [12] on the grey image to match objects and shapes as done before [77, 138], and HSV histograms (following the protocol by Avila et al. [40] and explained in the Chapter 3) to match the colour distribution.

Let f_1 and f_2 be the frames being compared. Denote by p_1 and p_2 the number of SURF points of interest in the respective frames. Let m_1 be the number of matches found from f_1 to f_2 , and m_2 , the number of matches from f_2 to f_1 . The matching score from the SURF features is taken to be

$$S_{\text{SURF}} = \frac{m_1 + m_2}{p_1 + p_2}.$$

The two frames are considered matching on SURF features if $S_{\text{SURF}} > \theta_{\text{SURF}}$, where $\theta_{\text{SURF}} \in [0, 1]$ is a threshold.

For the HSV feature space, a 32-bin histogram of the hue value was calculated for each frame. The bin counts were normalised so that the sum was 1 for each histogram. Let $B_j = \{b_{j,1}, \dots, b_{j,32}\}$ be the normalised histogram for f_j , $j = 1, 2$. The L_1 distance was calculated by

$$D_H = \sum_{i=1}^{32} |b_{1,i} - b_{2,i}|.$$

¹¹<https://github.com/MarcBS/SR-Clustering> (As of August 2019)

The two frames are considered matching on HSV features if $D_H < \theta_H$, where $\theta_H \in [0, 2]$ is a threshold.

To ensure that the frames are a true visual match they must be a match on the objects/shapes (SURF) as well as colour (HSV). Because of this conservative rule, we pick threshold values which will allow for a fairly liberal match on each components: $\theta_{\text{SURF}} = 0.05$ and $\theta_H = 0.6$.

To illustrate the matching method, we show in Figure 4.1 the results for matching the ground truth and the uniform, mid-event and CC (PHOG) summaries of video P03. The matched frames are highlighted in red.

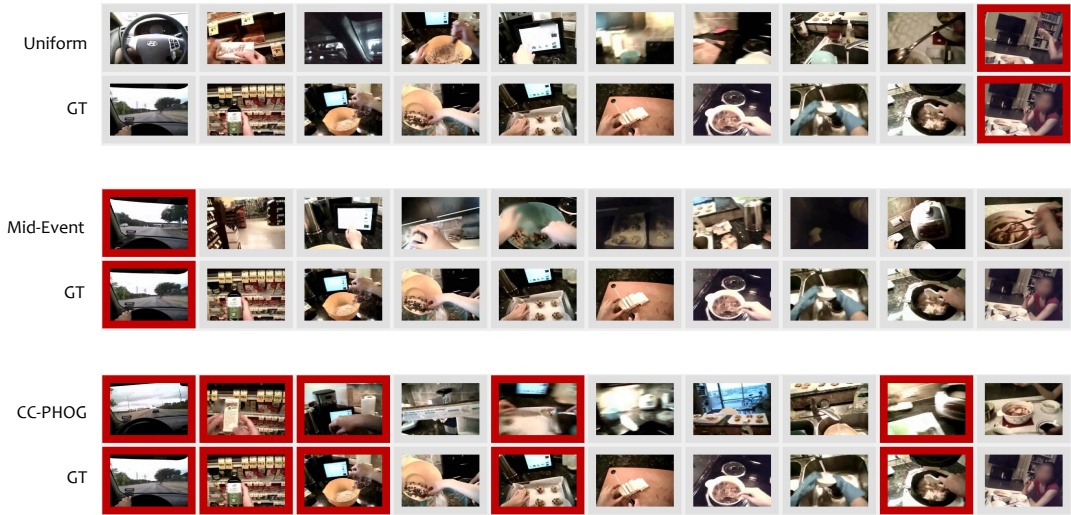


Figure 4.1: Illustration of the results from the matching procedure on the 10 events for video P03. The matches are highlighted with a red frame.

Finally, the match between the *summaries* can be calculated as the F-measure, which in this case reduces to the proportion of matches. For the examples in Figure 4.1, $F = \frac{1}{10} = 0.1$ for U and ME, and $F = \frac{5}{10} = 0.5$ for CC with PHOG features.

4.5.3 Results

We identified the CC summary for each descriptor, and quantified its proximity to the ground truth using the above matching procedure. Additionally, we prepared three alternative versions for each feature. We applied PCA and retained components explaining respectively 95%, 90% and 80% of the

Table 4.3: F-measure (in%) for the 4 videos for the U, ME and CC summaries with respect to the ground truth.

| Features | P01 | | | | P02 | | | | P03 | | | | P04 | | | | Average | |
|------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|---------|------|
| | Org | P95 | P90 | P80 | Org | P95 | P90 | P80 | Org | P95 | P90 | P80 | Org | P95 | P90 | P80 | Org | All |
| 1 ACC | 36 | 36 | 36 | 36 | 21 | 11 | 11 | 11 | 10 | 10 | 10 | 10 | 50 | 44 | 50 | 44 | 26.6 | 29.2 |
| 2 CEDD | 14 | 14 | 14 | 36 | 11 | 11 | 11 | 11 | 10 | 10 | 10 | 10 | 50 | 44 | 50 | 44 | 21.9 | 21.2 |
| 3 CLD | 7 | 7 | 7 | 7 | 16 | 11 | 11 | 5 | 0 | 0 | 0 | 0 | 19 | 25 | 19 | 19 | 9.6 | 10.5 |
| 4 FCTH | 14 | 14 | 14 | 21 | 5 | 5 | 16 | 16 | 40 | 30 | 20 | 10 | 38 | 50 | 44 | 44 | 21.8 | 24.2 |
| 5 FOH | 14 | 14 | 14 | 14 | 0 | 5 | 0 | 11 | 0 | 0 | 10 | 10 | 38 | 38 | 38 | 31 | 14.8 | 13 |
| 6 GIST | 21 | 14 | 21 | 7 | 0 | 0 | 0 | 0 | 10 | 10 | 0 | 10 | 31 | 31 | 31 | 31 | 13.6 | 15.5 |
| 7 HSVch | 29 | 29 | 21 | 29 | 11 | 11 | 16 | 16 | 30 | 40 | 10 | 20 | 38 | 31 | 38 | 31 | 25 | 27 |
| 8 JCD | 21 | 21 | 21 | 21 | 16 | 21 | 21 | 21 | 0 | 0 | 0 | 20 | 56 | 44 | 44 | 44 | 23.2 | 23.2 |
| 9 RGBch | 29 | 29 | 21 | 21 | 5 | 0 | 0 | 0 | 10 | 10 | 10 | 10 | 25 | 19 | 25 | 19 | 14.6 | 17.2 |
| 10 RGBcm | 14 | 14 | 14 | 7 | 16 | 21 | 21 | 16 | 10 | 10 | 20 | 20 | 50 | 31 | 38 | 31 | 20.8 | 22.5 |
| 11 SCD | 21 | 14 | 21 | 7 | 5 | 5 | 5 | 21 | 0 | 0 | 20 | 10 | 38 | 44 | 25 | 44 | 17.5 | 16 |
| 12 EHD | 29 | 29 | 21 | 21 | 16 | 16 | 16 | 16 | 20 | 20 | 10 | 10 | 50 | 44 | 44 | 50 | 25.7 | 28.7 |
| 13 Gabor | 21 | 21 | 21 | 21 | 5 | 5 | 0 | 0 | 20 | 10 | 10 | 10 | 19 | 25 | 25 | 25 | 14.9 | 16.2 |
| 14 LBP | 14 | 21 | 29 | 29 | 11 | 16 | 16 | 11 | 10 | 10 | 10 | 10 | 44 | 38 | 44 | 38 | 21.9 | 19.7 |
| 15 LBPriu2 | 21 | 14 | 14 | 14 | 32 | 21 | 5 | 5 | 10 | 0 | 0 | 10 | 38 | 19 | 19 | 31 | 15.8 | 25.2 |
| 16 Tamura | 29 | 14 | 36 | 21 | 5 | 5 | 11 | 11 | 0 | 0 | 10 | 10 | 38 | 25 | 25 | 25 | 16.6 | 18 |
| 17 PHOG | 29 | 29 | 29 | 29 | 11 | 16 | 5 | 0 | 50 | 50 | 40 | 40 | 38 | 44 | 44 | 50 | 31.5 | 32 |
| 18 HMP | 21 | 14 | 14 | 0 | 0 | 0 | 5 | 11 | 20 | 20 | 10 | 10 | 44 | 31 | 38 | 38 | 17.2 | 21.2 |
| 19 FV | 29 | 21 | 21 | 21 | 0 | 16 | 16 | 16 | 20 | 20 | 20 | 20 | 44 | 31 | 31 | 31 | 22.3 | 23.2 |
| 20 CNN | 7 | 7 | 7 | 21 | 0 | 0 | 5 | 5 | 20 | 20 | 20 | 0 | 38 | 38 | 44 | 38 | 16.9 | 16.2 |
| Uniform | 7 | | | | 16 | | | | 10 | | | | 13 | | | | 11.5 | |
| Mid-event | 7 | | | | 11 | | | | 10 | | | | 25 | | | | 13.2 | |

variability of the data. The the CC summaries were obtained, and the F-measure value was calculated for these additional reduced features. The results are shown in Table 4.3. The higher the values, the better the descriptor. We have shown for comparison the F-measures for the two baseline methods we contrast CC against: the uniform summary (U) and the mid-event summary (ME). Ideally, all F-values for CC will be higher than those for U and ME.

The results show that many descriptors lead to CC which matches the ground truth better than U or ME. The effect of PCA is not consistent. Sometimes the F-measure increases with the transformation and retaining the fewer features, and sometimes the effect is the opposite, both for the same feature space and different videos (e.g. the Gabor descriptor). To show the overall performance of the features, we averaged the F-values across the videos, first for only the original features and then for the 4 variants of each feature (across the columns of the table). Figure 4.2 shows the averaged across all values for the CC baseline method for the 20 feature spaces. The U and ME baselines are represented by horizontal lines as they do not depend on the feature spaces.

With small exceptions, the feature spaces are suitable for the CC baseline as the F-values for CC are higher than those for U and ME. The best feature

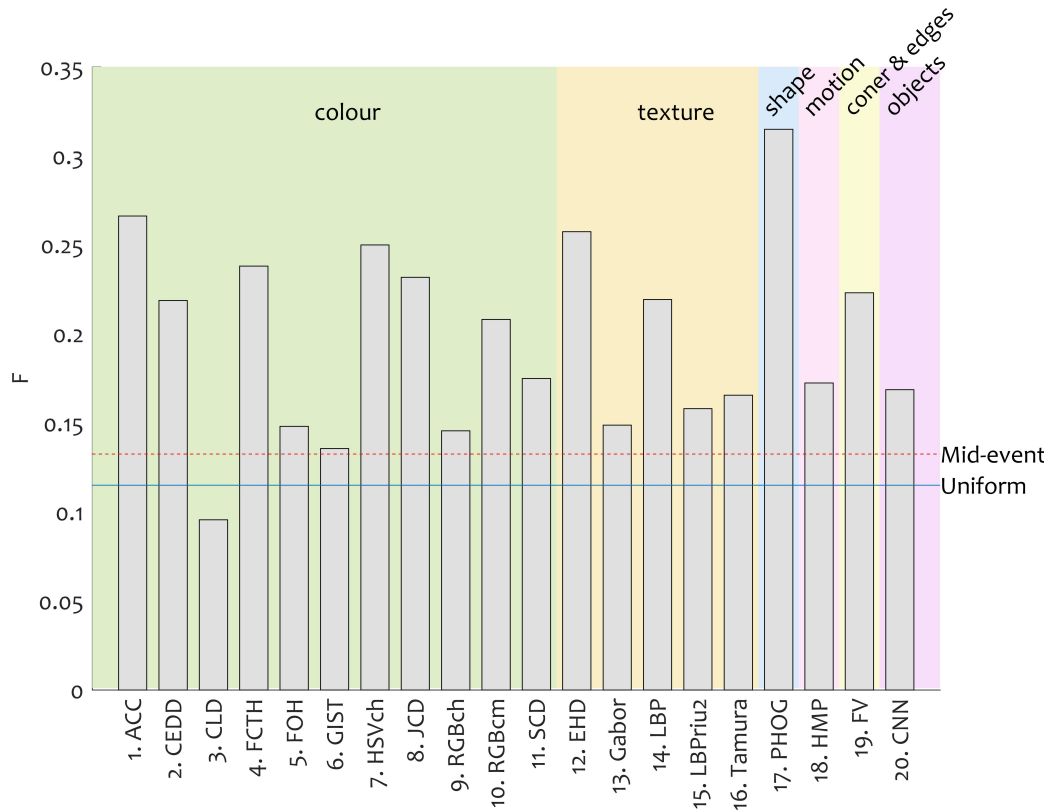


Figure 4.2: Averaged F measure comparing for the proposed baseline method (CC) and the ground truth for the 20 feature spaces. The F-values for U and ME are also shown for comparison.

space in this experiment happens to be PHOG. This can be explained with the fact that the SURF features used as a part of the matching procedure also account for the shapes in the frames. The same argument can be put forward for HSVch. The highly acclaimed CNN feature space showed a modest improvement of CC over U and ME. Note that lower values of the F-measure do not mean that the respective feature space is flawed. The F-values give us grounds for recommending a particular feature space for the CC baseline against which “proper” keyframe selection methods should be compared. Based on the results of this experiment, we recommend 17. PHOG, 1. ACC, 12. EHD, 7. HSVch and 4. FCTH.

4.6 Conclusion

Here we address one of the most acute problems in video summarisation: automatic evaluation of keyframe summaries. We propose a baseline model, Closest-to-Centroid and advocate its use instead of the weaker baselines

widely used thus far – the Uniform and the Mid-event selections. In addition, we propose an evaluation framework to compare summaries where each event is represented by a single keyframe.

The main limitations of CC and the matching procedure are as follows: the video must be already split into events; the matching procedure addresses only visual similarity between the frames.

Future experiments may refine the choice of a feature space for CC and the parameter values for the matching procedure. The CC can be applied to semantic feature spaces provided that those can be suitably quantified and equipped with a distance metric. To make the CC baseline even more competitive, an image quality component can be added to the Closest-to-Centroid criterion.

The following chapter will examine the use of prototype selection for the nearest neighbour classifier in selecting a keyframe summary.

Chapter 5

A Prototype Selection Technique for Video Summarisation

5.1 Generic Summary

The term 'Generic' means here that the summary is not prompted by a specific theme or purpose. It is meant to represent the whole content of the video.

The main properties to describe a video summarisation method are shown in Figure 2.2. Three additional topics required to describe a generic video summarisation, shown in Figure 5.1.

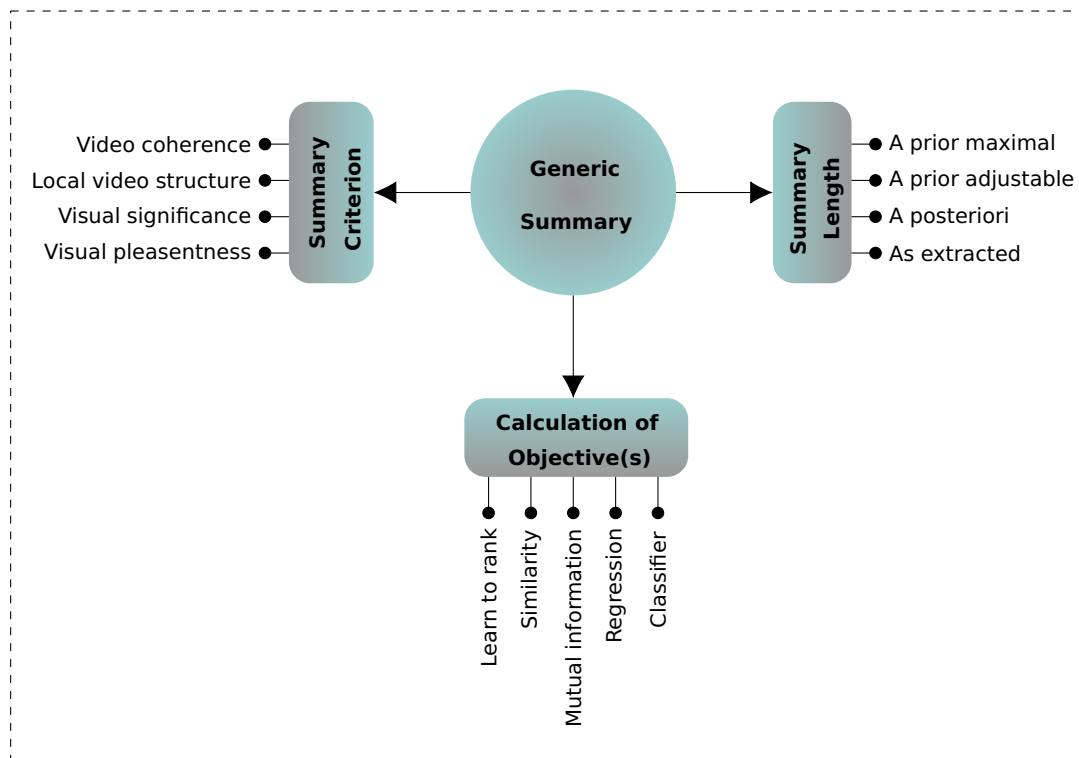


Figure 5.1: A classification of generic video summarisation methods.

• *Summary criterion.* To select the most appropriate frames for the summary, certain objectives are defined. Objectives can be characterised as a single function [60, 21, 175], or multiple functions [103, 61, 92, 96, 174]. These objectives can also be categorised based on their type into:

- *Video coherence*, such as uniqueness [92, 144, 96, 145] or diversity [103, 174] of events; temporal uniformity [61]; temporal smoothness and co-occurrence relation between story-elements [173]; and influence of consecutive subshots [103];
- *Local video structure*, such as representativeness [84, 61, 21];
- *Visual significance*, such as object-driven importance [103, 92, 61, 162], relevance [96], gaze fixation [174], interestingness [60, 175]; and
- *Visual pleasantness*, such as aesthetics [14], canonical views [84].

Uniqueness is typically employed in combination with other objectives, and defined to prevent redundancy in the selected frames. It calculates the degree of similarity among consecutive frames. Similarly, diversity penalises sequential subshots with similar scenes. While uniqueness and diversity are associated with differences of frame representations (distance) or their relations (covariance), uniformity is related to the frames indexes.

Focusing on the local structure of a video, representativeness selects the most repetitive frame of an event (in a video).

The visual significance is about the camera-wearer's visual attention and interactions. Object-driven importance gives higher scores to frames containing objects, and people which the camera wearer has interacted with. Comparably, relevance gives higher scores for frames including a larger number of salient objects, and faces to implicitly answer generic summarisation questions such as: *What is the user doing? Where is the user? Whom is the user interacting with?* Interestingness is a vague term and can cover image aesthetics (e.g. colourfulness, rule of third, symmetry), quality (e.g. image sharpness, blurriness), attention score, presence of landmarks, faces and significant objects interacted with. As opposed to the interestingness,

gaze fixation assigns higher scores to the subshots containing more frames with fixation.

The category-specific term ‘canonical view points’, means having various views of an object in order to capture informative images or videos. Canonical views can be advantageous in commercial use (e.g. selling a car) by selecting the most informative frames on different angles from the same object.

- *Calculation of Objective(s)*. Appropriate frames are discriminated from the rest of the frames in a video, using objective functions. These can be calculated explicitly by: learning/ calculating objective weights to rank frames or subshots individually (learn to rank) [103, 61, 14, 96, 174, 175], computing similarity [84, 21]; or implicitly by: modelling mutual information [174], regression [60, 92] or classifier [84].

Assume we have segmented a video into subshots (or events), and have the ground truth summary which includes selected frames related to the predefined objective(s). For each subshot, we want to select the most informative but also concise number of frames. The intersection between the selected frames and the subshot is measured by mutual information¹. Mutual information reduces the uncertainty of the informativeness of a selected subset (frames) with respect to the remainder of that sequence.

Similarity term refers to measuring statistical relationship between frames (similarity, dissimilarity, or correlation) where each frame is represented by its visual or motion features.

Classifiers are trained to detect frames containing a desired feature such as presence of landmarks, or objects interacted with by the camera wearer. After training, a classifier can label frames as ‘presence of feature vs absence of feature’, while regression predicts the likelihood of the frame containing a predefined feature.

¹A measure of the mutual dependence between two random variables.

- *Summary length*. Having a constraint on the number of selected frames is an additional feature for a generic video summarisation. Therefore, deciding on the length of a summary can be set as a *prior maximal constant* to ensure that the video summary is succinct [60]. Budget-constraint limits the selection based on a user request on the length budget [92] termed a *prior adjustable* in the diagram. Post-processing trims down an excessive keyframe set selected by a generic summarisation method [10, 160]; termed a *posteriori* in the diagram. Lastly, the summary may *stay as extracted* [1, 137, 149, 158, 9, 103, 84, 61, 112, 128, 21, 14, 51].

With an on-line application, the total number of frames will typically not be known beforehand. Deciding on the number of keyframes a priori may not be practical but is often done so as to ensure that the summary is suitable for the human viewer or complies with the the on-line constraints [10].

5.2 Edited Nearest Neighbour Approach for Keyframe Selection

The k-Nearest Neighbours (kNN) classifier is one of the most effective algorithms in data mining and pattern recognition [129, 143]. The classifier typically involves partitioning the data into training set (TR) and testing set (TS), where true labels are known [54]. Representing each element of the data set by its feature vector, the classifier uses the training examples with their true class labels to train. During the testing process, the class label of each element in the testing set is predicted.

kNN has relatively high computational complexity because for each new object to be classified, it has to identify the k nearest neighbours from the (possibly quite large) reference set. The solution is to reduce the data by selecting a lower size of training set which can obtain a similar or higher classification accuracy for the incoming data. This is known by various names in the literature, such as: Instance Selection [72], Prototype Selection or Reduction [131, 167] or Data Editing. Data editing has been a long-standing theme in pattern recognition. Following the two classical methods: Condensed

Nearest Neighbour [64] and Edited Nearest Neighbour [168], a large number of data editing approaches and methods have been proposed and periodically summarised [25, 39, 54, 157, 167].

Using data editing, a subset S is selected from TR , $S \subseteq TR$. Doing that, KNN looks for nearest neighbours only from the selected subset S instead of whole training set TR .

5.2.1 Motivation

Depending on the type, the length of a video may range from less than a minute to several hours, and the shot lengths can vary dramatically within. There is consensus among the researchers that a keyframe-based video summary should be *concise*, *informative*, should *cover* the content of the video, and should be *void of redundancies*. While the interpretation of these categories is domain-specific, they are valid across different video types and applications.

We illustrate the rationale behind our proposal by the following synthetic example. Assume that we have a recorded narrative of a day in a set of 4 events: (1) Met Mary, (2) Looked at the door, (3) Met Mary again, (4) Looked at the door again. The corresponding “video” is shown in Figure 5.2. Each row shows the frames correspond to one event, from left to right.

The standard approach which selects the frame closest to centre of the cluster will pick a frame with Mary (without the hat) for both events 1 and 3, and a frame with the door (without the cat) for both events 2 and 4, as shown in Figure 5.3 – Summary 1. If, however, the user wants to tell the story about their day to a friend, the user will likely pick the frames with the hat and the cat to distinguish events 1 from 3 and 2 from 4 (Summary 2 in Figure 5.3).

Admittedly, a diversity-wise selection method may also be expected to recover the different frames for events 3 and 4. However, we re-position this task as an edited nearest neighbour problem, which will not require manual setting of the balance between diversity and representativeness.

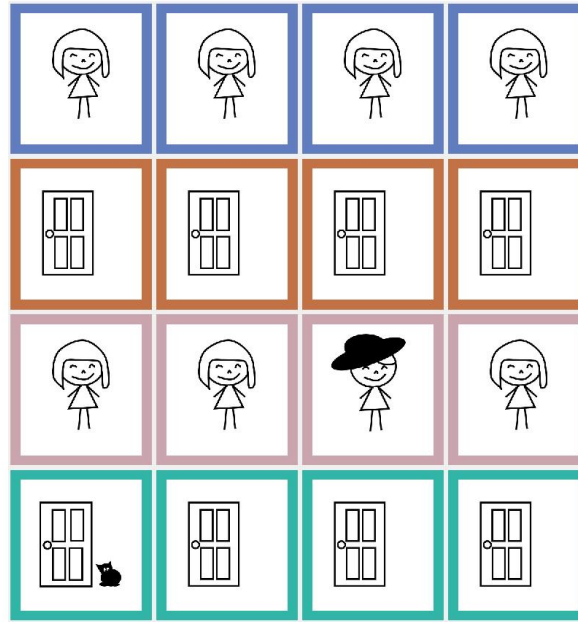
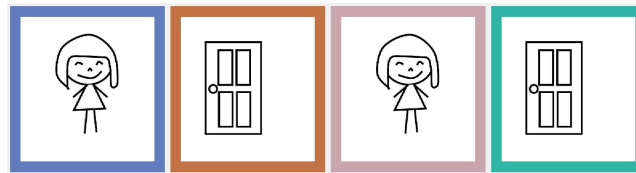
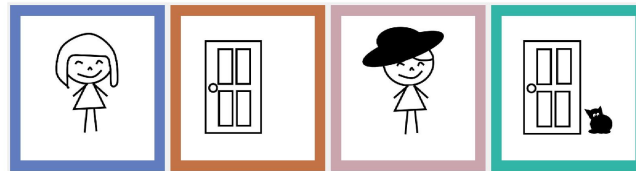


Figure 5.2: Example: A day with 4 events (each row shows an event).



Summary 1 (traditional): Closest to class centroid.



Summary 2 (proposed): Edited nearest neighbour.

Figure 5.3: Two keyframe summaries of the 4 events in the example in Figure 5.2.

5.2.2 Problem Statement

Using any type of video, we assume it has been segmented into units of interest² either manually or by applying any segmentation method. As before, frames are presented as points in an n -dimensional space \mathbb{R}^n . Our approach strives to select the smallest number of keyframes (one frame per unit) which

²units can be events, scenes, or shots.

allows for the best discrimination between units. The discrimination quality between units is defined as the estimated general accuracy of the nearest neighbour classifier (1-nn) using the selected frames as the reference set, where each unit is treated as a class.

While the proposed approach does not explicitly maximise the aesthetic quality [172] or memorability [71] of each image, due to its properties, it can present the story *as a whole*. This makes it potentially suitable for memory reinforcement or video browsing.

Let \mathbf{V} be the video including N frames which are temporally ordered. Our objective is to summarise this video considering coverage and diversity of the contents. It is assumed that the video is segmented into K units. Then the frames are labelled into the K segments, U_1, \dots, U_K , which we will treat as classes. It is assumed that each frame is presented by an n -dimensional feature vector, so that the video is represented as a data set of size $N \times n$ containing N class labels. Using S as the reference set and U_i as the class labels, the objective is to select a subset of frames S , where $S \subset \mathbf{V} = \{f_1, \dots, f_N\}$, to obtain the highest possible resubstitution accuracy.

Our hypothesis is that such a keyframe selection will work well for at least the following reasons:

- This approach ensures that S will contain frames which describe their own classes as accurately as possible (coverage/representativeness/relevance) while accounting for the differences between the classes (diversity).
- The frames are chosen collectively, in relation to one another, which counteracts redundancy, and contributes towards “story telling”.

A brief description of our approach in term of the components introduced in Figure 2.2 is shown in Table 5.1.

Table 5.1: Description of our method in terms of the video summarisation spider diagram in Figure 2.2.

| Property | : Value |
|-----------------------------|---|
| Summary Form | : Keyframes. |
| Frame Representation | : Any. |
| Method of Selection | : Unsupervised. |
| Processing | : Off-line. |
| Summary Type | : Generic Summary. |
| Summary Length | : As prior maximal. |
| Summary Criterion | : Local video structure, and video coherence. |
| Calculation of Objective(s) | : Statistical relationship. |
| Evaluation strategy | : Ground truth annotations. |
| Evaluation metric | : F -measure. |

5.3 Greedy Tabu Selector (One-per-Class)

5.3.1 The Algorithm Details

We initially list the universal pattern-recognition/machine-learning terms used in this chapter:

- *Instance* or *prototype* in video summarisation is defined as a frame,
- *Class*, in here, is defined as a unit or event which contains a sequence of time-contiguous frames which represent similar content or activities. The classes were obtained through segmentation of the video (hence no additional annotation is needed), and
- *Selected subset of prototypes* in this case is the keyframe summary.

The proposed algorithm is detailed as Algorithm 7. The algorithm starts by identifying the instance closest to the class centroid for each class. These c instances are taken together to be the first candidate reference set of prototypes S . This amounts to applying the CC (described in Chapter 4). The set is subsequently modified in the following process.

The nearest neighbour classifier (1-nn) is applied on F using S as the reference set. All classes are declared ‘available’ at the beginning. A ‘privileged’ class is chosen among the available classes as the one with

the worst proportion of correctly labelled instances. It is subsequently made unavailable for the next t iterations, where t is the ‘tabu’ parameter, $0 < t < c$. The prototype for the privileged class, say f^j , is marked for replacement. All remaining instances from class j are taken in turn to replace f^j in S , and the resubstitution error of 1-nn is calculated for each new version of S . Suppose that the reference set with the smallest error was S' , when f^j in S was replaced by f^{j*} . The 1-nn error with S' as the reference set is compared with the error with S . If the new error is smaller, the replacement is made permanent by setting $S \leftarrow S'$. Otherwise, no change is made to S , and the algorithm continues by selecting a new privileged class from the available classes.

Algorithm 7: Greedy Tabu Selector (One-per-Class)

Input: Data set $F = \{f_1, \dots, f_N\} \subset \mathbb{R}^n$ and the corresponding labels into classes $\{1, 2, \dots, c\}$. Tabu parameter, an integer t , $0 < t < c$.

Output: Selected set of prototypes $S \subset F$ with cardinality $|S| = c$, containing one instance from each class.

```

1 for  $i \leftarrow 1, \dots, c$  do
2   Find the centroid of class  $i$  and identify the instance  $f^i$  from this class
   closest to the centroid.
3 Construct the initial set of prototypes:  $S \leftarrow \{f^1, \dots, f^c\}$ .
4 Set all classes as ‘available’.
5 Initialise the minimum-error holder:  $E_{\min} \leftarrow 1$ .
6 Initialise the ‘no-change’ counter:  $w \leftarrow 0$ .
7 while  $w < c$  do
8   Among the ‘available’ classes, find the class with the highest
   proportion of misclassified instances, say class  $j$ .
9   Replace temporarily the current instance  $f^j \in S$  with each of the
   remaining instances from class  $j$ , one at a time. Identify the instance
    $f^{j*}$  which gives the minimum resubstitution error  $E$ .
10  Mark class  $j$  as ‘not-available’ for another  $t$  iterations.
11  if  $E < E_{\min}$  then
12     $E_{\min} \leftarrow E$ .
13    Replace  $f^j$  permanently:  $S \leftarrow S \setminus \{f^j\} \cup \{f^{j*}\}$ .
14     $w \leftarrow 0$ .
15  else
16     $w \leftarrow w + 1$ 
17 Return  $S$ .

```

The stopping condition of the algorithm is implemented as follows. A counter w of steps without changes is initially set to 0. This counter is incremented any time a privileged class is checked but no change to S is made (the ‘else’ statement in lines 15 and 16 in Algorithm 7). The counter is reset to 0 every time a change in S occurs. If there have been c steps without a change, the greedy approach cannot improve any further on the 1-nn resubstitution error, the search is terminated, and S is returned.

Note that, after the first t iterations, the choice will be only among the available $c - t$ classes. Therefore, if we set $t = c - 1$, the classes will be ordered during the first pass through all of them, and checked in this order thereafter.

















The distribution of frames varies amongst different video types, which affects the 1-nn resubstitution error value. For non-egocentric videos, frames are distributed with a simpler structure and more distinguishable event boundaries than the egocentric videos. Therefore, the 1-nn resubstitution error has a lower value for non-egocentric videos compared with the egocentric videos. Further discussion will be demonstrated in section 5.4.2.

5.3.2 Greedy Tabu Selector for the Cartoon Example

Consider applying the Greedy Tabu Selector (GTS) to the example in Figure 5.2. To quantify the frame data, we introduce 4 binary features: (1) Mary present, (2) hat present, (3) door present, and (4) cat present. The labelled data is shown in Table 5.2.

Set $t = c - 1 = 3$. At the initialisation step the Greedy Tabu Selector will pick frames $S = \{1, 5, 9, 14\}$, leading to 50% resubstitution error. The first privileged class will be class 3. After replacing frame 9 with frame 11, the error drops to 43.75%. Class 3 is banned from checking again in the next 3 steps. The next privileged class is 4, and frame 14 is replaced with frame 13, leading to error rate 37.50%. Class 1 and class 2, which are still available are checked next, and no change to S is made. At this step, class 3 becomes available again, and the check reveals that no improvement of the error is achieved.

Table 5.2: Cartoon example data

| Frame | Features | | | | Labels |
|---|----------|-----|-----|-----|--------|
| | (1) | (2) | (3) | (4) | |
| 1.  | 1 | 0 | 0 | 0 | 1 |
| 2.  | 1 | 0 | 0 | 0 | 1 |
| 3.  | 1 | 0 | 0 | 0 | 1 |
| 4.  | 1 | 0 | 0 | 0 | 1 |
| 5.  | 0 | 0 | 1 | 0 | 2 |
| 6.  | 0 | 0 | 1 | 0 | 2 |
| 7.  | 0 | 0 | 1 | 0 | 2 |
| 8.  | 0 | 0 | 1 | 0 | 2 |
| 9.  | 1 | 0 | 0 | 0 | 3 |
| 10.  | 1 | 0 | 0 | 0 | 3 |
| 11.  | 1 | 1 | 0 | 0 | 3 |
| 12.  | 1 | 0 | 0 | 0 | 3 |
| 13.  | 0 | 0 | 1 | 1 | 4 |
| 14.  | 0 | 0 | 1 | 0 | 4 |
| 15.  | 0 | 0 | 1 | 0 | 4 |
| 16.  | 0 | 0 | 1 | 0 | 4 |

Class 4 becomes available next, and again, no improvement is possible. As there have been 4 steps ($w = 4$) with no change to S , the best version is returned: $S = \{1, 5, 11, 13\}$, which corresponds to the desired summary shown in Figure 5.3 (Summary 2).

5.3.3 An Example with Generated Data

We illustrate the GTS performance on a synthetic data set, consisting of three classes in 2-dimensional space. Figure 5.4 shows the scatterplot of a 2D data set labelled in three classes, shown with different markers and colours. The Greedy Tabu Selector was applied to the data set. The migration of the prototypes in the original set (instances closest to the class centroids) is marked by lines. The final prototypes returned by the algorithm are circled. The error rate at the start is 22.28%, and the one at the end, with the selected set of three prototypes, is 17.89%, which demonstrates that

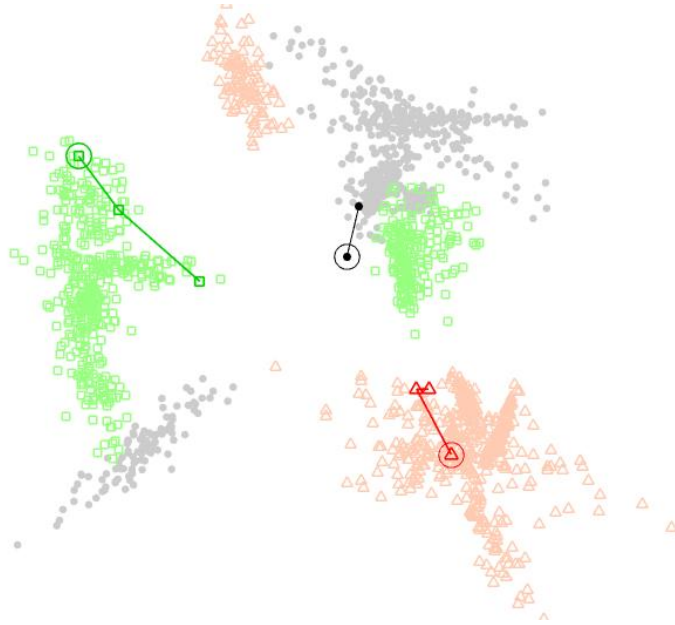


Figure 5.4: An example of 2D data labelled in three classes, shown here with different markers and colours. The migration of the prototypes in the original set is marked by lines. The final set of prototypes selected through the Greedy Tabu Selector algorithm are circled.

substantial improvement on the error can be achieved with a minimal-size set of prototypes obtained through a simple greedy approach.³

5.4 Experimental Evaluation

5.4.1 Feature Representations

While the generated summary produced by the GTS algorithm is independent from the video data representations, still some features may represent the relevant information on frames data better than the others. Therefore, we examined 7 features as summarised in Table 5.3 to compare their effects on the summary.

The low-level features include colour, texture, and shape based features. Among colour based features, we chose RGB colour moments and HSV histogram descriptors. The RGB representations are extracted as described in Chapter 4. For the HSV histogram descriptor, the frame was split again into

³MATLAB code for the GTS and the CC algorithms, as well as the data and code and this example are stored in GitHub: <https://github.com/LucyKuncheva/1-nn-editing> (As of August 2019).

Table 5.3: Feature descriptors

| Level | Information | Notation | Size |
|-------|-------------|----------|---------------|
| Low | colour | RGB | 54 |
| | colour | HSV | 144 |
| | texture | LBP | 59 |
| | shape | HOG | 864 |
| Mid | complex | CNN | 4096 |
| | complex | CNN90 | 84,89, 86, 74 |
| High | semantic | SEM | 1001 |

Note: The number of retained principal components was different for the four videos (described in 4.5.1), as listed in the last column for CNN90.

9 blocks, and a 16-bin colour histogram was computed from the hue value (H) of the HSV colour space.

From the texture type, we used the local binary patterns (LBP) [125], and for the shape type, the histogram oriented gradients (HOG) [38].⁴ The mid-level feature descriptors were calculated as described in Chapter 4.

We subsequently performed PCA on the CNN feature space (VGGNet) and retained the components which preserve at least 90% of the variability of the data in the CNN space. This feature space is denoted as CNN PCA (90%) or just CNN90. Different number of components were retained for each video; these numbers are shown in the last column of Table 5.3.

The last feature space in our collection is semantic labelling (SEM) obtained from the VGGNet classification (soft-max) layer. The output layer of 1000 probability estimates was taken as the feature space, and augmented by one variable to account for people being present in the frame. A non-zero value of this variable means that one of following is detected in the frame: a face, or a human figure.⁵ The value was rescaled to the magnitude of the largest posterior probability among the 1000 CNN outputs.

⁴For both feature spaces we used the respective functions in the MATLAB Computer Vision toolbox.

⁵The detection was done by the respective MATLAB functions included in the Computer Vision toolbox.

5.4.2 The Challenge of Egocentric Video Data

Egocentric videos offer an extra degree of challenge in video summarisation [115]. In this section we demonstrate the reasoning behind using the Greedy Tabu Search algorithm specifically for egocentric video, by comparing the output summaries obtained from three videos of different categories: a video professionally prepared as educational material⁶; a third-person casual video; and an egocentric video.

For the purposes of this illustration, we segmented the video into units of interest, and took only four units (shots, segments, events) from each video.⁷ The videos were as follows:

- Educational material⁸, video #21 which is also called “The Great Web of Water-segment 01”;
- A third-person casual video⁹, “Jumps”; and
- Sub-sampled egocentric video¹⁰, video P01.

Figures 5.5 – 5.7 show the results of applying the Closest-to-Centroid and the GTS method to the three videos. The top plots (subplots (a)) in the three figures show a montage of 10 frames uniformly spaced within each event. Each row corresponds to an event. In addition, the events are colour-coded by the frame borders. The colours are also carried forward in the scatterplots (c) and (d).

Subplot (b) in all three figures contains two 4-frame summaries. One frame has been selected from each event. The top row is the result of the Closest-to-Centroid method, and the bottom row is the result of the proposed GTS method. Note that, for the purposes of this illustration, in both methods we used the simple RGB feature space RGB^{cm} described in detail before in Section 5.4.1.

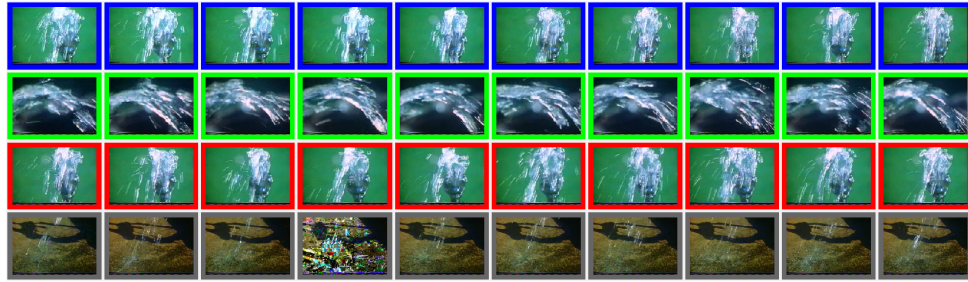
⁶This also refers to traditional videos

⁷We shall term the units of interest ‘events’.

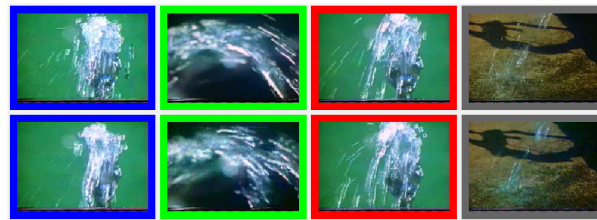
⁸VSUMM [40]:<https://sites.google.com/site/vsummsite/download>

⁹SUMME [60]:<https://people.ee.ethz.ch/~gyglim/vsum/>

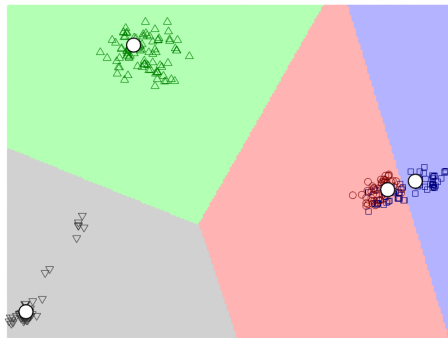
¹⁰UTEgo [91]:<http://vision.cs.utexas.edu/projects/egocentric/>



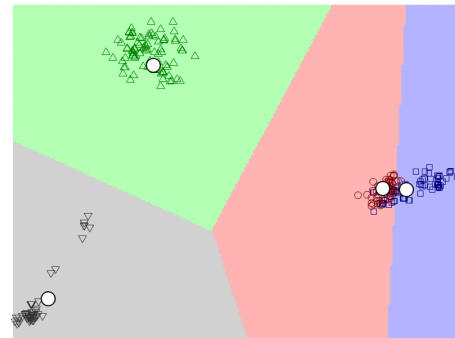
(a) Montage of uniformly spaced frames from the four events (shots in this case).



(b) Summaries of the four events. Top row: closest-to-centroid; bottom row GTS summary.



(c) Classification regions for the close-to-centroid method 1-nn error rate 7.4%

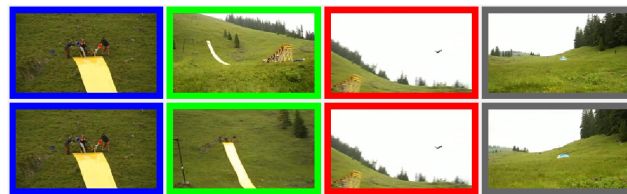


(d) Classification regions for the GTS method 1-nn error rate 4.1%

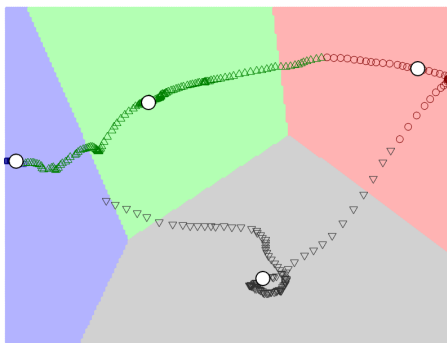
Figure 5.5: Educational video: Keyframe selection through Closest-to-Centroid (CC) and Greedy Tabu Search (GTS) for a part of video #21 from the VSUMM collection, RGB space.



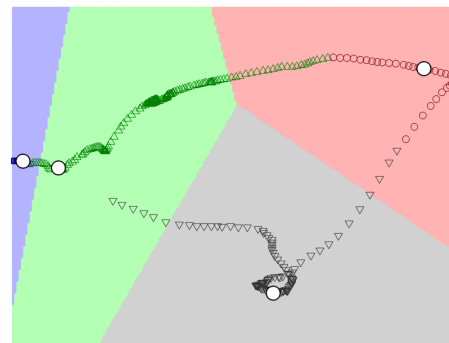
(a) Montage of uniformly spaced frames from the four events (segments in this case).



(b) Summaries of the four events. Top row: close-to-centroid; bottom row GTS summary.



(c) Classification regions for the close-to-centroid method 1-nn error rate 9.3%



(d) Classification regions for the GTS method 1-nn error rate 5.5%

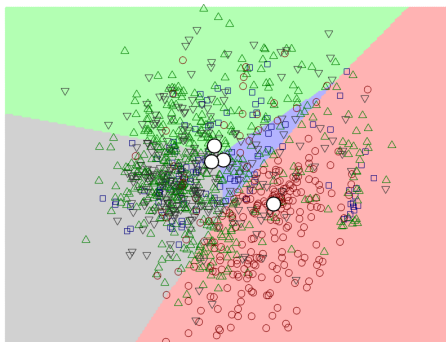
Figure 5.6: Third Person Video: Keyframe selection through Closest-to-Centroid (CC) and Greedy Tabu Search (GTS) for a part of video "Jumps" from the SUMME collection, RGB space.



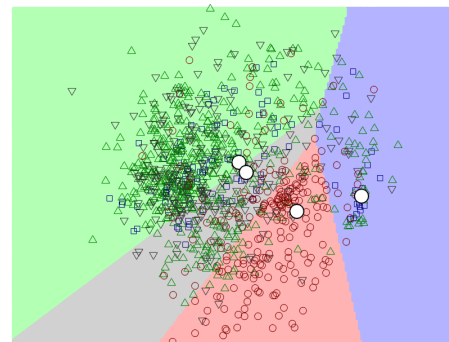
(a) Montage of uniformly spaced frames from the four events (events in this case).



(b) Summaries of the four events. Top row: close-to-centroid; bottom row GTS summary.



(c) Classification regions for the close-to-centroid method 1-nn error rate 55.2%



(d) Classification regions for the GTS method 1-nn error rate 40.1%

Figure 5.7: Egocentric Video: Keyframe selection through Closest-to-Centroid (CC) and Greedy Tabu Search (GTS) for a part of video P01 from the UTE collection, RGB space.

Finally, subplots (c) and (d) give the classification regions for the 4 events (treated as classes) for the two summaries. The scattered points correspond to frames of the video. Different events (classes) are denoted by different marker shapes and colours. The four selected frames are marked with large open-circle markers in each plot. The classification regions are shaded with the colour of the event. They are calculated *only* in the 2d projection space obtained as the first two principal components of the RGB^{cm} space. Shown in the subplot caption are the error rates obtained with the nearest neighbour classifier using the selected 4 frames as the reference set.

The figures demonstrate the dramatic differences between the types of videos. Non-egocentric videos are likely to have a much simpler structure in that the units of interest are represented by visually similar frames, as can be seen in Figures 5.5 and 5.6. The events are clearly distinguishable in all subplots. This is especially visible in the scatterplots (c) and (d). Conversely, these subplots in Figure 5.7 reveal that the classes are highly overlapping. This fact is also supported by a visual inspection of the frame montage for the four events. We can broadly label the events in this figure as: (1) Preparing the kitchen, (2) Cooking, (3) Eating, (4) Washing up. Because of the overlap, the Closest-to-Centroid summary picks similar frames as shown in the top row of subplot (b) in Figure 5.7. Our GTS method manages to ‘disentangle’ the events to some extent, as demonstrated by the differences between the keyframes in the bottom row of the same subplot.

Compare now the differences between (c) and (d) in the three figures. The regions for the egocentric video change the most, suggesting that GTS has a much stronger effect for this type of video. Another indication of the suitability of GTS for egocentric video is the reduction of error rate. The error rates for the educational video and the third-person video were not very large to begin with. This means that many similar frames can be chosen as the summary, and the summary will still be good. For these two types of video, GTS makes a small improvement on the error rate, but the two rival summaries CC and GTS are not really distinguishable. This is not the case for the egocentric video.

The two summaries are indeed different, and the proposed method leads to a more diverse and meaningful summary.

Hence, while many keyframe selection methods may give equivalent results for the first two video types, egocentric videos are significantly more complicated. This explains the abundance of criteria, approaches and methods for summarisation of this video type. More importantly, given the vast differences of possible *good* summaries of the same video, evaluation seems an impossible task. Matching keyframes in an automatic summary to a user summary considered to be a ground truth is hardly applicable to storyboard-type summaries of egocentric videos. As a byproduct of GTS, we have a standalone measure of the merit of a keyframe summary: the classification accuracy achieved by using this summary as the reference set for the nearest neighbour classifier. Lower error will mean that the keyframes are representative of the events they are meant to summarise, and diverse enough to allow for these events to be distinguishable.

5.4.3 Experimental Protocol

Data We chose the UTEgo data set [91] to demonstrate the work of the Greedy Tabu Selector. The data set is challenging because it contains a variety of daily activities with frequent illumination changes, camera view shifts, and motion blur.

Method The proposed Greedy Tabu Selector assumes that the video has already been segmented into units (events). For this experiment, each video was segmented by a subjective opinion. For each video and feature representation, we applied the Greedy Tabu Selector, and calculated the 1-nn resubstitution error. While minimising the error rate is used as a criterion enforcing coverage and diversity, it does not automatically imply high visual quality of the summary or adequate semantic content. We assume that by minimising the error, the obtained summary will be closer to a user-selected summary of the events. Here we rely on the hypothesis that a user would naturally select visually diverse frames, as in our cartoon example in Section 5.3.2. To evaluate this part, we created a user ground truth

summary for each video. To quantify the similarity between the summaries obtained from GTS and GT , we used a well-known measure based on the H-histogram [40], as detailed below. For comparison, we calculated the same values for the CC summary, which we treat as the baseline. An improvement on CC will demonstrate the effectiveness of the edited 1-nn for extracting keyframe summaries.

Matching procedure Our matching procedure is intended to pair two frames *for the same event* with respect to their visual appearance.

Let f_1 and f_2 be the frames being compared. A 16-bin histogram of the hue value is calculated for each frame. The bin counts are normalised so that the sum is 1 for each histogram. Let $B_j = \{b_{j,1}, \dots, b_{j,16}\}$ be the normalised histogram for f_j , $j = 1, 2$. The L_1 distance is calculated by

$$D_H = \sum_{i=1}^{16} |b_{1,i} - b_{2,i}|.$$

The two frames are considered matching if $D_H < \theta_H$, where $\theta_H \in [0, 2]$ is a threshold.

Finally, the F -measure is calculated using the number of matches. As both compared summaries have the same number of frames, the F -measure reduces to the proportion of matching frames.

The value of the F -measure depends on θ_H . The GTS summary itself depends on the tabu parameter t . We experimented with

- $\theta_H \in \{0.2, 0.3, 0.4, 0.5, 0.6\}$, and
- $t \in \{c - 3, c - 2, c - 1\}$, where c is the number of events.

5.4.4 Results

The first visual observation during our experiment was that the CC summaries were already an excellent match to the ground truth, as also reported in Chapter 4. In many cases, inspecting the event in the video together with the

three visually different summaries (user-GT, CC and GTS) leaves doubts as to which of the three summaries represents the event in the best way. Typically, the GTS frames gave a more diverse visual account of the storyline of the video.

Table 5.4 shows the F -values and the classification error (in parentheses) for the 4 videos for $\theta_H = 0.6$, and for the three values of the tabu parameter t . We use the following notations: $F(GTS, GT)$, abbreviated as F_{GTS} is the F -value for the comparison of the GTS summary and the user-GT summary. Similarly, $F(CC, GT)$, abbreviated as F_{CC} is the F -value for CC and the user-GT. E denotes the starting resubstitution error obtained with CC as the reference set, and E_{min} is the resubstitution error with the GTS summary.

Table 5.4: F -values and classification error (in parentheses, both shown in %) for the 4 videos for $\theta_H = 0.6$, and for the three values of the tabu parameter t . The entries in the boxes highlight the cases where GTS is strictly better than CC ($F_{GTS} > F_{CC}$), and the underlined values, the cases where GTS is strictly worse.

Tabu parameter $t = c - 1$

| Feature space | Video P01 (10 events) | | | | Video P02 (12 events) | | | | Video P03 (9 events) | | | | Video P04 (10 events) | | | |
|---------------|-----------------------|------|-----------|-----------|-----------------------|------|-----------|-----------|----------------------|------|-----------|-----------|-----------------------|------|-----------|-----------|
| | F_{CC} | E | F_{GTS} | E_{min} | F_{CC} | E | F_{GTS} | E_{min} | F_{CC} | E | F_{GTS} | E_{min} | F_{CC} | E | F_{GTS} | E_{min} |
| RGB | 40 | (68) | 40 | (54) | 25 | (54) | 25 | (43) | 33 | (79) | <u>67</u> | (58) | <u>40</u> | (66) | 30 | (40) |
| HSV | <u>50</u> | (49) | 30 | (38) | <u>58</u> | (48) | 50 | (38) | <u>78</u> | (62) | 44 | (45) | <u>50</u> | (56) | 30 | (33) |
| LBP | 50 | (66) | <u>60</u> | (51) | 50 | (57) | 50 | (44) | <u>89</u> | (70) | 33 | (50) | 20 | (55) | <u>30</u> | (36) |
| HOG | 20 | (67) | <u>60</u> | (54) | <u>33</u> | (76) | 25 | (54) | 44 | (80) | 44 | (49) | 30 | (65) | <u>40</u> | (40) |
| CNN | 50 | (47) | <u>70</u> | (34) | 42 | (27) | <u>58</u> | (20) | 56 | (64) | <u>78</u> | (29) | 30 | (40) | <u>50</u> | (20) |
| CNN90 | <u>50</u> | (46) | 40 | (30) | 42 | (27) | 42 | (19) | <u>67</u> | (59) | 56 | (29) | 30 | (37) | <u>50</u> | (19) |
| SEM | 40 | (67) | 40 | (50) | <u>42</u> | (56) | 33 | (45) | <u>44</u> | (72) | 33 | (37) | <u>30</u> | (58) | 10 | (48) |

Tabu parameter $t = c - 2$

| Feature space | Video P01 (10 events) | | | | Video P02 (12 events) | | | | Video P03 (9 events) | | | | Video P04 (10 events) | | | |
|---------------|-----------------------|------|-----------|-----------|-----------------------|------|-----------|-----------|----------------------|------|-----------|-----------|-----------------------|------|-----------|-----------|
| | F_{CC} | E | F_{GTS} | E_{min} | F_{CC} | E | F_{GTS} | E_{min} | F_{CC} | E | F_{GTS} | E_{min} | F_{CC} | E | F_{GTS} | E_{min} |
| RGB | <u>40</u> | (68) | 30 | (55) | 25 | (54) | 25 | (43) | 33 | (79) | <u>56</u> | (60) | <u>40</u> | (66) | 30 | (41) |
| HSV | 50 | (49) | 50 | (39) | 58 | (48) | <u>67</u> | (40) | <u>78</u> | (62) | 67 | (50) | <u>50</u> | (56) | 20 | (34) |
| LBP | 50 | (66) | 50 | (54) | 50 | (57) | 50 | (45) | <u>89</u> | (70) | 56 | (54) | 20 | (55) | <u>40</u> | (36) |
| HOG | 20 | (67) | <u>60</u> | (58) | <u>33</u> | (76) | 25 | (69) | 44 | (80) | 44 | (73) | 30 | (65) | 30 | (44) |
| CNN | 50 | (47) | <u>70</u> | (34) | 42 | (27) | <u>58</u> | (20) | 56 | (64) | 56 | (45) | 30 | (40) | <u>50</u> | (21) |
| CNN90 | 50 | (46) | <u>60</u> | (34) | <u>42</u> | (27) | 33 | (20) | 67 | (59) | 67 | (44) | 30 | (37) | <u>50</u> | (19) |
| SEM | 40 | (67) | <u>60</u> | (51) | <u>42</u> | (56) | 25 | (47) | 44 | (72) | 44 | (49) | <u>30</u> | (58) | 10 | (46) |

Tabu parameter $t = c - 3$

| Feature space | Video P01 (10 events) | | | | Video P02 (12 events) | | | | Video P03 (9 events) | | | | Video P04 (10 events) | | | |
|---------------|-----------------------|------|-----------|-----------|-----------------------|------|-----------|-----------|----------------------|------|-----------|-----------|-----------------------|------|-----------|-----------|
| | F_{CC} | E | F_{GTS} | E_{min} | F_{CC} | E | F_{GTS} | E_{min} | F_{CC} | E | F_{GTS} | E_{min} | F_{CC} | E | F_{GTS} | E_{min} |
| RGB | 40 | (68) | <u>50</u> | (55) | 25 | (54) | <u>58</u> | (44) | 33 | (79) | <u>56</u> | (61) | 40 | (66) | 40 | (46) |
| HSV | <u>50</u> | (49) | 40 | (39) | 58 | (48) | <u>67</u> | (40) | 78 | (62) | <u>89</u> | (51) | <u>50</u> | (56) | 20 | (34) |
| LBP | 50 | (66) | 50 | (55) | 50 | (57) | <u>58</u> | (45) | <u>89</u> | (70) | 56 | (58) | 20 | (55) | <u>40</u> | (41) |
| HOG | 20 | (67) | <u>50</u> | (60) | <u>33</u> | (76) | 25 | (69) | 44 | (80) | 44 | (73) | <u>30</u> | (65) | 20 | (45) |
| CNN | 50 | (47) | <u>70</u> | (35) | 42 | (27) | 42 | (22) | 56 | (64) | <u>67</u> | (45) | 30 | (40) | 30 | (24) |
| CNN90 | 50 | (46) | 50 | (36) | 42 | (27) | 42 | (21) | <u>67</u> | (59) | 56 | (44) | <u>30</u> | (37) | 20 | (23) |
| SEM | 40 | (67) | <u>70</u> | (52) | 42 | (56) | 42 | (46) | 44 | (72) | <u>56</u> | (50) | 30 | (58) | 30 | (48) |

Next we examine the effect of parameters θ_H and t . We note that large values of θ_H are more “liberal”, and lead to declaring more matches for the same summaries, which results in higher F -values. For the purpose of supporting our point, we look to demonstrate that the F -value for the GTS summary is larger than the F -value for the CC summary. This will indicate that the GTS summary is closer to the ground truth (GT) chosen by the user. Thus, we calculated

$$\Delta F = F_{GTS} - F_{CC},$$

and note that high positive values of ΔF are desirable.

Figure 5.8 shows ΔF as a function of θ_H for the three values of the tabu parameter t and the 7 feature spaces. Each plot contains the curves for all 7 feature spaces plotted in grey. The curve for the feature space in the title of the plot is shown in black. This allows for an instant comparison of the feature space with the remaining ones. For reference, we plot the 0-line (red) in each plot. If the black curve runs above the 0-line, ΔF is positive, and GTS improves on CC for the respective feature space.

One conclusion from the results so far is that different feature spaces behave differently. It can be observed that HOG, and CNN offer improvements on the baseline for almost all parameter combinations. While CNN and HOG are not affected much by the value of t , RGB and SEM prefer the GTS summaries obtained with tabu parameter $t = c - 3$. The PCA selection and the reduction of the dimensionality does not seem to pay off; the values for CNN90 are lower than those for CNN. The least successful feature spaces in our experiment were LBP and HSV.

To evaluate visually the improvement of GTS over CC for each video, we identified the parameter combination and feature space which lead to the largest ΔF . The results are shown in Figures 5.9–5.12. Each figure contains the three summaries: user-GT, CC and GTS. The matches for CC-GT and

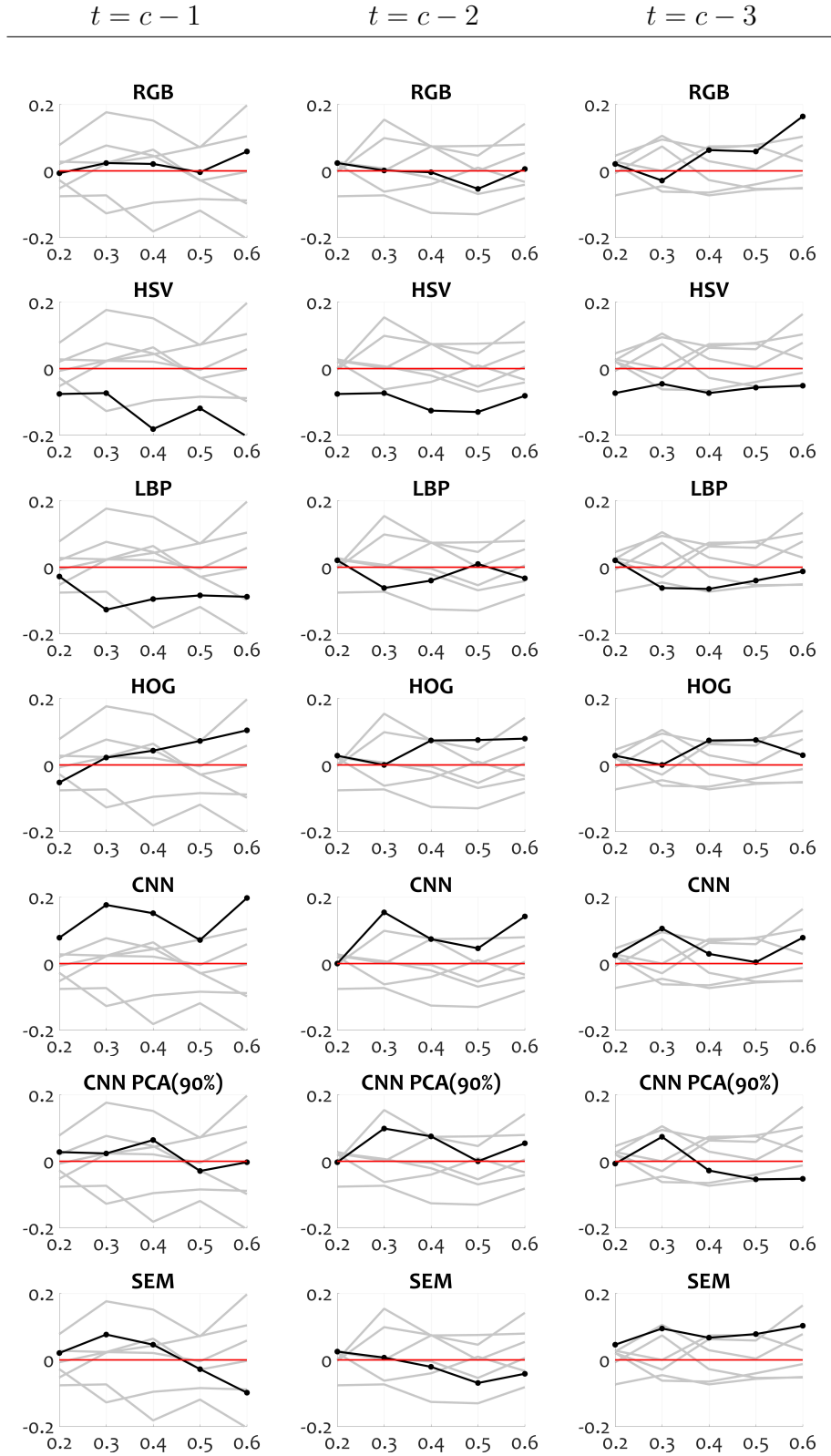
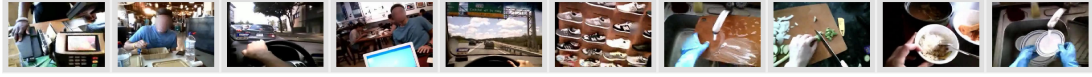


Figure 5.8: Improvement ΔF for the three values of the tabu parameter t and the 7 feature spaces. Each plot contains the curves for all 7 feature spaces plotted in grey. The curve for the feature space in the title of the plot is shown in black. For reference, the zero line is plotted in red.

GTS-GT found by our matching procedure are highlighted by the colour of the rim.¹¹

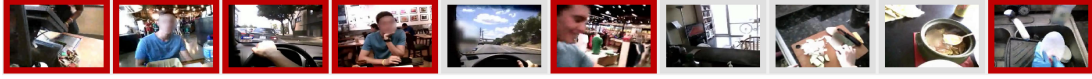
¹¹A full set of figures for $\theta = 0.6$, all videos and all feature spaces is shown in the supplementary material of the respective publication.



(a) Ground truth



(b) Closet-to-Centroid (CC) summary. Matches with GT are highlighted.

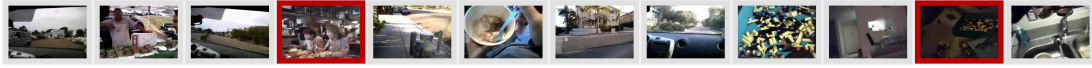


(c) Greedy Tabu Search (GTS) summary. Matches with GT are highlighted.

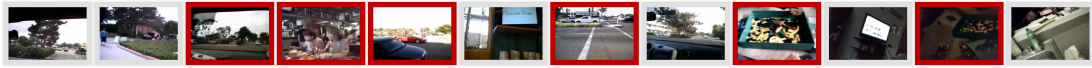
Figure 5.9: *Video P01*. Summaries: GT, CC and GTS with highlighted matches. $\Delta F = 0.40$ for $\theta_H = 0.6$, $t = c - 1$, space HOG.



(a) Ground truth



(b) Closet-to-Centroid (CC) summary. Matches with GT are highlighted.



(c) Greedy Tabu Search (GTS) summary. Matches with GT are highlighted.

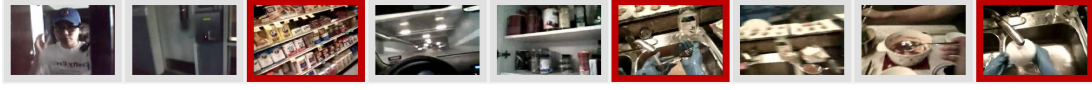
Figure 5.10: *Video P02*. Summaries: GT, CC and GTS with highlighted matches. $\Delta F = 0.33$ for $\theta_H = 0.5$, $t = c - 3$, space RGB.

The figures show that our matching algorithm has flaws. Some matches are missed, and some of the found matches are not convincing. Nonetheless, in the absence of a perfect matching algorithm, or one which the community agrees upon, an imperfect algorithm applied across all feature spaces, videos and parameter choices will have to suffice. Our results are in agreement with the general view that mid and high-level feature spaces (CNN, SEM) lead to better summaries. For these spaces, we were able to improve on CC by applying the proposed GTS method.

Assume that the the F -value is a reasonably faithful estimate of the quality of the GTS summary. It would be reassuring if the resubstitution error rate correlated with F . Table 5.5 shows the correlation between F and E for



(a) Ground truth

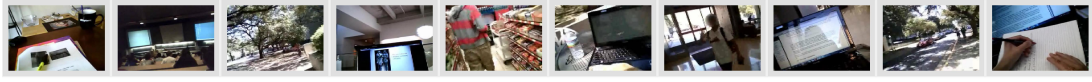


(b) Closet-to-Centroid (CC) summary. Matches with GT are highlighted.

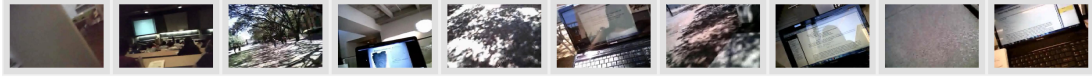


(c) Greedy Tabu Search (GTS) summary. Matches with GT are highlighted.

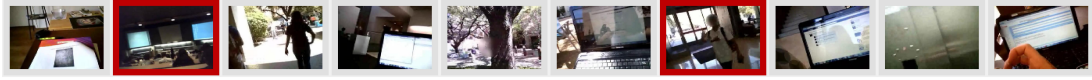
Figure 5.11: Video P03. Summaries: GT, CC and GTS with highlighted matches. $\Delta F = 0.33$ for $\theta_H = 0.6$, $t = c - 1$, space RGB.



(a) Ground truth



(b) Closet-to-Centroid (CC) summary. Matches with GT are highlighted.



(c) Greedy Tabu Search (GTS) summary. Matches with GT are highlighted.

Figure 5.12: Video P04. Summaries: GT, CC and GTS with highlighted matches. $\Delta F = 0.20$ for $\theta_H = 0.2$, $t = c - 1$, space CNN.

the best-scoring feature space in our experiment, CNN. To calculate each coefficient, for each video and each t , we concatenated F_{CC} and F_{GTS} for the 5 values of θ_H for each video, thus obtaining a vector \mathbf{f} with 10 values. The same was done for E and E_{\min} to obtain vector \mathbf{e} . The entries in the table are the Pearson correlation coefficients between 10-element vectors for F and for E .

Table 5.5: Correlation coefficients between F -values and the error rate E for the CNN feature space for the 4 videos and the three tabu parameter values.

| | $t = c - 1$ | $t = c - 2$ | $t = c - 3$ |
|-----|-------------|-------------|-------------|
| P01 | -0.2040 | -0.2040 | -0.1601 |
| P02 | -0.2261 | -0.2261 | 0.1048 |
| P03 | -0.3246 | -0.2010 | -0.1448 |
| P04 | -0.6509 | -0.1843 | -0.3592 |

The negative values in the table (lower error, higher match) supports our overarching hypothesis is that classification error can be linked to the interpretability and usefulness of the summary. GTS has a single tuning parameter, t . In our experiment the results were not significantly different across the values of t which we examined. We propose that for an egocentric video split into 9-12 events, $t = c - 1$ is a good choice, based on the correlation between F and E in Table 5.5.

We note that overtraining, which is a major concern in pattern recognition, is not an issue here. Generalisation accuracy of the edited 1-nn classifier is not a quantity of interest because the aim is to minimise the error on the *training* data, given an extremely limited budget of one frame per event.

5.5 Conclusion

In this study we relate the keyframe selection for video summarisation to prototype (instance) selection for the nearest neighbour classifier (1-nn). Drawing upon this analogy, we propose a Greedy Tabu Selection method for extracting a keyframe summary. It is assumed that the video has already been split into units (segments or events), and each such unit is regarded as a class. Our hypothesis is that better 1-nn classification accuracy of the video using the selected set of keyframes as the reference set (resubstitution accuracy) is linked to a better summary.

We compared 7 feature representations including low level features (colour, texture, shape) and high-level features (people and objects). According to our results, the CNN feature space was consistently better than the alternatives. Applying GTS on the CNN space led to better summaries than the baseline ones, obtained through the Closest-to-Centroid method.

The difficulties in evaluating summaries for egocentric videos come from several sources. First, because of the intrinsic diversity of each event, many selections of representative frames, which may be visually quite different, could be equally good summaries of the video. Thus a comparison

with a single user summary may score low potentially good automatic summaries. Second, the CC baseline is often an excellent summary already, and improvements on that summary may be difficult to rank. This holds in general, not only for the present study. Many times, authors of new video summarisation methods choose a baselines which are not very competitive (random, uniform, mid-event), and still, the results from user studies are less impressive than expected. Perhaps this difficulty in distinguishing between summaries within a narrow margin for improvement, combined with the subjective uncertainty involved in any such evaluation are the reason for the lack of large-scale experimental comparisons of video summarisation methods.

There are several interesting directions for further research. First, with a larger budget (more than one frame allowed for each segment), new, more accurate variants of the GTS can be developed. Second, combination of feature spaces can be explored to find even better summaries. While concatenation of feature spaces is a straightforward solution, classifier ensembles may be more effective. Finally, the error-rate criterion for selecting the frames can be combined with quality-enforcing criteria to boost the aesthetic quality of the summary in addition to diversity and coverage. Last but not least, we remark that a lot of effort in developing new summarisation methods may be fruitless without a standard, widely accepted method for comparing keyframe summaries.

Taking user's interest and preferences into account, further study is required in developing a method to extract multiple summaries from the same stream. This area will be explored in the next chapter.

Chapter 6

Selective Search for Producing Query-Based Summary

6.1 Query-Based Video Summary

Recently, several query-based summarisation methods have been proposed either by a given lexical query [173, 145, 162], or a video query [116]. This review is focused on the former group, and mainly on summarisation methods proposed for egocentric data streams.

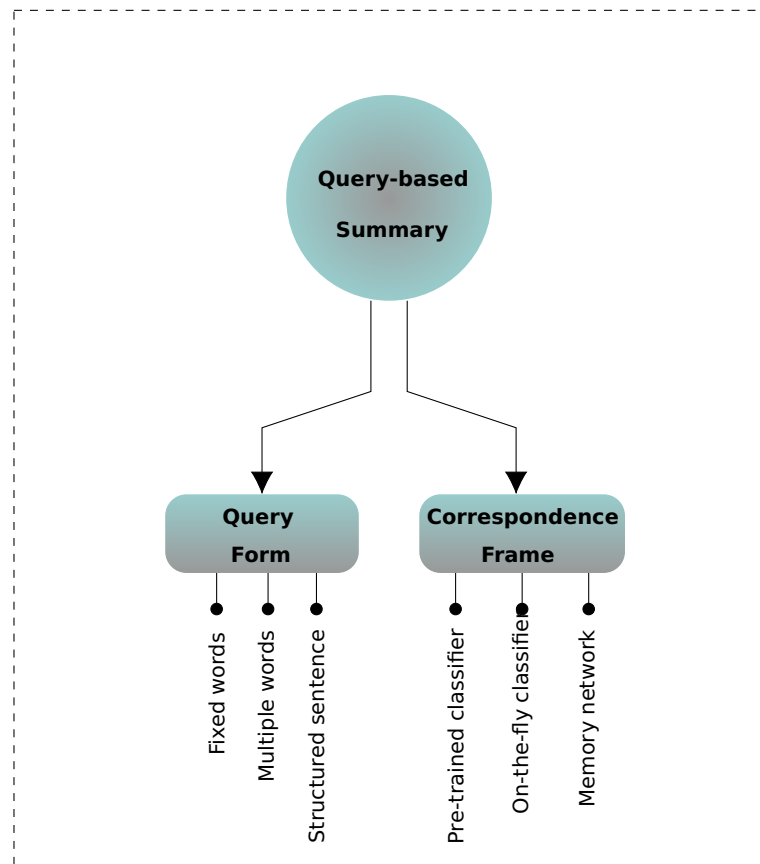


Figure 6.1: A classification of query-based video summarisation methods.

Following the properties introduced in Figure 2.2, here Figure 6.1 shows two additional topics required to describe a query-based video summarisation.

- *Query form.* For each video, user's preference can be given as an input query. The lexical query can be either in the form of structured sentence [173]; or a set of fixed [144, 145] or multiple [162] sequences of words.
- *Correspondence frames.* Video frames associated with the user preference are estimated to filter out from the rest of frames. The criterion for finding related frames can be based on a pre-trained classifier [173], an on-the-fly classifier [162], or a memory network [145].

6.2 Problem Statement

Hitherto, many state-of-the-art approaches were built to summarise egocentric videos [103, 84, 60, 61, 92, 21, 175, 96]. Typically, each of these approaches generates just a single summary for all users. A single generic summary may not suit everyone, given the unconstrained scenarios in most egocentric videos and lifelogging data streams. This form of summary can be suitable in some controlled domains such as video surveillance of a specific area with constant background and predefined salient events. Moreover, available annotated data show considerable discrepancies between summaries made by different users [60]. Users may prefer to obtain a summary related to a specific concept or event. For instance, a user who follows a diet would be interested in a summary of their eating routine during the day. An elderly user may want to extract summary of faces of the people they have met during their day.

Lately, several authors addressed this problem often as a supervised approach of generating a summary built from a user's query [173, 144, 145, 162]. Here we propose a new query-based summarisation approach where we preserve the frame-time relationship in order to answer the question 'when?'.

The study conducted by Le et al. [87] indicates that additional non-visual contextual details and meta-data (e.g. date and time) would improve the practicality of using video summarisation on reinforcing the memory. Therefore, our approach, includes the time tags in constructing the final keyframe summary set. The proposed method can be useful in retrieving memories of daily experiences, behaviours of interest or concern, or in spotting rare occasions when a certain object becomes a part of the view.

We aim to solve the problem of query summarisation as an unsupervised approach which requires no particular set-up or knowledge of domain, and can be used for daily living. A brief description of our approach in term of the components introduced in the diagrams of Figure 2.2, and Figure 3.1 is shown in Table 6.1.

Table 6.1: Description of our method in terms of the video summarisation diagram.

| Property | : Value |
|-----------------------|-------------------------------|
| Summary Form | : Keyframes. |
| Frame Representation | : Mid-level (complex CNN). |
| Method of Selection | : Unsupervised. |
| Processing | : Off-line. |
| Summary Type | : Query-based Summary. |
| Query form | : Fixed word. |
| Correspondence frames | : Any pre-trained classifier. |
| Application domain | : Daily living. |
| Evaluation strategy | : Ground truth annotations. |
| Evaluation metric | : F-measure. |

6.3 Methodology

6.3.1 Description of the Proposed Process

Figure 6.2 illustrates the proposed approach¹, and Figure 6.3 depicts the steps of the implementation algorithm. First, after obtaining the user's query, we identify all frames in the video related to it through semantic concept search. We call these frames 'correspondence frames'. The user's query is given as a

¹MATLAB code is available at: <https://github.com/pariay/Selective-Summary> (As of August 2019).

word (e.g. food, phone, laptop, book). Next, the identified frames are grouped along the timeline to form events.

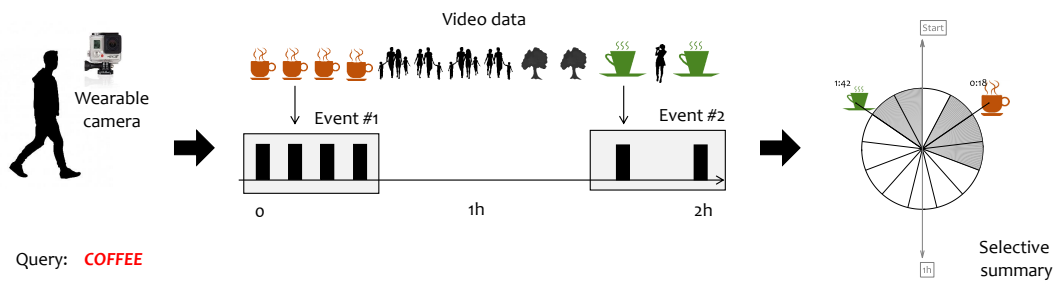


Figure 6.2: Diagram of the proposed method for selective egocentric video summarisation. Shaded sectors of the circle are the events detected through the algorithm.

We apply an algorithm which we call “occurrence-led clustering” to find time intervals which will be the events to summarise. At the next step, we extract keyframes from the events. Finally, we visualise the summary using a new approach, which we term a “compass summary”.

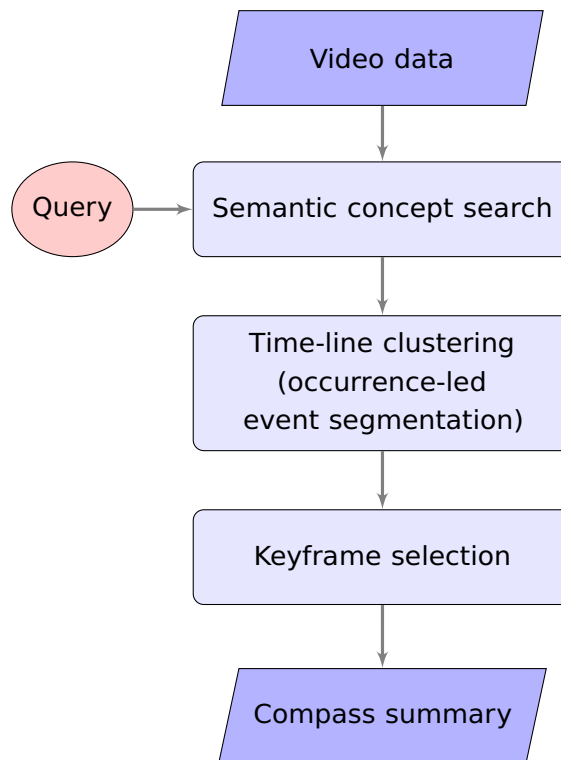


Figure 6.3: Flowchart of the proposed method for selective video summarisation.

6.3.2 Semantic Concept Search

In order to compute the object representation, we propose to use the winner of the ImageNet Large Scale Visual Recognition Competition 2015 (ILSVRC),

Residual Network (ResNet) [66]. As a result, for each frame, the network returns a set of lexical concepts of the detected dominant object along with a prediction score. For example, a dog could be presented as the dominant object in the frame with score 0.2, measuring the certainty that the identified object corresponds to the image content.

Inspired by Dimiccoli et al. [42], we used WordNet [113, 114] to post-process the results and calculate the similarity score between a detected object and the user's query. WordNet is a lexical database which groups English words into a set of synonyms, provides a short definition of the words and shows usage examples. The value for a given frame is calculated as follows. The word representing the dominant object detected by ResNet and the query are entered in WordNet, which then outputs a degree of similarity. This degree varies from 0 for dissimilarity to 1 for identity. We considered the frame to be relevant to the user query if the similarity was equal to 1.

The semantic search algorithm returns a vector representing the presence (Label 1) or absence (Label 0) of the user's query for each frame in the video.

The CNN (ResNet 50) used here has been pre-trained on images with a canonical view and correct level of illumination without any motion blur. These conditions are rarely met in egocentric images. Therefore, we set a threshold of 0.3 on the probability prediction score of the CNN. Frames with dominant objects whose score is less than the threshold are considered to be empty.

Some popular queries have bespoke solutions. An example is 'food'. For a user with an eating disorder problem (overeating or under-eating), it is important to regularly check their dietary routine (by themselves or by a doctor). Being of a great public interest, the problem of detecting food has been addressed in the past as a binary classification problem where the algorithm has to distinguish whether the given image contains food or not [5, 79, 148, 136]. Our approach can make use of such solutions at the semantic search step, bypassing the need to use ResNet and WordNet.

An example of the semantic search step is shown below. Figure 6.4 shows a frame from video P02. ResNet returned description: car mirror. The search query was “automobile”. The similarity score between the tokenised frame description and the query was assessed at value 1 by WordNet. According to our threshold, the frame was given label 1 indicating that it matches the query.

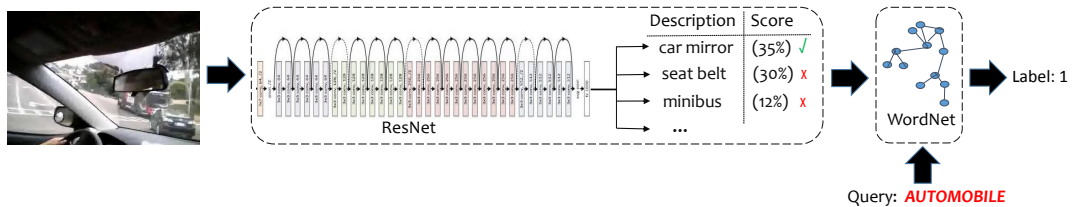


Figure 6.4: Illustration of the semantic search process using a frame from video P02 (UTEgo data set) and query ‘automobile’.

The poor quality (e.g. motion blur, composition, illumination) of the images in egocentric videos often leads to false positive and false negative detections. Two such examples are shown in Figure 6.5. The image in Figure 6.5 (a) is a false positive detection for query ‘television’, and the image in Figure 6.5 (b) is a false negative for query ‘food’. The true dominant objects in these images were respectively ‘car window’ or ‘street’ in Figure 6.5 (a) and ‘food’ in Figure 6.5 (b).



Figure 6.5: Frames from egocentric videos P02 and P01 (UTEgo data set) mislabelled by the semantic labelling algorithm. (a): false positive for ‘television’, and (b): false negative for food.

6.3.3 Occurrence-led Event Segmentation

An Occurrence-led Event Segmentation (OLES) is proposed here as the next step. The term “occurrence-led” is coined by us to denote the process of finding temporal clusters on the time line based on presence-absence

(occurrence). After the frames relevant to the query have been identified, we cluster only their time occurrences (not the frame content or feature representation). For a given concept, we prepare a binary vector with consecutive elements corresponding to the frames in the video. Value 1 indicates that the respective frame contains the concept of interest, and value 0, that it does not. Hierarchical agglomerative clustering was applied to cluster *time-adjacent* frames together based on their geometric centroid.

Consider the toy example in Figure 6.2. The query “coffee” returns the following vector relating the 13 frames with the searched concept:

$$\begin{array}{cccccccccccccc} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ \hline & \text{Event \#1} & & & & & & & & & \text{Event \#2} & & \end{array}$$

The data which we cluster here is the sequence of *occurrences* of the query concept on the time line. We apply the single linkage procedure using the centroid method.

One drawback of nearly any clustering method, including hierarchical clustering, is that the number of clusters is not known in advance. When we cluster a single-dimensional time variable, we have the advantage of being able to interpret the clusters and pose time constraints as deemed necessary. For the video summarisation purposes, we can argue that an event should not be shorter than a given time interval, and that the time gap between events should be no less than a given amount. If two candidate-events are closer to one another than this gap, they are likely parts of the same event. In the toy example, imposing the restriction that the centroids of two clusters must not be closer than 3 frames, the method returns two clusters marked as Events above.

As to the minimum length of an event, we decided not to pose any restrictions. The reason for this are twofold. First, even a glimpse of a certain object may be of high interest. For example, a casual glance at a shelf with wines in the supermarket may need to be flagged in the summary. Second, the camera wearer may not be focusing their gaze on a particular object for a long time

even though they may be interacting with this object. An example of this is a chat on the phone. The user may look at the screen for a moment to verify the caller's identity, and then the phone will be pressed to the user's ear, and out of the camera view. For the gap between events, though, we chose a 20-minute threshold. Given the typical length of the egocentric videos (few hours), and lifelog records, we found that this threshold leads to summaries of reasonable length.

6.3.4 Keyframe Selection

Once the events have been determined through OLES, the next step is to select a good subset of keyframes (one keyframe per event). This step needs a feature representation of all frames. For this representation we chose the 4096 deep features extracted as it is already explained in Chapter 4. Treating the temporal events as "clusters" in the respective 4096-dimensional space, the frame closest to the centroid of the cluster was chosen to represent that cluster.

6.3.5 The Compass Summary Visualisation

We demonstrate the result of our summarisation method using a "compass view" as shown in Figure 6.6. Consider query "phone" in video P01 from the UTEgo data set [91]. The semantic concept search identified 90 frames containing a mobile phone as the dominant object (the actual number of frames related to the "phone" query is 153).

The duration of the video, rounded up to the closest hour, is represented by a circle, and the hours are denoted with annotated long spikes. The individual frames where the query concept is found, are plotted with short black spikes (90 in this case). Shaded sectors of the circle are the events detected through the OLES algorithm. Finally, the spikes with the offset images are the proposed summary. The summary should be read clockwise, starting from the box 'Start' at the top.

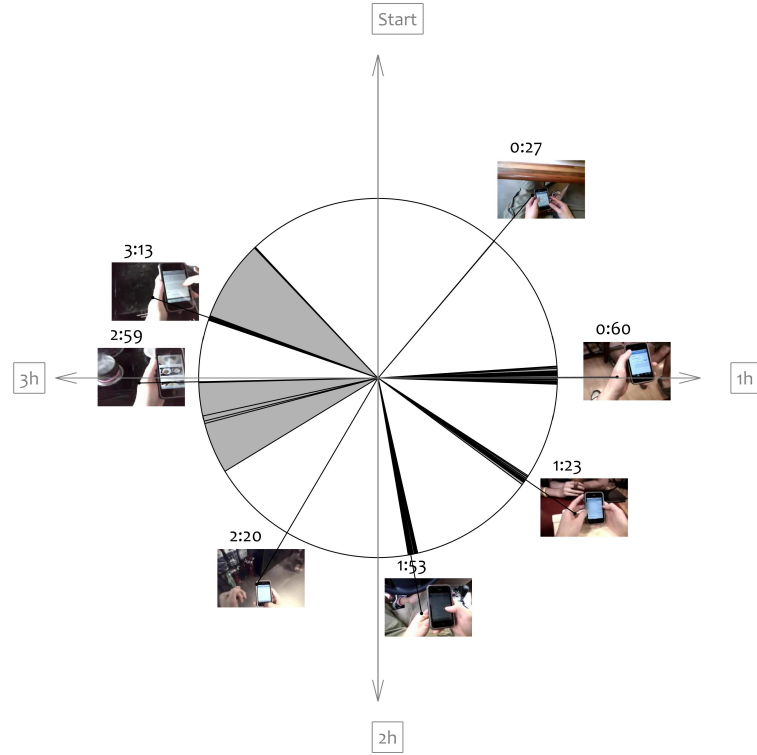


Figure 6.6: An example of a compass summary of the system’s output for video P01 from the UTEgo database for query ‘phone’. Shaded sectors of the circle are the detected events. The individual frames related to the query concept are plotted with short black spikes.

The compass view allows the user to see the whole video at a glance and indicates the time positions of the summary frames.

6.4 Experimental Results

This section presents quantitative experimental results on two egocentric data sets. The aim of the experiment is to demonstrate the effectiveness of the presented selective keyframe summarisation process. In the first leg of this experiment, we assess quantitatively the semantic concept search. This part of the pipeline pre-determines the success of the subsequent clustering and keyframe selection parts (Figure 6.3), dictating to a large extent the quality of the final summary. Next, we estimate the effectiveness of the whole selective summary.

6.4.1 Data Sets

To demonstrate the performance of the approach, two data sets were selected: the University of Texas Egocentric video (UTEgo) [91]; and the Egocentric data set of the University of Barcelona-objects (EDUB-obj) [22]. The given results illustrate that our selective summary approach works on both type of data (egocentric video and lifelog series of images).

Each UTEgo video was sub-sampled as explained in Chapter 4. The EDUB-Obj comprises of 4916 images of daily activities: eating, working, attending meetings and shopping. Images were recorded by 4 different subjects in 8 different days (each of them having captured 2 days). This data set is acquired by the wearable Narrative camera which captures images in a passive way every 30-60 seconds. Number of images per subject are as follows:

- Subject 1-1, 588 images.
- Subject 1-2, 721 images.
- Subject 2-1, 589 images.
- Subject 2-2, 557 images.
- Subject 3-1, 726 images.
- Subject 3-2, 437 images.
- Subject 4-1, 610 images.
- Subject 4-2, 684 images.

We prepared a ground truth by identifying the dominant object for each individual frame for all videos. The most common objects found in both data sets were: car, food, phone, laptop/computer. In addition there were other objects such as: glass, beer, coffee, book, desk, light, sign, refrigerator and television. We are interested in one dominant object per frame, and ignore any other object in that particular frame.

6.4.2 Effectiveness of the Semantic Search Algorithm

For each video, we identified the most represented objects. Then we applied the semantic search, separately for each identified object. To do this, the

frames were labelled with 0 and 1, as described in Section 6.3.2. The result from the semantic search was represented in the same format, which allowed us to calculate Precision, Recall, and the F -measure. For each video we averaged the Precision, Recall, and the F -measure across the query terms. The results are shown in Table 6.2.

Table 6.2: Result of the concept search algorithm for different user queries per video (in %). The Precision, Recall, and the F -measure are averaged across the query terms.

| <i>data set</i> | <i>Name</i> | <i>Precision</i> | <i>Recall</i> | <i>F-measure</i> | <i>Concepts</i> |
|-----------------|-------------|------------------|---------------|------------------|--|
| <i>UTEgo</i> | P01 | 92.2 | 49.2 | 60.4 | food, car, phone, computer, shoe |
| | P02 | 80.4 | 26.2 | 36.6 | food, car, glass, book, television |
| | P03 | 88.7 | 37.5 | 49.5 | food, car, phone, grocery, refrigerator, washbasin |
| | P04 | 100 | 20 | 31.7 | food, laptop, book |
| <i>EDUB</i> | Subject 1-1 | 88.5 | 34.5 | 39.5 | food, car, phone, building |
| | Subject 1-2 | 80.4 | 54.1 | 61.6 | food, car, mobile, beer, coffee, glass, cup, sign |
| | Subject 2-1 | 100 | 55.5 | 67.8 | phone, computer, light, grocery |
| | Subject 2-2 | 83.2 | 37 | 47.8 | food, phone, glass, laptop, light |
| | Subject 3-1 | 87.75 | 40.5 | 46.5 | phone, laptop, book, train |
| | Subject 3-2 | 99.5 | 46 | 58 | food, phone, computer, desk |
| | Subject 4-1 | 100 | 33.3 | 48.7 | computer, desk, building |
| | Subject 4-2 | 94.7 | 24.7 | 44 | car, computer, train |

The table shows that our detection algorithm performs well in finding frames related to the user search (high Precision values), however it also misses a considerable number of frames which are related to the concept (low Recall values). Considering that we are using poor quality images (egocentric video), we regard our semantic search as reasonably successful.

6.4.3 Effectiveness of the Selective Summarisation Method

The aim of this part are: (1) to determine the success of the Occurrence-led Event Segmentation algorithm followed by the keyframe selection; and (2) subsequently to determine the effectiveness of the entire selective summary method.

To this end, we made a user summary GT for each video and each concept: ‘phone’, ‘food’, and ‘car’. The selected frames account for the events when the camera wearer is interacting with the object of interest (one frame per event). An ideal output from our method would match reasonably the number, timing and content of GT . We must note, however, that many frames of different

visual content and at different time moments may represent the same event equally well. Thus, a summary returned by our method may not be an ideal match for GT and still be of high quality.

Table 6.3: Results of the Selective Summary process for different user queries per video (in %).

| Data Set | Name | Selective Summary without Concept Search algorithm | | | Selective Summary method | | |
|----------|-------------|---|--------|-----------|-----------------------------|--------|-----------|
| | | Precision | Recall | F-measure | Precision | Recall | F-measure |
| UTEgo | P01 | 96.6 | 87.6 | 91.4 | 70 | 88.4 | 75.4 |
| | P02 | 78.2 | 100 | 87.4 | 72.6 | 78.4 | 70.8 |
| | P03 | 95.8 | 100 | 97.7 | 86.2 | 90.3 | 86.8 |
| | P04 | 83.3 | 100 | 89 | 85.7 | 100 | 91 |
| EDUB | Subject 1-1 | 85 | 91.75 | 87.5 | 80 | 79.25 | 70 |
| | Subject 1-2 | 81.8 | 93.8 | 84.63 | 54.5 | 72.9 | 58.4 |
| | Subject 2-1 | 75 | 100 | 83.5 | 68.3 | 81.3 | 61.8 |
| | Subject 2-2 | 93.4 | 100 | 96 | 70 | 90 | 72.8 |
| | Subject 3-1 | 92.5 | 93.8 | 92 | 80 | 80 | 70.8 |
| | Subject 3-2 | 74.3 | 100 | 82.5 | 71.8 | 87.5 | 71.8 |
| | Subject 4-1 | 100 | 90.3 | 94.3 | 100 | 73.7 | 83.3 |
| | Subject 4-2 | 75 | 100 | 80 | 48.7 | 100 | 57.7 |

For the first part of the evaluation, for every concept w , we applied OLES and the keyframe selection algorithm to the frames *manually labelled as* w . Thus we bypass the semantic search part and assume an ideal input for the OLES and keyframe selection. The resultant keyframe summaries were compared with those for GT . The left part of Table 6.3 provides the experimental results for this part. This time, the matches were calculated as follows: a keyframe containing the object of interest is considered true positive (TP), if the event it represents is also represented by a keyframe in GT . Frames in GT which were not associated with an event returned by OLES were considered false negative (FN). Finally, a frame representing event which was not included in GT is considered false positive (FP). The values are averaged across the queries.

For the second part, we applied OLES and the keyframe selection algorithm to the frames returned by the semantic concept search. The results are presented in the right part of Table 6.3. As expected, the values are lower than the first part due to the imperfection of the semantic search part of the pipeline.

6.5 Summarisation Examples

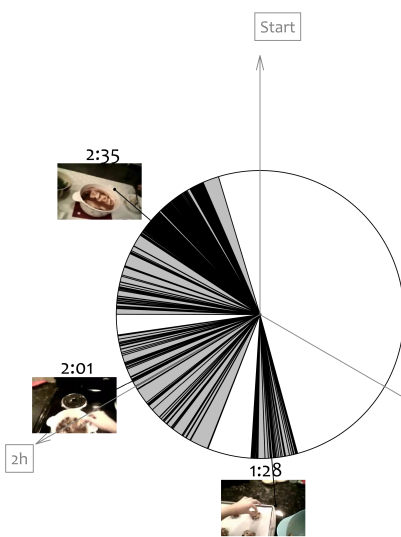
We provide two examples built with the Selective Summarisation method. The results are shown next to the user summary GT , and the summary without the semantic search algorithm.

Figure 6.7 displays an example from the UTEgo video (P03) answering a user’s query on “food”. Our selective summarisation method misses an event 1.5 hours into the video (Figure 6.7 (c)). We note that the frames returned by the closest-to-centroid keyframe selection method in Figure 6.7 (b) are very close to the user selection, both semantically and visually. This indicates that, should we have a better semantic search algorithm, the selective summarisation method may be expected to be accurate and useful.

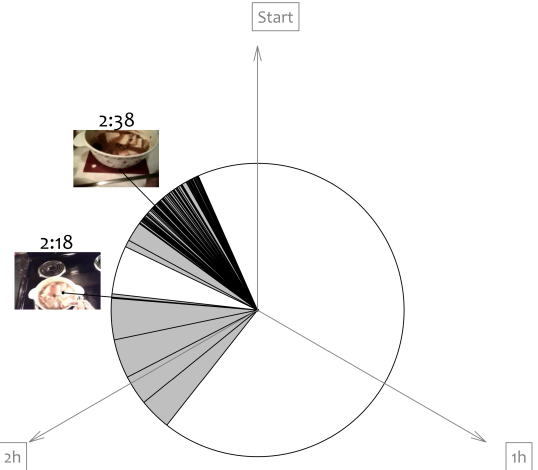
The match counts for this example are as follows: for Figure 6.7 (b): $TP = 3$, $FP = 0$, $FN = 0$, ($F = 100\%$); and for Figure 6.7 (c): $TP = 2$, $FP = 0$, $FN = 1$, ($F = 80\%$).



(a) User keyframe selection (ground truth)



(b) Without Semantic Search



(c) With Semantic Search

Figure 6.7: An example keyframes of the ground truth summary GT and the proposed summary for video P03 of the UTEgo data set. The user’s query is ‘food’.

As a second example, Figure 6.8 shows the results obtained from the EDUB data set (Subject 1-2) answering query on “coffee”. Our selective summary system detected an extra event (Figure 6.8 (c)) at 2.2 hours, due to the lower number of detected frames (low Recall value). Even so, Figure 6.8 (b) still presents a good match with the user selection.

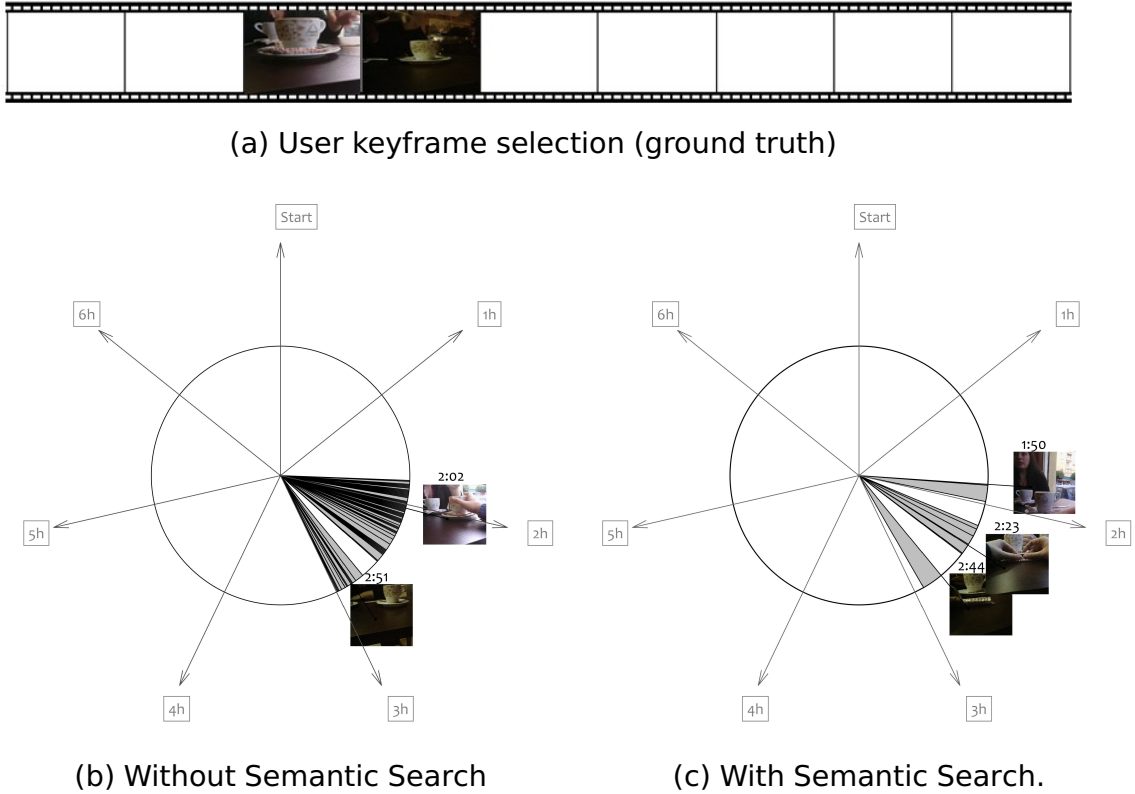


Figure 6.8: An example keyframes of the ground truth summary GT and the proposed summary for Subject 1-2 of the EDUB data set. The user’s query is ‘coffee’.

6.6 Conclusion

We propose a method to extract a selective, time-aware keyframe summary of an egocentric video. The problem was solved by applying a pipeline of a semantic concept search, occurrence-led event segmentation, and finally a cluster centroid keyframe selection. A compass-type diagram was proposed to visualise the selective summary. We demonstrate the effectiveness of our system through experiments with user-defined ground truth and two egocentric video databases.

We found that the major bottleneck of our approach is the semantic search part. Identifying objects and their related concepts is a challenge when the images are blurred, the illumination is poor, and the scene is cluttered. This is the predominant type of images in egocentric video. Thus, the main possibility to improve the accuracy of our selective summarisation system would come from honing the object detection and recognition in egocentric video.

Comparisons with alternative video summarisation methods would not be useful here because we are solving a different problem whereby the summary preserves the time position of the selected frames. We are not aware of other works proposing summarisation methods for this problem.

Future research direction include incorporating user searches on faces and people (known persons or general encounter of groups and crowds). This will involve face detection, people detection and face recognition. We were not able to explore this aspect with the publicly available databases because any faces in the frames were purposely blurred for identity protection. Experiments with own egocentric videos will give us the opportunity to expand the system in this direction.

Combining feature spaces is also an interesting area to explore for a potential improvement on keyframe selection.

A commercially built selective summarisation system may be used for monitoring addictive behaviours, e.g. those related to alcohol, smoking, and overeating.

Further to exploring off-line video summarisation methods, we examined on-line video summarisation as discussed in the next chapter.

Chapter 7

On-Line Video Summarisation

7.1 Motivation

Wearable camcorders provide consumers with the ability to record their daily activities all day long. Having a voluminous and at the same time largely redundant stream of frames makes browsing the videos a disagreeable task. Selecting a summary for such a video on-the-fly would make it possible to keep recording for a long time within the limited resources of the wearable device. Such an approach could be useful in applications including monitoring the daily routines of elderly people [117], memory support [90, 170, 87, 68], and health behavior monitoring such as sedentary behavior [83] or dietary analysis [122].

7.2 Problem Statement

Many of the methods used for generating video summaries typically assume that the full video is available for processing. Here we are interested in on-line summarisation, where keyframes are selected for the summary *before* the entire video has been captured or received. To develop a method fit for egocentric data stream, it is first instructive to understand and assess existing on-line video summarisation methods. We wish to identify the aspects of methods that influence performance and the restrictions inherent in on-line applications.

In this chapter, we initially classify the on-line video summarisation methods by identifying their most relevant descriptive properties. We investigate

nine on-line summarisation methods by specifying them in the terms of the taxonomy, and subsequently apply them to synthetic data with an objectively “best” solution available, and to a collection of real videos. At the end of this experiment, we identify the two methods that perform best in producing an on-line summary.

Following that, we propose a new on-line video summarisation method with robust parameters to a different video type. This method meets the requirements of low computational complexity for feature extraction and summary selection.

Finally, we compare our new method against the two “winner” methods (obtained from the last comparative experiment) by running the experiment on both synthetic and real data sets.

7.3 A Classification of On-line Summarisation Methods

The main properties to describe a video summarisation method are shown in Figure 2.2, Page 13. Figure 7.1 shows the additional topics required to describe an on-line video summarisation. Note all methods contain the same basic components:

- *Set management.* In on-line video summarisation the frames are acquired one by one, as the stream is being processed. We distinguish between two approaches for the keyframe set management: fixed and dynamic. According to the “fixed” approach, once a frame has been included in the summary, it cannot be replaced or removed [1, 9, 51, 112, 128, 137, 158]. Conversely, in the “dynamic” approach, frames may be dropped or replaced [10]. Dynamic management may not be practical in applications where latency is a constraint, and keyframes must be transmitted as soon as they are selected.

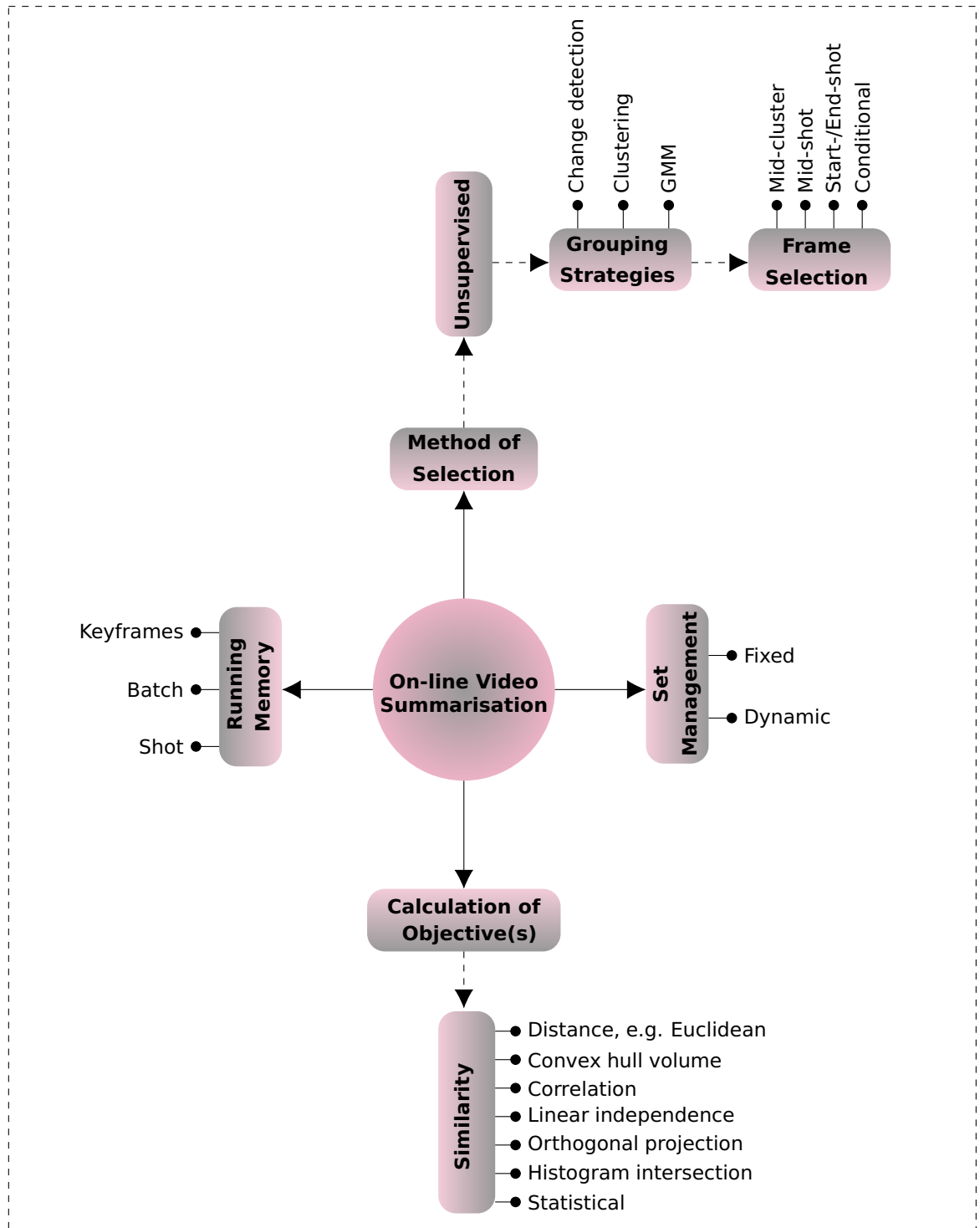


Figure 7.1: A classification of on-line video summarisation methods.

• *Method of selection.* As it is explained in Chapter 2, a concise number or sequences of frames will be selected to represent a video. Methods proposed for on-line video summarisation are mainly unsupervised. Among the unsupervised methods, grouping strategies are the popular choice. We detailed grouping strategies and frame selection in below.

★ *Grouping strategies.* Representative frames are selected from groups of frames, which may or may not be time-contiguous. The groups can be created from the data stream either explicitly, e.g. clustering [10, 51], or implicitly, e.g. change detection [1, 9]. Gaussian mixture models (denoted as GMM) group the frames into a fixed [128] or variable [149] number of Gaussian distributions.

★ *Frame selection.* Particular frames are selected to represent each group. The criterion for selecting a frame can be its location within the cluster; typically the most central frame is chosen [10, 51]. Alternatively, frames can be selected based on their location within a shot, e.g. the first [1] or middle frame [137]. Some methods consider each frame within a group and progressively select keyframes based on some condition, e.g. the difference to existing keyframes [128].

- *Running memory.* Some methods only need to store the current keyframe set [10, 112, 158], whereas others have potentially larger memory requirements such as buffering an entire shot in addition to maintaining the keyframe set [9, 137]. Methods that process frames in batches will need to hold the full batch in memory [1, 51, 149].

- *Calculation of objective(s).* Methods proposed for on-line video summarisation used similarity the most, among the other calculation options introduced in Chapter 2.

★ *Similarity.* How representative a keyframe is can be measured by how similar it is to the frames from which it is selected. To evaluate similarity between frames we can use the feature representation in \mathbb{R}^n and metrics defined on this space. Examples of such metrics are the Euclidean or Cosine distances [51, 128]; the volume of the convex hull of a set of frames [10]; the correlation between two frames [9]; the degree of linear independence between batches of frames [1]; the orthogonal projection of a frame onto the span of existing keyframes [112]; and the intersection of colour histogram bins [137]. Finally, some methods use statistical measures, such as the

likelihood that a frame belongs to a distribution of existing frames, or the equivalence of two sets of frames in terms of mean and variance [149].

7.4 Methods Included in the Comparison Study

We review and compare nine methods for on-line video summarisation. First we give a brief description of each method, and then Table 7.1 categorises the methods according to the classification given in Figure 2.2, and Figure 7.1.

7.4.1 Shot Boundary Detection (SBD)

Abd-Almageed [1] uses change in the rank of the feature space matrix, formed by a sliding window of frames, to identify shot boundaries. The first frame in a shot is selected as a keyframe. The method parameters are the window size and a threshold on the rank for identifying changes.

7.4.2 Zero-mean Normalised Cross-Correlation (ZNCC)

Almeida et al. [9] also look for shot boundaries. They compare the similarities between consecutive frames, using the zero-mean normalised cross correlation as a measure of similarity. Zero-mean normalised cross correlation value is an integer obtained by comparing two grey scale images. Once shots have been identified, a predefined parameter determines whether or not the shot should be included in the summary. Keyframes are selected at uniform intervals throughout a shot. The authors define the desired interval size in terms of the full video length, which typically will not be known in the on-line case. They apply their method in the compressed domain, where it can produce either keyframe sets or skims.

7.4.3 Diversity Promotion (DIV)

The approach taken by Anirudh et al. [10] is to group frames into clusters, while simultaneously maximising the diversity between the clusters. They use the volume of the convex hull of the keyframe set as a measure of

diversity. Incoming frames replace existing keyframes as cluster centres if doing so increases the diversity of the keyframe set. This diversity measure introduces a constraint on the number of keyframes in relation to the feature space size; the number of keyframes must be greater than the feature space dimensionality. The authors recommend the use of PCA to reduce a high-dimensional feature space. However, it is not clear how they calculate the principal components for data in an on-line manner.

7.4.4 Submodular Convex Optimisation (SCX)

Elhamifar and De Paolis Kaluza [51] process frames in batches, and propose a “randomised greedy algorithm for unconstrained submodular optimisation” to select representative frames for each batch. These representatives can be a combination of existing keyframes and new keyframes from within the batch itself. In their experiment on videos they pre-process the data to extract shots and use these as batches. An alternative choice, such as a fixed batch size, will have to be used in a true on-line setting. Similarly, their experiment defines a regularisation parameter in terms of the maximum observed distance between frames; a value that will not be available when running the method on-line.

7.4.5 Minimum Sparse Reconstruction (MSR)

The MSR method [112] uses the orthogonal projection of a frame onto the span of the current keyframe set to calculate the percentage of reconstruction for the frame. A predefined threshold then determines whether the frame is adequately represented by existing keyframes, or it is added to the keyframe set. The use of the orthogonal projection forces a constraint on the number of keyframes used for reconstruction, which is limited to the number of dimensions of the feature space. Once the maximum number of frames is reached, only the keyframes that best represent the others in the set are used to calculate the percentage of reconstruction.

7.4.6 Gaussian Mixture Model (GMM)

Ou et al. [128] use the components of a Gaussian mixture model to define clusters of frames. Each new frame is assigned to the nearest cluster, provided it is sufficiently close to the cluster mean, or otherwise forms a new cluster. The number of clusters is fixed, so any new clusters replace an existing one. Two parameters for the method interact to determine how long clusters are remembered for. This memory affects whether non-contiguous, similar frames are grouped together or not. This method has substantially more parameters to tune than the other methods. The number of clusters, and the initial variance and weight for new clusters must be defined, in addition to the two learning-rate parameters. The authors describe the algorithm as a method for video skimming rather than keyframe selection.

7.4.7 Histogram Intersection (HIST)

Rasheed and Shah [137] propose a multi-pass algorithm that first detects shot-boundaries, and then explores scene dynamics. For the on-line scenario here, we consider just the shot-boundary detection. The detection algorithm uses the intersection of HSV histograms for consecutive frames. An overlap below a pre-defined threshold defines a shot boundary. Once a full shot has been identified, frames from the shot are sequentially added to the keyframe set if they are not sufficiently similar to any existing shot keyframes.

7.4.8 Merged Gaussian Mixture Models (MGMM)

Similar to Ou et al., Song and Wang [149] sequentially update a GMM to describe the distribution of a data stream. However, rather than a fixed number of clusters, their method allows new ones to be added if necessary and also provides a mechanism for combining statistically equivalent clusters.

The MGMM method is for clustering a generic on-line data stream. For a comparison with video summarisation methods, we add an additional step of selecting a representative from each cluster as a keyframe. At each stage of processing, the frame closest to each cluster mean is stored as the current

keyframe. Frames may be replaced if a subsequent frame is closer to the mean. As the cluster means are dynamic, the final set of keyframes may not be the optimal set that would be chosen if the full data set is kept in memory, and the keyframes selected at the end of processing.

7.4.9 Sufficient Content Change (SCC)

The change-detection algorithm from Truong and Venikatesh [158] selects the first frame sufficiently different to the last keyframe as the next keyframe. Unlike the other change-detection algorithms, this method does not require a buffer of all frames that have appeared within a shot so far. Only the keyframe set is stored in memory. The authors describe this algorithm in terms of a generic content change function. Here we implement the algorithm using Euclidean, Minkowski¹ or Cosine distance.

7.5 Control-Charts Method for On-line Video Summarisation

Here we propose a method that uses the statistical process of control-charts to identify shots from a streaming video. Control-charts [146] monitor a quantity of interest to detect when a process moves out of control. The mean, μ , of the quantity is used as a baseline value, and the process deemed to be “in control” while observations remain within a specified limit from the mean, typically three standard deviations, σ .

7.5.1 Control-Charts Method (CCS)

Assuming that each frame is represented as a point in some L -dimensional space, we take the Euclidean distance, d , between consecutive frames as the process to be monitored. A distance $d > \mu + 3\sigma$ defines a shot boundary. Once a full shot has been identified, a keyframe is selected as the frame closest to the centre of the cluster defined by the shot.

¹Minkowski distance between two vectors calculates as the L_p norm of their differences. It is a generalisation of the Manhattan (L_1) and Euclidean (L_2) distances.

Potential issues with such a method are that: (1) consecutive shots identified by the algorithm may be too similar to warrant separate keyframes, and (2) short transitions may be identified as shots, but are not important to the summary. We address these issues as follows:

- We define a measure of similarity between frames, as follows [40]. Use the HSV representation of the frames to obtain 16-bin histograms of the hue value (H). If the Minkowski distance between the normalised histograms is less than a threshold of 0.5, the frames are similar.

Table 7.1: Description of the methods included in the comparisons in terms of the classification in Figure 2.2, and Figure 7.1 (in alphabetical order of the first author).

1 Shot boundary detection (SBD)* Abd-Almageed [1].

| Property | : Value |
|-----------------------------|----------------------------------|
| Summary Form | : Keyframes. |
| Feature Representation | : Low-level (colour histograms). |
| Method of Selection | : Unsupervised. |
| Grouping Strategy | : Change-detection. |
| Frame Selection | : Start-shot. |
| Set Management | : Fixed. |
| Running Memory | : Batch. |
| Calculation of Objective(s) | : Similarity. |
| Similarity | : Linear independence. |
| Summary Length | : As extracted. |

2 Zero-mean normalised cross-correlation (ZNCC) Almeida et al. [9].

| Property | : Value |
|-----------------------------|----------------------------------|
| Summary Form | : Keyframes or skim. |
| Feature Representation | : Low-level (colour histograms). |
| Method of Selection | : Unsupervised. |
| Grouping Strategy | : Change-detection. |
| Frame Selection | : Mid-shot. |
| Set Management | : Fixed. |
| Running Memory | : Shot. |
| Calculation of Objective(s) | : Similarity. |
| Similarity | : Correlation. |
| Summary Length | : As extracted. |

Table 7.1: CONTINUED**3 Diversity promotion (DIV)*** Anirudh et al. [10].

| Property | : Value |
|-----------------------------|---|
| Summary Form | : Keyframes. |
| Feature Representation | : Mid-level (convolutional neural network). |
| Method of Selection | : Unsupervised. |
| Grouping Strategy | : Clustering. |
| Frame Selection | : Mid-cluster. |
| Set Management | : Dynamic. |
| Running Memory | : Keyframes. |
| Calculation of Objective(s) | : Similarity. |
| Similarity | : Convex hull volume. |
| Summary Length | : <i>A priori & a posteriori</i> . |

4 Submodular convex optimisation (SCX)* Elhamifar and De Paolis Kaluza [51].

| Property | : Value |
|-----------------------------|--|
| Summary Form | : Keyframes. |
| Feature Representation | : Mid-level (convolutional neural networks). |
| Method of Selection | : Unsupervised. |
| Grouping Strategy | : Clustering. |
| Frame Selection | : Mid-cluster. |
| Set Management | : Fixed. |
| Running Memory | : Batch. |
| Calculation of Objective(s) | : Similarity. |
| Similarity | : Euclidean distance. |
| Summary Length | : As extracted. |

5 Minimum sparse reconstruction (MSR) Mei et al. [112].

| Property | : Value |
|-----------------------------|--------------------------|
| Summary Form | : Keyframes. |
| Feature Representation | : Low-level (texture). |
| Method of Selection | : Unsupervised. |
| Grouping Strategy | : Clustering. |
| Frame Selection | : Conditional. |
| Set Management | : Fixed. |
| Running Memory | : Keyframes. |
| Calculation of Objective(s) | : Similarity. |
| Similarity | : Orthogonal projection. |
| Summary Length | : As extracted. |

Table 7.1: CONTINUED**6 Gaussian mixture model (GMM)*** Ou et al. [128].

| Property | : Value |
|-----------------------------|--------------------------------|
| Summary Form | : Skim. |
| Feature Representation | : Low-level (Colour - MPEG-7). |
| Method of Selection | : Unsupervised. |
| Grouping Strategy | : Gaussian mixture model. |
| Frame Selection | : Conditional. |
| Set Management | : Fixed. |
| Running Memory | : Keyframes. |
| Calculation of Objective(s) | : Similarity. |
| Similarity | : Euclidean distance. |
| Summary Length | : As extracted. |

7 Histogram intersection (HIST)* Rasheed and Shah [137].

| Property | : Value |
|-----------------------------|----------------------------------|
| Summary Form | : Keyframes. |
| Feature Representation | : Low-level (colour histograms). |
| Method of Selection | : Unsupervised. |
| Grouping Strategy | : Change-detection. |
| Frame Selection | : Mid-shot & conditional. |
| Set Management | : Fixed. |
| Running Memory | : Shot. |
| Calculation of Objective(s) | : Similarity. |
| Similarity | : Histogram intersection. |
| Summary Length | : As extracted |

8 Merged Gaussian mixture models (MGMM)* Song and Wang [149].

| Property | : Value |
|-----------------------------|---------------------------|
| Summary Form | : Keyframes. |
| Feature Representation | : Any. |
| Method of Selection | : Unsupervised. |
| Grouping Strategy | : Gaussian mixture model. |
| Frame Selection | : Mid-cluster. |
| Set Management | : Dynamic. |
| Running Memory | : Batch. |
| Calculation of Objective(s) | : Similarity. |
| Similarity | : Statistical. |
| Summary Length | : As extracted. |

Table 7.1: CONTINUED**9 Sufficient content change (SCC)** Truong and Venkatesh [158].

| Property | : Value |
|-----------------------------|---------------------|
| Summary Form | : Keyframes. |
| Feature Representation | : Any. |
| Method of Selection | : Unsupervised. |
| Grouping Strategy | : Change detection. |
| Frame Selection | : Start-shot. |
| Set Management | : Fixed. |
| Running Memory | : Keyframes. |
| Calculation of Objective(s) | : Similarity. |
| Similarity | : Any. |
| Summary Length | : As extracted. |

* denotes where the method name is our own.

- After identifying a shot and selecting the representative keyframe, we compare this frame with the previous keyframe (if available). If the two consecutive keyframes are similar according to the above measure, we assume that a shot boundary has been falsely identified. The boundary is removed, and the two shots are merged. A new keyframe is selected from the combined shot to replace the two keyframes from the individual shots.
- We define an empirical constant to state the minimum shot length. If a shot contains fewer frames, the shot is ignored and no keyframe is selected.

The CCS method requires three parameters: a pre-defined threshold θ for classifying keyframes as similar, a minimum shot length ms , and initial buffer size B for calculating the starting mean and standard deviation. If we assume that the number of frames per second will be constant across videos, and that the duration required for a shot to be of interest is largely independent of video content, the optimal value for ms should be consistent across videos. We select two seconds to be the minimum duration of a shot for it to be of interest. The full control-chart method is given in Algorithm 8.

Algorithm 8: On-line control-charts method

Input: Data stream $F = \{f_1, \dots, f_N\}$, $f_i \in \mathbb{R}^L$, minimum shot length ms , initial buffer size B , threshold for keyframe similarity θ .

Output: Selected set of keyframes $P \subset F$

```
// Initialisation
1  $P \leftarrow \emptyset$ 
2  $j \leftarrow 1$  // Shot number
3  $S_j \leftarrow \{f_1, \dots, f_B\}$  // First shot
4 for  $i \leftarrow \{2, \dots, B\}$  do
5    $d_i \leftarrow d(f_i, f_{i-1})$  // Euclidean distance
6    $\mu \leftarrow \text{mean}(d_2, \dots, d_B)$ 
7    $\sigma \leftarrow \text{std}(d_2, \dots, d_B)$ 

// Process video frame-by-frame
8 for  $i \leftarrow \{B+1, \dots, N\}$  do
9    $d_i \leftarrow d(f_i, f_{i-1})$ 
10  if  $d_i < \mu + 3\sigma$  then
11    // No new shot detected
12     $[\mu, \sigma] \leftarrow \text{update } \mu \text{ \& } \sigma \text{ with } d_i$ 
13     $S_j \leftarrow S_j \cup F(i)$ 
14  else
15    // New shot detected
16    if  $|S_j| > ms$  then
17      // Shot is sufficiently long
18       $p_j \leftarrow \text{Select-Keyframe}(S_j)$ 
19       $\delta \leftarrow \text{Keyframe-Diff}(p_j, p_{j-1})$ 
20      if  $\delta < \theta$  then
21        // Shots are too similar: Merge
22         $S_j \leftarrow S_{j-1} \cup S_j$ 
23        // Remove last keyframe from set
24         $P \leftarrow P(1 : \text{end} - 1)$ 
25         $p_j \leftarrow \text{Select-Keyframe}(S_j)$ 
26         $P \leftarrow P \cup p_j$ 
27         $j \leftarrow j + 1$ 
28      else
29        // Shot too short: Ignore
30         $S_j \leftarrow \emptyset$ 

25
26 Function  $f = \text{Select-Keyframe}(Y)$ 
27 // Select the frame closest to the mean
28  $f \leftarrow \underset{x \in Y}{\text{argmin}} d(x, \bar{Y})$ 
29
30 Function  $\delta = \text{Keyframe-Diff}(f_1, f_2)$ 
31 // Compare 16-bin Hue histograms of frames  $f_1$  and  $f_2$ 
32  $h_i = \text{Hist16}(\text{Hue}(f_i))$  // Normalised 16-bin Hue histogram
33  $\delta = \sum_{j=1}^{16} |h_1(j) - h_2(j)|$ 
```

7.5.2 Feature Representation

For an on-line application, two factors must be considered when choosing a descriptor: (1) the ability of the chosen feature space to identify the meaningful attributes of the scene; (2) the computational cost of processing (the extraction process, and algorithm running time associated with the feature dimensionality).

Our control-charts method may be used with any descriptor. Preferably, we choose a descriptor which has a low dimensional features with a shorter extraction time (detailed in Table 7.7). The RGB moments is selected as the descriptor to use in the CCS method.

Further, to select a suitable descriptor for testing the algorithm, we implement the extraction of a number of different features, including those used by existing on-line summarisation methods. We experimentally compare these features on the CCS method to find a suitable descriptor. The descriptors are listed as follows:

1. *RGB moments*. The RGB colour moments are obtained as it is already described in Chapter 4.
2. *Colour Layout (MPEG7)* [82]. An input RGB image is uniformly divided into 8×8 blocks. The average value of the pixel colours for each block is calculated. The average RGB colours is converted into YCbCr colour space and then quantized into three sets of 64 DCT coefficients (total of 192 features).
3. *CENTRIST descriptor*. CENSus TRAnsform hISTogram (CENTRIST) [171]. Census Transform compares the intensity value of a pixel with its eight neighboring pixels. The binary results from the 8 comparisons are transformed in a decimal number between 0 and 255. A histogram of these numbers is then generated with 256 bins, one for each Census intensity. The two end bins (corresponding to 0 and 255) are removed, leaving a 254-dimensional feature space. We used a MATLAB implementation to extract the descriptor [18].

4. *HSV histograms*. The feature space is extracted by a quantisation of the HSV color space into a 256-dimensional histogram vector of 32 bins for Hue, 4 bins for Saturation and 2 bins for Value. To increase speed the original image is resized to 1/64th of its original size.
5. *GIST* [126]. This descriptor is computed by convolving an image with 32 Gabor filter (4 scales and 8 orientations), producing 32 feature maps. Each feature map is divided into 4×4 regions and the average feature values calculated for each region. The 16 average values of 32 feature maps are concatenated resulting 512-dimensional descriptor.
6. and 7. *Places205-AlexNet and VGGNet*. We included two mid-level feature descriptors extracted through deep learning neural networks. The 4096 deep features are extracted right before the classification (soft-max) layer of two pre-trained CNNs, known as VGGNet architecture [147] and Places205-AlexNet model [182], using Caffe deep learning toolbox [74].

7.6 Experiments on Comparing Nine On-line Methods

7.6.1 Data

We test each of the nine methods on two synthetic data sets, and subsequently illustrate their performance on the 50 real videos from the VSUMM collection [40].

The first data set reproduces the example of Elhamifar et al. [52]. The data consists of three clusters in 2-dimensional space as illustrated in Figure 7.2. Each point represents a frame in the video. The three clusters come in succession but the points within each cluster are generated independently from a standard normal distribution. The order of the points in the stream is indicated by a line joining every pair of consecutive points. The time tag is represented as the grey intensity. Earlier points are plotted with a lighter shade. The “ideal” selected set is shown with red target markers.

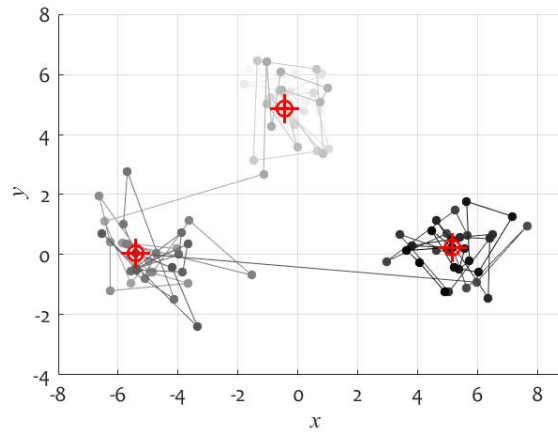


Figure 7.2: Synthetic Data set#1. The time tag is represented as the grey intensity. Earlier points are plotted with a lighter shade. The “ideal” selected set is shown with red target markers.

The second synthetic data set, shown in Figure 7.3, follows a similar pattern but the clusters are less well-defined, they have different cardinalities, and the features have non-zero covariance. Data set #2 is also larger, containing 250 points, compared to 90 in Data set #1. The difference in cluster size and total number of points between the two data sets will guard against over-fitting of parameters that may be sensitive to shot and video length.

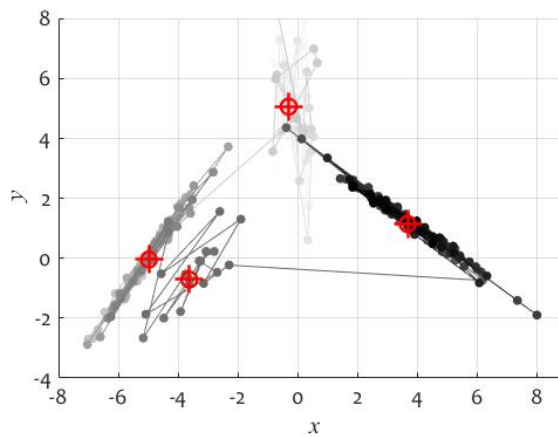


Figure 7.3: Synthetic Data set#2. The time tag is represented as the grey intensity. Earlier points are plotted with a lighter shade. The “ideal” selected set is shown with red target markers.

For both data sets we add two dimensions of random noise (from the distribution $\mathcal{N}(0, 0.5)$). A higher-dimensional feature space is used so that the MSR method is not penalised by being constrained to a maximum of two

keyframes for reconstruction. The additional dimensions and noise also make the synthetic examples a more realistic test for the methods.

Finally, we use the 50 videos from the VSUMM collection, and five ground truth summaries for each video. Since the choice of feature representation may have serendipitous effect on some methods, we experiment with two basic colour descriptors: the HSV histogram and the RGB moments. These two spaces are chosen in view of the on-line desiderata. HSV histograms and RGB colour moments are among the most computationally inexpensive and, at the same time, the most widely used spaces. For the HSV histogram, each frame is divided uniformly into a 2-by-2 grid of blocks (sub-images). For each of the four resulting blocks we calculate a histogram using eight bins for hue (H), and two bins each for saturation (S) and value (V). The RGB colour descriptor is computed as explained in Section 7.5.2.

For the four methods (DIV, SCX, MSR, and GMM) developed using a specific feature space, other than colour histograms, we extract the original features (CNN, Centrist, MPEG7 colour layout) for the VSUMM collection. These original features are used to test whether using an alternative feature space leads to an unfair representation of the performance of a method.

7.6.2 Evaluation Metrics

The aim of video summarisation is to produce a comprehensive representation of the video content, in as few frames as possible. If the video is segmented into units (events, shots, scenes, etc.), the frames must allow for distinguishing between the units with the highest possible accuracy (Chapter 5). Therefore we use three complementary *objective* measures of the quality of the summary:

$$\text{Cardinality} : K = |P| \quad (7.1)$$

$$\text{Approximation error} : J = \sum_{i=1}^N d(\mathbf{f}_i, \mathbf{p}_i^*) \quad (7.2)$$

$$\text{Accuracy} : A = 1\text{-nn}(P) \quad (7.3)$$

where $F = \langle \mathbf{f}_1, \dots, \mathbf{f}_N \rangle$ is the sequence of video frames, N is the total number of frames in the video, $P = \{\mathbf{p}_1, \dots, \mathbf{p}_K\}$ is the selected set of keyframes, \mathbf{p}_i^* is the keyframe closest to frame \mathbf{f}_i , d is the Euclidean distance, and $1\text{-nn}(P)$ is the resubstitution classification accuracy in classifying F using P as the reference set. To obtain a good summary, we strive to maximise A while minimising J and K .

For the tests on synthetic data, we can evaluate the results of the summaries against the distributions used to generate the data. However, we acknowledge that what constitutes an adequate summary for a video is largely subjective. If user-derived ground-truth is available for a video, one possible way to validate an automatic summary is to compare it with the ground truth. The match between the summaries obtained through the nine examined on-line methods and the ground truth is evaluated using the approach proposed by Avila et al. [40]. According to this approach, an F -measure is calculated (large values are preferable) using 16-bin histograms of the hue value of the two compared summaries (Chapter 3).

7.6.3 Experimental Protocol

We first tune parameters by training each method on the synthetic Data set #1. Table 7.2 shows the parameters and their ranges for the nine methods.

Some methods have a parameter that defines the number of frames in a batch. For these methods, we define an upper limit of the batch size to represent the inherent on-line constraints of memory and processing. This

Table 7.2: Parameters for the nine methods tested, the ranges used for tuning the methods to synthetic Data set #1, and the parameter value that generates the best result. *The number of keyframes in the representative set is limited to the feature space dimensionality.

| Method | Parameter | Range | Optimum |
|--------|---|---------------------------------|---------|
| SBD | Batch size (N) | 5 - 30 | 14 |
| | Change threshold (e) | 0.1 - 0.5 | 0.18 |
| ZNCC | Change threshold (e) | 0.01 - 0.5 | 0.13 |
| | Minimum segment length - % (ms) | 0.1 - 10 | 1.2 |
| DIV | # keyframes (K) | 3 - 10 | 3 |
| | Regularisation (λ) | 8 - 12 | 10 |
| | Error to diversity weighting (τ) | 0.2 - 1 | 0.6 |
| | Probability of random update (p) | 0.5 - 3 | 2 |
| SCX | Batch size (N) | 5 - 30 | 25 |
| | Regularisation (λ) | 0.6 - 2 | 1 |
| MSR | Representation threshold - % (e) | 0.3 - 0.9 | 0.3 |
| | # representative keyframes (K) | 4* | 4 |
| GMM | Number of clusters (C) | 3 - 10 | 8 |
| | Learning rate (α) | 0.003 - 0.005 | 0.004 |
| | Selection threshold (e) | 0.1 - 0.5 | 0.2 |
| | Initial cluster variance (σ_0^2) | 2 - 5 | 3.5 |
| | Initial cluster weight (w_0) | 0.05 - 0.5 | 0.1 |
| HIST | Change threshold (e_c) | 0.05 - 1 | 0.05 |
| | Selection threshold (e_s) | 0.05 - 1 | 0.8 |
| MGMM | Batch size (N) | 5 - 30 | 30 |
| | Significance level for match (ρ) | 0.01 - 0.5 | 0.01 |
| SCC | Change threshold (e) | 0.1 - 800 | 1.1 |
| | Distance function (fn) | Euclidean, Cosine, or Minkowski | Cosine |

limit ensures that tuning the batch size does not cause it to increase to an essentially off-line, full data set implementation.

We extract the Pareto sets for the three criteria described in Section 7.6.2, and sort them in decreasing order of accuracy, A . Results with equal accuracy are arranged by increasing values of K (smaller sets are preferable), and then, if necessary, by increasing values of J (sets with lower approximation error are preferable). As A and J achieve their optimal values by including all frames as keyframes, we discount solutions that select more than ten keyframes. An example of the results of training the SCX method on Data set #1 is shown in Table 7.3.

Table 7.3: The Pareto sets for the SCX method trained on Data set #1, describing the optimal combinations of accuracy, cardinality of the keyframe set, and approximation error. The parameter values that generated the results are also shown.

| Batch size | Regularisation | Approximation | | |
|------------|----------------|---------------|-------------|-------|
| | | Accuracy | Cardinality | error |
| 25 | 1 | 1 | 3 | 157 |
| 20 | 0.8 | 1 | 4 | 154 |
| 15 | 1 | 1 | 5 | 135 |
| 10 | 0.6 | 1 | 7 | 126 |
| 5 | 0.8 | 1 | 8 | 121 |
| 5 | 0.6 | 1 | 10 | 114 |
| 15 | 0.6 | 0.99 | 6 | 132 |
| 25 | 1.6 | 0.67 | 2 | 339 |

To assess the robustness of the method parameters across different data samples, the best parameters for each method, as trained on Data set #1, are used to produce summaries for an additional 40 randomly generated data sets: 20 samples following the same cluster size and distributions as Data set #1 (Figure 7.2), and 20 samples following the cluster distributions of Data set #2 (Figure 7.3). We can think of the first 20 samples as “training”, and the latter 20 samples as “testing”, and place more value on the testing performance.

For all 40 data sets, the results for the methods are ranked one to nine; a lower rank indicates a better result. Tied results share the ranks that would have been assigned without the tie. For example, if there is a tie between the top two methods, they both receive rank 1.5.

We next illustrate the work of the algorithms on real videos separately on the HSV and the RGB feature spaces described in Section 5.4.1. We tune the parameters of each method on Video #21 of the VSUMM database. The ranges described in Table 7.2 are used for parameters that are independent of the feature space and number of data points. Ranges for parameters that are sensitive to the magnitude and cardinality of the data are adjusted appropriately. The parameter combination taken forward is the one that maximises the average F -measure obtained from comparing the summary from the method and the five ground-truth summaries. We then select the more successful of the two feature spaces and use the optimal parameter set

for each algorithm to generate summaries for the full set of VSUMM videos. The F -measures are calculated for the comparisons of each video, method and ground-truth summary, and the average for each method compared.

Finally, we repeat the training and testing on the VSUMM database using the original features used by the methods, where applicable. As methods may have been developed and tuned to use a specific feature space, this procedure ensures that methods are not disadvantaged by using the colour-based features.

7.6.4 Results

The relative performance of the methods on the synthetic data sets is shown in Figure 7.4. The merging Gaussian mixture model method consistently generates one of the best summaries. While MGMM still performs relatively well on Data set #2 examples, it suffers from some over-fitting of its batch-size parameter on Data set #1.

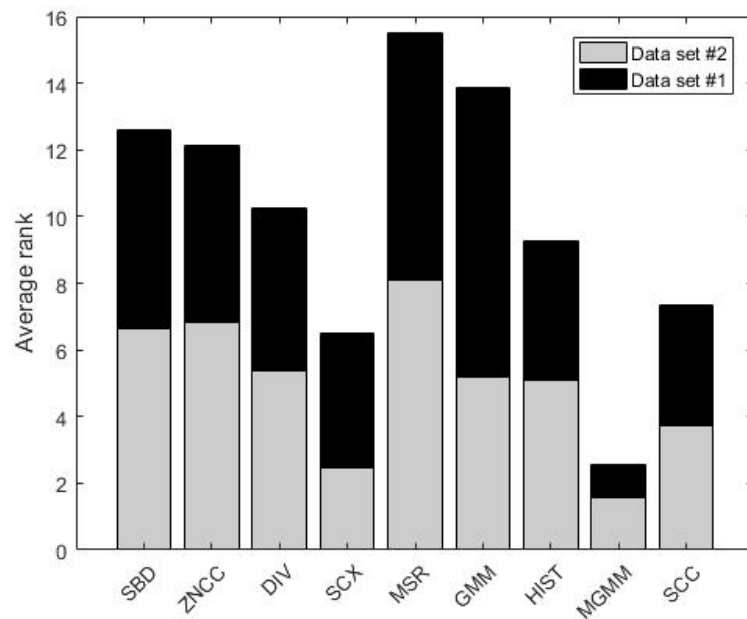


Figure 7.4: Average rank for each method for summaries of 40 randomly generated Data sets (20 each following the cluster distributions of Data sets #1 and #2). On each data set, summaries from all methods are compared and ranked. Better methods receive lower rank.

The SCX and SCC methods also perform relatively well, and are reasonably robust across changes in the data distribution. This robustness is

demonstrated by the relative sizes of the grey and black parts of the bar for these methods; the SCX method receives better ranks on Data set #2 than on Data set #1, and the SCC method performs equally well across the two data sets.

The relatively poor performance of the GMM method may be due to the fact that this algorithm is designed to generate video skims, and therefore tends to return a higher number of keyframes than other methods. The MSR method is potentially affected by constraints from the low feature space dimensionality.

The comparison of the two features spaces on VSUMM video #21 is shown in Figure 7.5 and Table 7.4. Sensitivity to the respective feature space can be observed both in terms of the optimal parameter values found (Table 7.4), and the quality of the match to the ground-truth summaries (Figure 7.5):

Table 7.4: Method parameters tuned on VSUMM Video #21 using HSV histogram and RGB moments to represent frames.

| Method | Parameter | HSV | RGB |
|--------|--------------|-----------|-----------|
| SBD | N | 20 | 19 |
| | e | 0.14 | 0.13 |
| ZNCC | e | 0.01 | 0.05 |
| | ms | 0.1 | 0.5 |
| DIV | K | 6 | 15 |
| | λ | 11 | 9 |
| | τ | 0.2 | 0.6 |
| | p | 1 | 1.5 |
| SCX | N | 80 | 100 |
| | λ | 1.4 | 2 |
| MSR | e | 0.56 | 0.78 |
| | K | 10 | 4 |
| GMM | C | 10 | 9 |
| | α | 0.003 | 0.003 |
| | e | 0.5 | 0.5 |
| | σ_0^2 | 2 | 2.5 |
| | w_0 | 0.05 | 0.05 |
| HIST | e_c | 0.1 | 0.8 |
| | e_s | 0.2 | 0.1 |
| MGMM | N | 200 | 170 |
| | ρ | 0.1 | 0.1 |
| SCC | e | 6 | 516 |
| | fn | Minkowski | Euclidean |

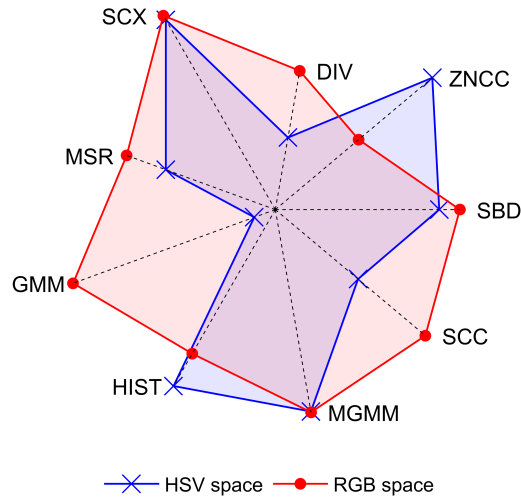


Figure 7.5: Average F -measure for each method compared to five user ground-truth summaries for Video #21. Method summaries are generated using HSV and RGB feature spaces. Summaries are matched using histograms of hue values for the selected frames.

- Some methods (GMM, ZNCC, SCC, and DIV) perform quite differently when the two different feature spaces are used, with a significantly better average F -measure with one of the spaces.
- The two methods that perform relatively well on the synthetic data sets (MGMM and SCX) generate very similar results when HSV and RGB features are used.
- For most methods, including those with very different results (e.g. GMM), the tuned parameters are similar for both feature spaces.
- However, parameters directly related to the feature space are naturally very sensitive to a change in features. For example, the optimum distance threshold parameter for the SCC method is 516 in RGB space, compared to 6 in HSV space.

Most of the methods perform better with the RGB moment features. Therefore, we use these features and the corresponding tuned parameters to generate summaries for the full set of VSUMM videos. Table 7.5 shows the average F -measure across all VSUMM videos, and the median number of frames selected.

Table 7.5: Average number of frames and F -measure for summaries generated by each method of the 50 VSUMM videos using RGB moments, and average F -measure with the features originally used with the method. The F -measures are also averaged across the five ground truth summaries for each video.

| Method | RGB | | Original. features |
|--------|----------------------------|----------------------|----------------------|
| | Median number of frames | Mean F -measure | Mean F -measure |
| SBD | 10 | 0.52 | 0.40 |
| ZNCC | 1 | 0.18 | 0.17 |
| DIV | 15 | 0.39 | 0.20 |
| SCX | 13 | 0.54 | 0.54 |
| MSR | 2 | 0.23 | 0.35 |
| GMM | 0 | 0.03 | 0.12 |
| HIST | 4 | 0.38 | 0.39 |
| MGMM | 17 | 0.52 | - |
| SCC | 3 | 0.27 | - |

The method generating the best results on the synthetic data (MGMM), again produces relatively good summaries for the videos. The MSR method performs markedly better on the real videos, with a higher-dimensional feature space, than on the synthetic data. The SBC method performs differently when feature spaces are changed. The SCX method has the highest average F -measure. As an illustration of the results, the summary generated by this method for Video #29 is shown in Figure 7.6 in comparison to the ground truth summary from user 3. The method matches 7 of 8 frames selected by this user (shown next to the SCX frames in Figure 7.6).

There is little difference in the performance of the methods using their original features, compared to RGB moments, both in terms of average F -measure and overall ranking. The SCX method maintains the highest average F -measure, and although the average score for the GMM method improves, it still remains lower than the other methods. The DIV method scores a lower average F -measure when the original features are used, highlighting the importance of considering simple, efficient feature spaces.

Three observations can be made from the video summaries:

- The F -measures in Table 7.5 are generally low compared to those reported in the literature for other video summarisation methods. This

difference is to be expected because here we compare *on-line* methods which do not have access to the whole collection of frames.

- Most methods are highly sensitive to their parameter values. The optimal values tuned on video #21 are not directly transferable to the remaining videos. Most methods (ZNCC, MSR, GMM, HIST, and SCC) typically select too few keyframes. This indicates the importance of tuning. In the on-line scenario, data for tuning will not be available, especially the segment labels needed for calculating A .
- Most methods are tested using a different feature representation than that recommended by the authors (HSV histograms are used in only three of the methods: SBD, ZNCC, and HIST; none of the methods use RGB features). However, the relative performances do not appear to be overly sensitive to the choice of feature space.



Figure 7.6: Comparison of VSUMM Video #29 summaries from ground truth User #3 and the SCX method. The matches have been calculated using the 16-bin histogram method with threshold 0.5 [40]. The F -measure for the match is 0.88.

7.7 Experiments on the Proposed Method

Here we compare the results for the proposed CC method with the two existing methods, SCX and MGMM, found to perform the best in the Section 7.6.

7.7.1 Results on Synthetic Data

We first consider the performance of the three methods on seven synthetic data sets. The first data set follows the example of Elhamifar et al. [52], as explained in Section 7.6.1. Data sets #2 - #5 each contains an additional two noise dimensions. Data sets #6 and #7 follow a similar structure but with more dimensions, six and eight, respectively. All data sets are shown in Figure 7.7. Using synthetic data allows an objective assessment of the summaries produced.

We train the method parameters on 50 randomly generated data sets following the distribution of Data set #1. Solutions are evaluated as follows:

- Find the Pareto set for the three criteria A , K and J .
- Exclude any results in the Pareto set with $K > 10$. This step removes the solution that selects *all* frames as keyframes, giving perfect accuracy and no error.
- Select the summary with the best accuracy. Where multiple summaries tie, select that with the fewest frames, and use the approximation error to split any remaining ties.

Taking the 50 optimal parameter sets as a cluster, the set closest to the cluster centre is chosen as the tuned method parameters.

The methods are then tested on 300 randomly generated data sets, 50 from each of the remaining six data set patterns, using the parameters tuned on Data set #1. For each data set the accuracy, cardinality and approximation error are calculated for each method. The methods are then

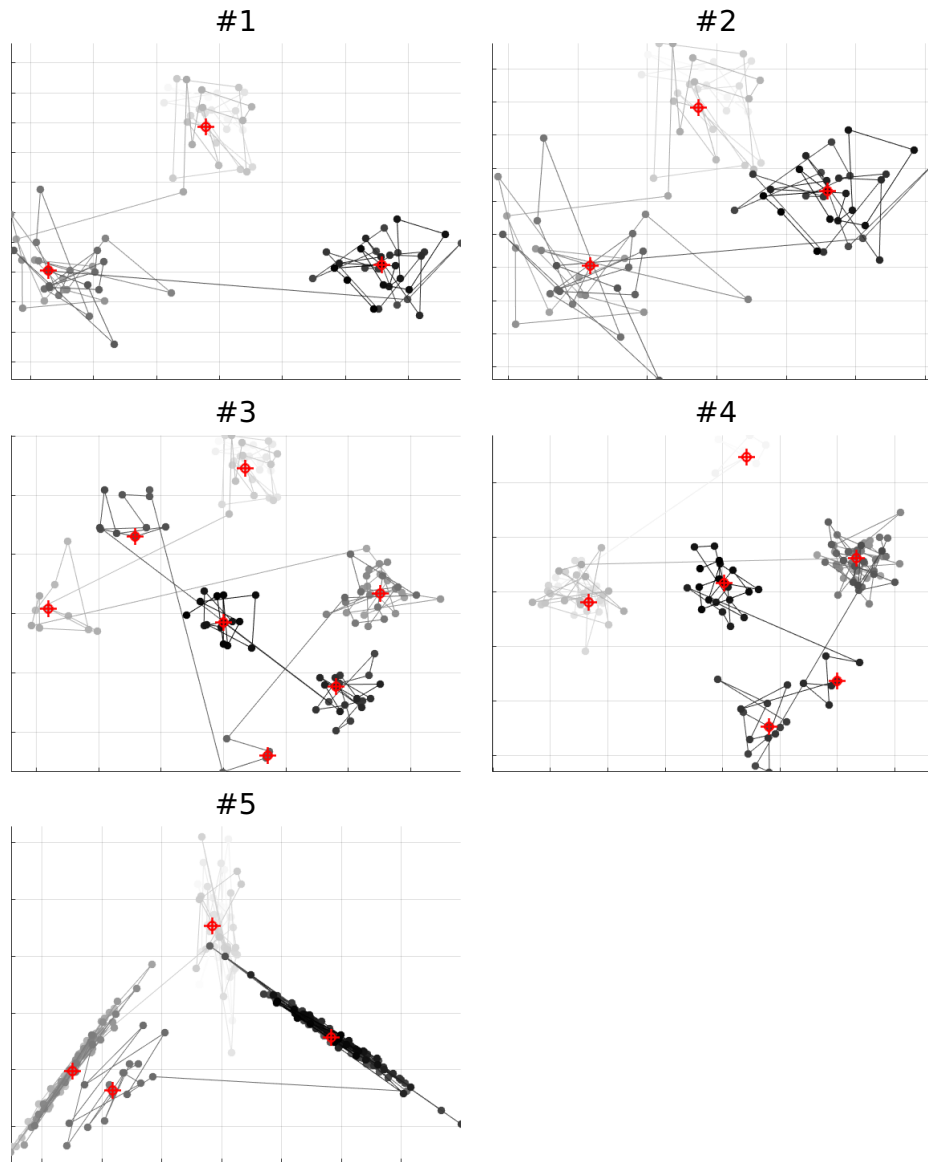


Figure 7.7: Synthetic Data sets #1 - #5. The time tag is represented as the grey intensity. Earlier points are plotted with a lighter shade. The “ideal” selected sets are shown with red target markers.

ranked. Four paired-sample t-tests are performed, comparing the accuracy and approximation error for our proposed CC method against the two existing methods.

Table 7.6: Results of paired-sample t-tests comparing the accuracy (A) and approximation error (J) for the CCS method summaries and the summaries generated by the MGMM and SCX methods. The confidence interval for the difference is shown for significant results (at the 0.05 significance level).

| Method | Test | P-value | Confidence interval |
|--------|----------------------|---------|---------------------|
| MGMM | $A_{CCS} - A_{MGMM}$ | 1e-5 | [0.02, 0.04] |
| | $J_{CCS} - J_{MGMM}$ | 6e-4 | [-1.7, -0.4] |
| SCX | $A_{CCS} - A_{SCX}$ | 0.7 | - |
| | $J_{CCS} - J_{SCX}$ | 3e-23 | [-4.0, -2.7] |

Table 7.6 shows the results of the paired-sample t-tests. At the 0.05 level, there is no significant difference between the accuracy values for the CCS and SCX methods (i.e. the difference has a zero mean). All other tests find a significant difference. The confidence intervals for the mean differences are less than zero for J , implying that the error tends to be less for the CCS method, and greater than zero for A , implying that the accuracy tends to be greater for the CCS method. The CCS method summaries tend to rank best according to our three criteria; an average of 1.4 across the 300 experiments, compared to the existing methods that have average ranks of 2.2 and 2.3 for the MGMM and SCX methods, respectively.

7.7.2 Results on VSUMM Videos

The methods are tested on the VSUMM collection [40]. Whereas the summaries of the synthetic data can be assessed in relation to a “correct” result, there is no such objective assessment available for real videos; what constitutes a good summary is somewhat subjective. Again, we follow the approach of Avila et al. [40] for similarity and metric, as explained in Section 7.5.1.

Parameters for each method are tuned on Video #21. We select the parameters that produce the summary with the highest average F -measure when compared with the five user ground-truth summaries. These parameters are used to run the methods on the other 49 videos.

Figure 7.8 shows the F -measure (averaged across the five user summaries) versus the number of keyframes selected by each method for the VSUMM videos. Each point on the plot corresponds to a video. The ideal summary has a high F -measure, and low number of frames. Points in the upper-left corner of the plots shown in Figure 7.8 therefore represent the better summaries. The points for all methods are plotted with grey colour on all plots. The points of the method in the title of the subplot are shown with black markers. The CCS method generates a higher proportion of good summaries than the existing two methods. As an illustration of these results, Figure 7.9 shows the summary of Video #47 produced by the CCS method, compared to the

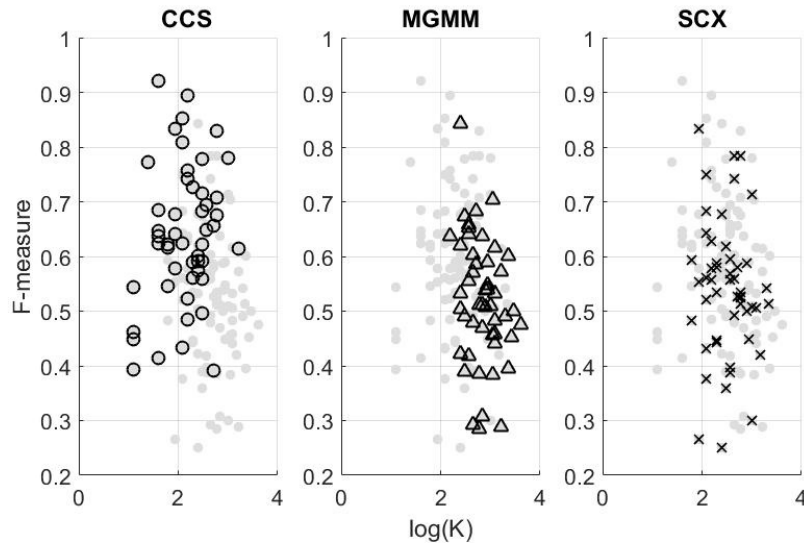


Figure 7.8: Number of keyframes (K) and F -measure averaged over five user ground-truths, for summaries of the 50 VSUMM videos. Filled, grey circles show the results for all three methods, with the points for the named method highlighted in black.

summary from User #1. All five frames in the user summary are matched in the CCS method summary.

7.8 Experiments on Comparing the Descriptors

The purpose of this experiment is to evaluate the feature spaces in regard to their suitability for on-line keyframe summarisation, application for egocentric videos. Thus, we chose to test the algorithm on the Activity of Daily Living (ADL) data set [133]. The ADL data set was recorded using a chest-mounted GoPro camera which consists of 20 videos of subjects performing their daily



Figure 7.9: Comparison of VSUMM Video #47 summaries from User #1 and the CCS method. The matches have been calculated using the 16-bin histogram method with threshold 0.5 [40].

activities in the house. For the experiment, we consider two aspects: ease of calculation of the feature space and the quality of the produced summary.

Table 7.7: Comparison of the average time of feature extraction for the toy video and the average MCC-value for all 20 videos. The best value for each column is highlighted in Bold.

| | Used in | Image Size | Visual Info. | | | |
|-------------------|---------|------------|--------------|----------------------------|------------|-------------|
| | | Resized | Original | Colour Scene Deep learning | Dimensions | Time(sec) |
| RGB moments | — | ✓ | ✓ | | 54 | 50 |
| Color Layout | [128] | ✓ | ✓ | | 192 | 519 |
| CENTRIST | [112] | ✓ | | ✓ | 254 | 160 |
| HSV histogram | [8] | ✓ | | ✓ | 256 | 30 |
| Gist | — | ✓ | | ✓ | 512 | 232 |
| Places205-AlexNet | — | | ✓ | ✓ | 4096 | 494 |
| VGGNet | [10] | ✓ | | ✓ | 4096 | 2377 |
| | | | | | | 0.68 |
| | | | | | | 0.52 |
| | | | | | | 0.63 |
| | | | | | | 0.45 |
| | | | | | | 0.45 |
| | | | | | | 0.46 |
| | | | | | | 0.43 |

7.8.1 Extraction Time

All experiments were carried out on a laptop, 2.20 GHz Intel Core *i5* CPU, with 8GB RAM. The first part of our analyses compares the processing time to extract the different features for the toy video. The ‘toy video’ is a selection of the initial 495 frames from Video #8 of the same data set. For each descriptor, we calculated the average time of extraction by repeating the process 20 times. The results are shown in Table 7.7. The extraction time for the simple colour spaces (RGB moments and HSV histograms) is shorter than the time for the other descriptors, whereas the popular VGGNet has the longest extraction time.

7.8.2 Performance Measure

We chose the Matthews correlation coefficient (MCC) [111] between the selected summary S and a given ground truth as a performance indicator. The MCC defined as:

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (7.4)$$

where TN is the number of true negatives, and TP , FP , and FN are as defined in Section 6.4.3. The ground truth for the data set was created as follows: Each event in the video is distinguished by a number of terms. The frames in an event are labelled as informative/not informative based on whether they contain semantic information that is included in the relevant terms for this event. Consequently, any informative frame from the event can be considered ground truth for that event.

7.8.3 Quality of the Keyframe Summary

The average MCC-values using the chosen feature space for 20 videos are shown in Table 7.7. The higher the value, the better the quality of the summary. The RGB moments has the highest MCC-value, and the VGGNet descriptor, the lowest value. CENTRIST feature space gave better performance than CNN, and was also faster to extract. The difference between MCC-values for the HSV histogram, Gist and the CNN descriptors are not large. However, the HSV histogram has fewer dimensions and substantially faster processing time.

7.9 Conclusions

Our experiments highlight the difficulty in pre-tuning the parameters of on-line video summarisation algorithms. This limitation suggests that algorithms are needed which are more robust to their parameter fluctuations, and ideally should adapt with the streaming data.

The relative performance of the methods appears to be independent of the strategy for grouping the frames into segments or clusters and of the similarity measure used. We note that, according to our experiments, no strategy or measure produced consistently good or consistently bad summaries. The methods that select the cluster centres as the keyframe set produce better summaries than those that select keyframes conditionally. Perhaps unsurprisingly, the method that decides the number of keyframes *a priori*, tends to perform less well than those that can continue to add keyframes as required, suggesting that on-line algorithms need flexibility to adapt the

number of keyframes to the data. This requirement must be balanced with the memory restrictions inherent in on-line video summarisation.

The videos used for testing in the comparative study have well-defined shots, providing a relatively easy summarisation task. The performance of the methods may be different on other types of video, e.g. where the shots are less clearly defined or the variability within shots is greater. Examples of such type of data are egocentric videos and lifelogging photo streams.

Our proposed CCS method performs well in comparison to existing methods, both on small synthetic data sets and real videos. On-line methods require computationally inexpensive feature spaces. The experiments show that for our on-line summarisation, simple, colour-based descriptors offer a substantially more efficient and higher quality summary than the complex CNN features tested. For the colour-based descriptors, the use of resized images does not appear to adversely affect the summary quality. Image compression is therefore an interesting area to explore for on-line video summarisation, with a potential for further gains in efficiency.

Performance on longer videos must also be considered. For the application of wearable devices, it may be necessary to introduce a restriction on the number of keyframes that can be selected.

Similarly, when shots can potentially become very long, or consecutive shots very similar, a more dynamic approach to sampling, and the shot detection and similarity thresholds may be beneficial, and will be investigated in future work.

Following this, we will develop on-line video summarisation method for egocentric video stream in the next chapter.

Chapter 8

Control-Charts for Generating Budget-Constrained On-line Summary

8.1 Problem Statement

Nine on-line video summarisation methods were described and experimentally compared on non-egocentric video in Chapter 7. While these methods work fairly well for non-egocentric videos, it is reasonable to expect that loosely defined event boundaries in egocentric videos will render their performance inadequate. Therefore, this chapter proposes a new on-line summarisation method suitable for egocentric video (Figure 8.1).

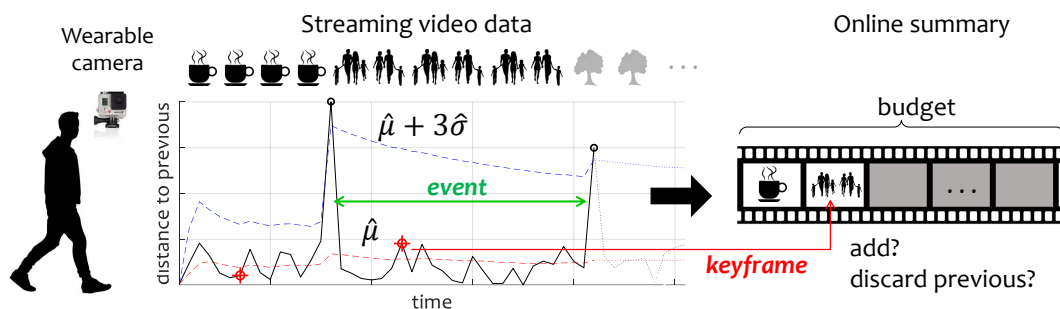


Figure 8.1: A sketch of the proposed on-line video summarisation method for egocentric video. The plot shows the Shewhart chart of the distance between consecutive frames, with the mean μ and the 3σ event-detection boundary, both calculated from the streaming data.

At any moment of the recording video, a valid summary is accessible up to that moment. We required that the new method has low computational complexity and is robust with respect to the feature representation of the

video frames. We compare our method against the top-performing on-line summarisation method from the Chapter 7 (called ‘submodular convex optimisation’ [51]), and a baseline method of uniform sampling of events (named ‘uniform events’). Moreover, we evaluate results making ground truth based on annotating frames (of a video) on semantic information rather than pixel-based comparison with a set of frames representing ground truth (visual comparison).

8.2 On-line Video Summarisation

Consider a scenario where the user’s daily activities are recorded using a wearable camera. To create an on-line summary, the video frames are represented as feature vectors in some feature space. A ‘budget’ is set as the maximal allowed number of frames in the summary. Next, the system saves the extracted keyframes generated by the on-line video summarisation algorithm if the budget allows for this. Should the limit be reached, one or more of the frames already stored in the summary is removed. Below we explain the steps of our algorithm.

8.2.1 Budget-Constrained On-line Video Summarisation

In statistics, control charts have been used to monitor and control ongoing processes over time. In Chapter 7, we introduced the use of control charts to identify event boundaries from a streaming video. The closest frame to the center of each event, represented as a cluster in the feature space, is selected as a keyframe. Here, we additionally, impose a constraint on the number of keyframes, hence the term ‘budget-constrained’ video summarisation. We also introduce a dynamic, similarity threshold into the algorithm that varies the probability of selecting new keyframes according to the number of existing keyframes and total budget. The pseudo-code of the algorithm is given in Algorithm 9 ¹.

¹Matlab code is available at: <https://github.com/pariay/Budget-constrained-on-line-Video-Summarisation-of-Egocentric-Video> (As of August 2019)

Algorithm 9: Budget-constrained online video summarisation

Input: Data stream $F = \{f_1, \dots, f_N\}$, $f_i \in \mathbb{R}^L$, initial buffer size b , minimum event length ms , threshold parameter for keyframe difference θ , desired number of keyframes β .

Output: Selected set of keyframes $K \subset F$, $|K| \leq \beta$.

```
// Initialisation
1  $K \leftarrow \emptyset$ 
2  $E \leftarrow \{f_1, \dots, f_b\}$  // initial buffer
3 Calculate the  $b - 1$  distances between the consecutive frames in  $E$ .
4  $\mu \leftarrow$  average distance.
5  $\sigma \leftarrow$  standard deviation.

// Processing of the Video
6 for frame number  $i = b + 1, \dots, N$  do
7    $d_i \leftarrow d(f_i, f_{i-1})$  // Calculate distance to previous frame
8   if  $d_i \leq \mu + 3\sigma$  then // Same event
9      $[\mu, \sigma] \leftarrow$  update  $\mu$  &  $\sigma$  with  $d_i$ 
10     $E \leftarrow E \cup f_i$  // Store the frame
11  else if  $|E| < ms$  then // Event too short
12     $E \leftarrow f_i$  // Remove frames in  $E$  and start a new event
13  else // Event sufficiently long
14     $k \leftarrow \text{Select-Keyframe}(E)$ 
15    if  $K$  empty then // First keyframe
16       $K \leftarrow k$ 
17    else //  $k$  included if sufficiently different to  $K$ 
18       $k_{last} \leftarrow$  last keyframe in  $K$ 
19       $\delta \leftarrow \text{Keyframe-Diff}(k, k_{last})$ 
20       $\delta_{min} \leftarrow$  smallest distance among consecutive keyframes in  $K$ 
21      if  $|K| < \beta$  &  $\delta > \text{Diff-Threshold}(|K|, i, \theta, \beta, N)$  then // In budget
22         $K \leftarrow K \cup k$ 
23      else if  $\delta \geq \delta_{min}$  then // Over budget
24        Remove from  $K$  one of the keyframes in the closest pair.
25         $K \leftarrow K \cup k$ 
26     $E \leftarrow f_i$  // new event

27
28 Function  $f = \text{Select-Keyframe}(data)$ 
29  $f \leftarrow \underset{x \in data}{\text{argmin}} d(x, \text{mean}(data))$ 
30
31 Function  $\delta = \text{Keyframe-Diff}(f_1, f_2)$ 
32  $h_i = \text{Hist16}(\text{Hue}(f_i))$  // Normalised 16-bin Hue histogram
33  $\delta = \frac{1}{16} \sum_{j=1}^{16} |h_1(j) - h_2(j)|$ 
34
35 Function  $\theta_{new} = \text{Diff-Threshold}(n_k, t, \theta, \beta, T)$ 
36  $n_t \leftarrow \beta \times t/T$  // Expected number of keyframes, assuming linear
    distribution
37 if  $n_t == \beta$  then
38    $\theta_{new} = 0$ 
39 else
40    $\theta_{new} \leftarrow \frac{\theta \times (\beta - n_k) + (n_k - n_t)}{\beta - n_t}$ 
```

Given an integer constant β , the purpose is to select a set of no more than β keyframes which describe the video as fully and accurately as possible. Unlike the classical summarisation approaches, we derive the summary on-the-go by processing each frame as it comes and selecting keyframes before the full video content is available. The algorithm requires only a limited memory to keep the frames selected thus far, and the frames belonging to the current event.

A control chart is used to detect the event boundaries [146]. The quantity being monitored is the difference between consecutive frames, defined by the distance between the frames in some chosen feature space \mathbb{R}^L . Assuming that the frames are represented as points \mathbb{R}^L , the hypothesis is that different events in the video are represented by relatively distant clusters. Then transition from one event to the next will be associated with large distance between consecutive frames. As both outlier and transition frames may be detected as an event boundary, we observe a minimum event size, ms . If the number of frames in an event is less than ms , the algorithm ignores the candidate-event without extracting a keyframe. This approach is suitable for clearly distinguishable shots (events), as seen for the traditional video stream in Chapter 7. For application to egocentric videos, in this chapter we adapt the approach to allow for less well-defined shots. In addition, the budget constraint provides a means of defining an expected or desired number of events to be captured. Egocentric videos are not easily split into coherent events. To improve the event detection, we compare a selected keyframe with its immediate predecessor. If the keyframes of the adjacent events are deemed similar, the new event is ignored, without extracting a keyframe. The tolerance for accepting similarity between frames varies in relation to how close to the overall budget the existing set of keyframes is, and how many more events may be expected in the video. Note that this assumes prior knowledge of roughly how long the video will be. If the budget for keyframes is reached while frames are still being captured, keyframes from any additional events are only saved if the keyframe set is made more diverse by the substitution of the new keyframe for an existing keyframe.

Assume a video stream is presented as a sequence of frames, $F = \{f_1, \dots, f_N\}$, $f_i \in \mathbb{R}^L$, where L indicates the dimensions of the frame descriptor. For any upcoming frame, the similarity of consecutive frames f_i and f_{i-1} is calculated using Euclidean distance $d(.,.)$ in \mathbb{R}^L . Denote $d_i = d(f_i, f_{i-1})$. In the process of monitoring quality control, the probability p of an object being defective is known from the product specifications or trading standards. This probability is the quantity being monitored. For the event boundary detection in videos, we need to monitor the distance d_i . The initial values can be calculated by taking average values of the first b distances: $\mu = \frac{1}{b-1} \sum_{i=2}^b d_i$, and computing the standard deviation value of the first b distances as: $\sigma = \sqrt{1/(b-1) \sum_{i=2}^b (d_i - \mu)^2}$. At time point $i+1$, the distance value d_{i+1} is calculated and compared with the μ and σ at time point i . A change is detected if $d_{i+1} > \mu + \alpha\sigma$. The value of α typically is set to 3, but other alternatives are also possible.

The measure of similarity between two selected adjacent keyframes follows the study of Avila et al. [40]. Those keyframes are represented by 16-bins histograms of the hue value. Keyframes are similar if the Minkowski distance between their normalised histograms is less than a threshold θ , and are dissimilar otherwise.

The proposed algorithm requires four parameters: the initial buffer size (b), the minimum event length (ms), the pre-defined threshold value for keyframe similarity (θ), and the maximum number of keyframes (β).

8.2.2 Choosing Parameter Values

An empirical value for the desired number of the keyframes, β , has been obtained following the study by Le et al. [87]. The authors collected a total of 80 image sets from 16 participants from 9am to 10pm using lifelogging devices. An average of 28 frames per image set were chosen by the participants to represent their day. Therefore, in our experiment we set this parameter to $\beta = 28$.

We sample one frame per second for each video. The buffer size b was selected to be equal to one minute, $b = 60$. The minimum event length empirically was set to thirteen seconds, $m = 13$. The empirical threshold value for keyframe similarity was set to $\theta = 0.7$.

8.2.3 Feature Representation

The proposed algorithm is not tailor-made for any particular descriptor, therefore any type of descriptor may be applied. Following the preliminary experiment in Chapter 7, we chose the RGB feature space as the best compromise between the two criteria.

8.3 Experimental Results

8.3.1 Data Set

The algorithm performance was evaluated on the Activity of Daily Living (ADL) dataset² [133]. This dataset was recorded using a chest-mounted GoPro camera and consists of 20 videos (each lasting about 30 minutes to one hour) of subjects performing their daily activities in the house.

8.3.2 Annotation Strategy

Evaluation of keyframe video summarisation for egocentric videos is still a challenging task [115, 60]. Using visual comparison between a computer-generated summary and a ground truth set, human annotators show discrepancies on selecting one ‘ideal’ frame per event to represent the video summary. Despite that human annotators can simply demonstrate the semantic information through words [177]. Besides, many frames can represent one semantic concept of what’s happened in that event whereas the event can be likely expressed by just one sentence. Yeung et al. [177] suggested to evaluate summaries through text using the VideoSET method³. In their experiments, the author provided text annotations per frame for the video to be summarised. The VideoSet method converts the summary into

²<https://www.csee.umbc.edu/~hpirsiav/papers/ADLdataset/> (As of August 2019)

³<http://ai.stanford.edu/~syueung/videoset.html> (As of August 2019)

text representation. Then the content similarity between this representation and a ground truth text summary was measured through Natural Language Processing (NLP).

Motivated by [177], we annotated the ADL dataset rather using numbers than text. The numbers are organised to describe sequences of events. We made a list of events in each video, using an action list from [133]. The frames are labelled with their relevant event, or as not informative if the event cannot be recognised from the frame (semantic information). Consequently, any informative frame from the event can be considered ground truth for that event. Given a video summary, the number of matches and then the F-measure can be subsequently calculated.

8.3.3 Rival On-line Video Summarisation Methods

We compared the following methods:

1. The proposed Budget-constrained Control Chart algorithm (BCC).
2. Submodular convex optimisation [51] (SCX).
3. Uniform Events baseline method (UE). To implement the UE algorithm, the video is uniformly divided into ϵ number of events (segments). The ϵ value follows the number of keyframes extracted by our on-line algorithm. The closest frame to the center of each segment (in \mathbb{R}^L) is taken to represent the event.

To have a fair comparison we tuned the SCX and the UE for each video to their best performance. Doing that, the value for ϵ was adjusted with the number of keyframes extracted by our on-line algorithm. The same adjustment applied for the SCX.

8.3.4 Keyframe Selection Results

Table 8.1 shows the F-value for the match between the summaries generated through BCC, SCX and UE, and the semantic-category ground truth for the 20

videos. As seen from these results, the proposed on-line method performs consistently better than the two competitors.

Table 8.1: F-values for the comparison of the proposed method (BCC), and the two rival methods (SCX and UE) on the 20 videos in ADL video database. The best value for each video is emphasised in bold.

| Video | Number of Frames | F-measure | | | Parameters | |
|----------|------------------|-------------|-------------|-------------|------------------|------------------|
| | | BCC | SCX | UE | SCX(λ) | UE(ϵ) |
| P_{01} | 1,794 | 0.73 | 0.45 | 0.60 | 0.33 | 13 |
| P_{02} | 2,860 | 0.63 | 0.35 | 0.67 | 0.07 | 27 |
| P_{03} | 2,370 | 0.50 | 0.37 | 0.56 | 0.15 | 19 |
| P_{04} | 1,578 | 0.52 | 0.31 | 0.44 | 0.25 | 18 |
| P_{05} | 1,475 | 0.42 | 0.30 | 0.42 | 1 | 5 |
| P_{06} | 1,550 | 0.67 | 0.53 | 0.47 | 0.2 | 20 |
| P_{07} | 2,643 | 0.81 | 0.43 | 0.54 | 0.17 | 18 |
| P_{08} | 1,592 | 0.56 | 0.40 | 0.60 | 0.08 | 27 |
| P_{09} | 1,288 | 0.67 | 0.61 | 0.56 | 0.15 | 25 |
| P_{10} | 956 | 0.80 | 0.40 | 0.80 | 0.7 | 8 |
| P_{11} | 493 | 0.87 | 0.52 | 0.78 | 0.6 | 10 |
| P_{12} | 844 | 0.69 | 0.43 | 0.69 | 0.3 | 14 |
| P_{13} | 1,768 | 0.63 | 0.28 | 0.51 | 0.11 | 24 |
| P_{14} | 1,531 | 0.78 | 0.54 | 0.63 | 0.09 | 23 |
| P_{15} | 1,585 | 0.59 | 0.37 | 0.59 | 0.25 | 13 |
| P_{16} | 840 | 0.89 | 0.64 | 0.59 | 0.19 | 13 |
| P_{17} | 885 | 0.44 | 0.44 | 0.22 | 0.28 | 9 |
| P_{18} | 1,150 | 0.47 | 0.47 | 0.40 | 0.095 | 21 |
| P_{19} | 3,797 | 0.77 | 0.33 | 0.57 | 0.08 | 28 |
| P_{20} | 1,609 | 0.69 | 0.31 | 0.50 | 0.17 | 16 |

Figure 8.2 displays the summaries obtained by the BCC, SCX and UE methods, highlighting matched frames with the ground truth. Our BCC method misses one event (Event number 7) in the ground truth (see (a) in Figure 8.2) resulting in the F-value of 0.89.

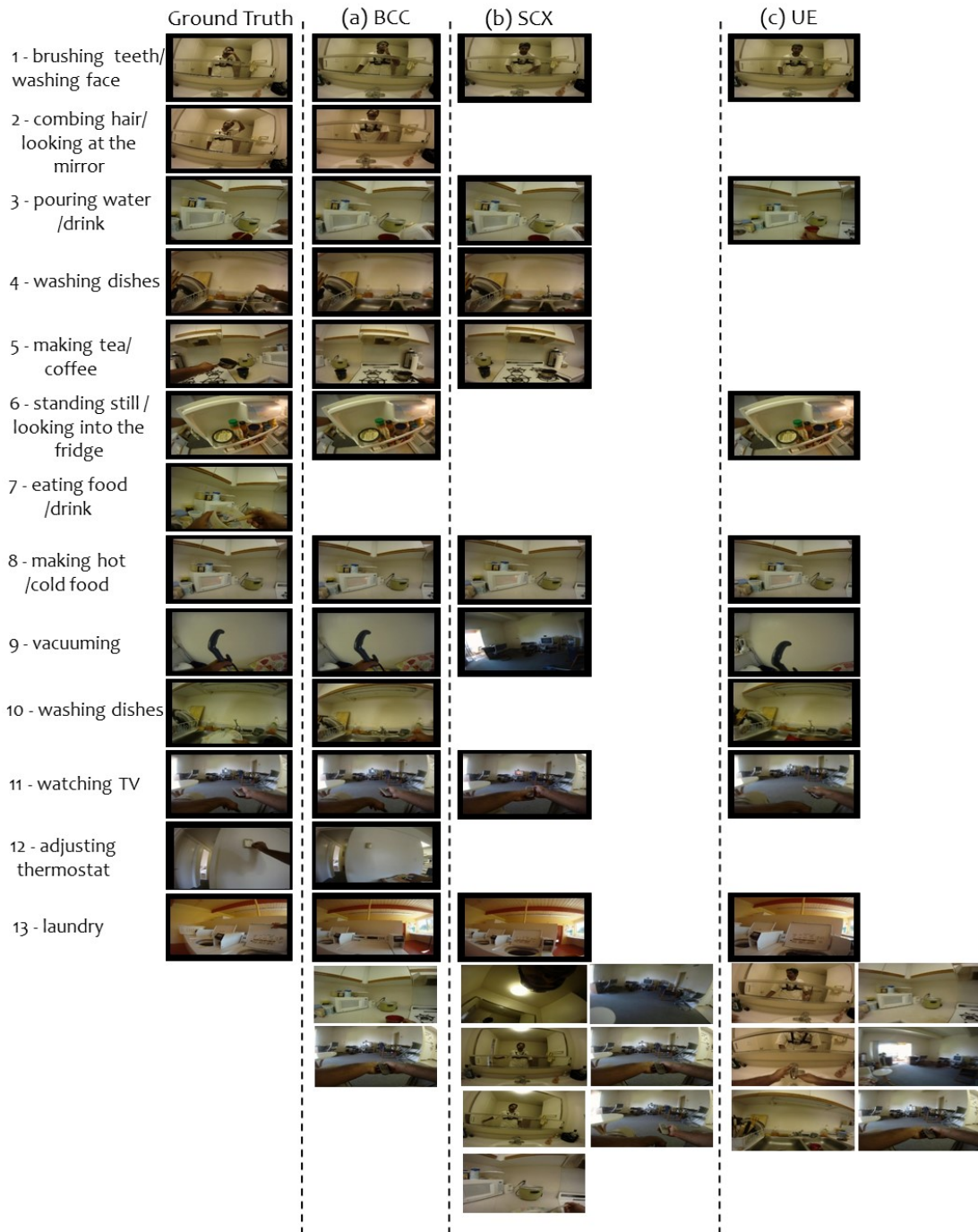


Figure 8.2: Example of keyframe summaries obtained by the (a) BCC, (b) SCX and (c) UE methods and their matched frames with the ground truth, for ADL dataset video #16. The total number of events in ground truth for this video is 13, and the BCC just missed one event on eating food/drink.

8.4 Conclusion

The purpose of the chapter was to propose a fast and effective method (BCC) to extract a keyframe summary from a streaming video. The proposed method applies control charts to detect event boundaries on-line, and observes a

maximum limit on the number of selected keyframes (budget-constrained). Our experiments with 20 egocentric videos from the ADL video database demonstrate that BCC performs well in comparison with two existing methods, state-of-the-art SCX and baseline UE. The BCC method uses colour-based descriptor (e.g. RGB moments), rather than the complex CNN features because they are significantly faster to extract and able to produce a relatively high-quality summary.

The requirement to store all frames for an event before the keyframe is selected could present memory issues in the event of excessively long, sedentary events, e.g. sleeping. Such events may be relatively common in application areas such as monitoring daily activity. One way to deal with this issue is the introduction of a dynamic frame-rate, with far fewer frames recorded during such events.

Chapter 9

Conclusions and Future Work

9.1 Conclusions

Recent advances in technology enable users to capture every single moment of their lives with an egocentric camera, leading to an increase in demand for a summarisation system that retrieves information requested by the user. The aim of this project was to contribute solutions to some of the problems in the area of egocentric video summarisation.

Objective 1 was to investigate the current approaches for evaluating video summarisation methods. To accomplish that, we proposed a new automatic evaluation protocol for comparing keyframe summaries. We investigated experimentally a range of choices for the different components of the protocol. This includes ten feature types, six algorithms for matching (pairing) of two summaries. We propose a “discrimination capacity” measure, which evaluates by how much a given summary improves on the uniform keyframe summary of the same cardinality. Using a benchmark video data, we offer empirical recommendations.

Our protocol is limited to discovering only visual similarity and ignores time sequence. Useful extensions could focus on contextual or semantic similarity as well as comparing the whole “story” captured by the two summaries.

Objective 2 was to propose a stronger baseline method for comparative evaluation. To complete this, we propose a new baseline model for creating a keyframe summary, called “Closest-to-Centroid”. Using a widely-used

egocentric video database, we examine the new baseline model on twenty feature types. We show that it is a better contestant compared to the two most popular baselines: uniform sampling, and choosing the mid-event frame. Random sampling is not taken forward because it is deemed to be the weakest baseline anyway.

Next objective was to propose a new keyframe summarisation method. To complete Objective 3, we cast the problem of selecting a keyframe summary as a problem of prototype (instance) selection for the nearest neighbour classifier (1-nn). It is assumed that the video has already been split into units (segments or events), and represented as a data set in some feature space. We propose a Greedy Tabu Selection (GTS) method for extracting a keyframe summary. Following a cartoon example, we illustrate that re-positioning a diversity-wise selection as an edited nearest neighbour problem requires no manual setting of the balance between diversity and representativeness/coverage. An experiment with a widely-used egocentric video database, and seven feature representations illustrates the proposed keyframe summarisation method. GTS leads to improved match to the user ground truth compared to the closest-to-centroid baseline summarisation method.

Considering the need for personalised summary (Objective 4), we proposed a method to extract a selective, time-aware keyframe summary of an egocentric video. The problem was solved by applying a pipeline of a semantic concept search, occurrence-led event segmentation, and finally a cluster centroid keyframe selection. A compass-type diagram was proposed to visualise the selective summary. Using our system, a user can query the same video stream by multiple vocabulary of terms, and obtain multiple time-tagged summaries related to the query concepts. The system is evaluated in two commonly used egocentric data sets.

Finally, Objective 5 was to explore the current state-of-the-art on-line video summarisation. To complete this objective, we investigated nine existing on-line video summarisation methods. We proposed a classification for on-line

video summarisation methods based upon their descriptive and distinguishing properties such as feature space for frame representation, strategies for grouping time-contiguous frames, and techniques for selecting representative frames. Subsequently, we propose an on-line video summarisation algorithm to generate keyframe summaries during video capture. Event boundaries are identified using control charts and a keyframe is dynamically selected for each event. The number of keyframes is restricted from above which requires a constant review and possible reduction of the cumulatively built summary. The new method was compared against a baseline and a state-of-the-art on-line video summarisation method. The summaries generated by the proposed method outperform those generated by the two competitors.

9.2 Future Work

Video summarisation would benefit from extracting computationally inexpensive features. Further investigation and experimentation into combining feature spaces is recommended. While concatenation of feature spaces is a straightforward solution, classifier ensembles may be more effective.

A future research line includes incorporating user searches on faces and people. Different to the research by Aghaei et al. [4], it is possible to develop a pipeline to recognise mostly seen faces, and more importantly detect new faces with their time tags. Given that specific query, the automatic analysis of recorded videos or photo streams can be used for improving security for the elderly or for reinforcing the memory. We were not able to explore this aspect with the publicly available databases because any faces in the frames were purposely blurred for identity protection.

An interesting step forward would be to create query-based on-line video summarisation methods. This is feasible by combining the work we have reported in Chapters 6 and 8.

References

- [1] W. Abd-Almageed, 'Online, simultaneous shot boundary detection and key frame extraction for sports videos using rank tracing', in *IEEE 15th International Conference on Image Processing (ICIP 2008)*, San Diego, CA, Oct. 2008, pp. 3200–3203 (pp. 14, 15, 76, 118, 120, 121, 125).
- [2] M. Aghaei, M. Dimiccoli, C. C. Ferrer and P. Radeva, 'Towards social pattern characterization in egocentric photo-streams', *Computer Vision and Image Understanding*, 2018 (p. 10).
- [3] M. Aghaei, M. Dimiccoli and P. Radeva, 'With whom do i interact? detecting social interactions in egocentric photo-streams', in *23rd International Conference on Pattern Recognition (ICPR 2016)*, IEEE, 2016, pp. 2959–2964 (p. 10).
- [4] —, 'All the people around me: Face discovery in egocentric photo-streams', in *IEEE International Conference on Image Processing (ICIP 2017)*, IEEE, 2017, pp. 1342–1346 (pp. 10, 161).
- [5] K. Aizawa, Y. Maruyama, H. Li and C. Morikawa, 'Food balance estimation by using personal dietary tendencies in a multimedia food log', *IEEE Transactions on multimedia*, vol. 15, no. 8, pp. 2176–2185, 2013 (p. 105).
- [6] S. Alletto, G. Serra, S. Calderara and R. Cucchiara, 'Understanding social relationships in egocentric vision', *Pattern Recognition*, vol. 48, no. 12, pp. 4082–4096, 2015 (p. 10).
- [7] J. Almeida, N. J. Leite and R. d. S. Torres, 'Comparison of video sequences with histograms of motion patterns', in *18th IEEE International Conference on Image Processing (ICIP 2011)*, IEEE, Sep. 2011, pp. 3673–3676 (p. 65).

- [8] —, 'VISON: Video Summarization for ONline applications', *Pattern Recognition Letters*, vol. 33, no. 4, pp. 397–409, 2012 (pp. 15, 17, 56, 146).
- [9] —, 'Online video summarization on compressed domain', *Journal of Visual Communication and Image Representation*, vol. 24, no. 6, pp. 729–738, Aug. 2013. doi: doi.org/10.1016/j.jvcir.2012.01.009 (pp. 14, 17, 76, 118, 120, 121, 125).
- [10] R. Anirudh, A. Masroor and P. Turaga, 'Diversity promoting online sampling for streaming video summarization', in *IEEE International Conference on Image Processing (ICIP2016)*, Phoenix, AZ, Sep. 2016, pp. 3329–3333 (pp. 14, 76, 118, 120, 121, 126, 146).
- [11] Y. S. Avrithis, A. D. Doulamis, N. D. Doulamis and S. D. Kollias, 'A stochastic framework for optimal key frame extraction from mpeg video databases', *Computer Vision and Image Understanding*, vol. 75, no. 1-2, pp. 3–24, 1999 (p. 16).
- [12] H. Bay, A. Ess, T. Tuytelaars and L. Van Gool, 'Speeded-up robust features (SURF)', *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008 (p. 67).
- [13] E. L. Berry, A. Hampshire, J. Rowe, S. Hodges, N. Kapur, P. Watson, G. Browne, G. Smyth, K. Wood and A. M. Owen, 'The neural basis of effective memory therapy in a patient with limbic encephalitis', *Journal of Neurology, Neurosurgery & Psychiatry*, 2009 (p. 8).
- [14] V. Bettadapura, D. Castro and I. Essa, 'Discovering picturesque highlights from egocentric vacation videos', in *IEEE Winter Conference on Applications of Computer Vision (WACV 2016)*, IEEE, Mar. 2016, pp. 1–9 (pp. 10, 14, 16, 23, 56, 74–76).
- [15] S. Bhattacharya, B. Nojavanasghari, T. Chen, D. Liu, S.-F. Chang and M. Shah, 'Towards a comprehensive computational model for aesthetic assessment of videos', in *Proceedings of the 21st ACM international conference on Multimedia*, ACM, Oct. 2013, pp. 361–364 (p. 59).
- [16] S. Bianco, L. Celona, P. Napoletano and R. Schettini, 'Predicting image aesthetics with deep learning', in *International Conference on Advanced Concepts for Intelligent Vision Systems*, Springer, 2016, pp. 117–125 (p. 59).

- [17] A. Bifet and R. Gavalda, 'Learning from time-changing data with adaptive windowing', in *Proceedings of the 2007 SIAM international conference on data mining*, SIAM, 2007, pp. 443–448 (p. 12).
- [18] S. Boehm, *Matlab centrist*, <https://github.com/sometimesfood/spact-matlab>, Accessed: 2018-08-01 (p. 130).
- [19] M. Bolaños, M. Dimiccoli and P. Radeva, 'Toward storytelling from visual lifelogging: An overview', *IEEE Transactions on Human-Machine Systems*, vol. 47, no. 1, pp. 77–90, 2017 (p. 13).
- [20] M. Bolaños, A. Ferrà and P. Radeva, 'Food ingredients recognition through multi-label learning', in *In 3rd International Workshop on Multimedia Assisted Dietary Management (ICIAP 2017)*, Springer, 2017, pp. 394–402 (p. 9).
- [21] M. Bolaños, R. Mestre, E. Talavera, X. Giró-i-Nieto and P. Radeva, 'Visual summary of egocentric photostreams by representative keyframes', in *IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, IEEE, Jun. 2015, pp. 1–6 (pp. 11, 14–16, 23, 56, 58, 60, 74–76, 102).
- [22] M. Bolaños and P. Radeva, 'Ego-object discovery', *arXiv preprint arXiv:1504.01639*, vol. abs/1504.01639, 2015 (p. 110).
- [23] —, 'Simultaneous food localization and recognition', in *23rd International Conference on Pattern Recognition (ICPR 2016)*, Cancun, Mexico, Dec. 2016, pp. 3140–3145. doi: 10.1109/ICPR.2016.7900117 (p. 9).
- [24] A. Bosch, A. Zisserman and X. Munoz, 'Representing shape with a spatial pyramid kernel', in *Proceedings of the 6th ACM International Conference on Image and Video Retrieval*, ACM, 2007, pp. 401–408 (p. 64).
- [25] H. Brighton and C. Mellish, 'Advances in instance selection for instance-based learning algorithms', *Data Mining and Knowledge Discovery*, vol. 6, no. 2, pp. 153–172, 2002 (p. 77).
- [26] R. Brindley, A. Bateman and F. Gracey, 'Exploration of use of sensecam to support autobiographical memory retrieval within a cognitive-behavioural therapeutic intervention following acquired brain injury', *Memory*, vol. 19, no. 7, pp. 745–757, 2011 (p. 8).

- [27] G. Browne, E. Berry, N. Kapur, S. Hodges, G. Smyth, P. Watson and K. Wood, 'Sensecam improves memory for recent events and quality of life in a patient with memory retrieval difficulties', *Memory*, vol. 19, no. 7, pp. 713–722, 2011 (pp. 8, 9).
- [28] E. J. C. Cahuina and G. C. Chavez, 'A new method for static video summarization using local descriptors and video temporal segmentation', in *XXVI Conference on Graphics, Patterns and Images*, IEEE, 2013, pp. 226–233 (p. 49).
- [29] N. Cao, Y.-R. Lin, F. Du and D. Wang, 'Episogram: Visual summarization of egocentric social interactions', *IEEE Computer Graphics and Applications*, vol. 36, no. 5, pp. 72–81, 2016 (p. 10).
- [30] G.-C. Chao, Y.-P. Tsai and S.-K. Jeng, 'Augmented keyframe', *Journal of Visual Communication and Image Representation*, vol. 21, no. 7, pp. 682–692, 2010 (p. 60).
- [31] S. A. Chatzichristofis and Y. S. Boutalis, 'FCTH: Fuzzy Color and Texture Histogram - a low level feature for accurate image retrieval', in *9th International Workshop on Image Analysis for Multimedia Interactive Services*, May 2008, pp. 191–196 (p. 62).
- [32] S. A. Chatzichristofis and Y. S. Boutalis, 'CEDD: Color and Edge Directivity Descriptor: A compact descriptor for image indexing and retrieval', in *International Conference on Computer Vision Systems (ICVS 2008)*, Springer Berlin Heidelberg, 2008, pp. 312–322 (p. 61).
- [33] S. Chatzichristofis, Y. Boutalis and M. Lux, 'Selection of the proper compact composite descriptor for improving content based image retrieval', in *Proceedings of the 6th IASTED International Conference*, vol. 134643, Feb. 2009, p. 064 (p. 63).
- [34] S. Chowdhury, P. J. McParlane, M. S. Ferdous and J. Jose, 'My day in review: Visually summarising noisy lifelog data', in *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*, ACM, Jun. 2015, pp. 607–610 (p. 60).
- [35] M. Cooper and J. Foote, 'Discriminative techniques for keyframe selection', in *IEEE International Conference on Multimedia and Expo*, IEEE, Jul. 2005, 4–pp (pp. 15, 60).

- [36] J. Corso, A. Alahi, K. Grauman, G. D. Hager, L. Morency, H. Sawhney and Y. Sheikh, 'Video analysis for body-worn cameras in law enforcement: A white paper prepared for the computing community consortium committee of the computing research association', Washington, DC, white paper, 2015 (p. 10).
- [37] F. Crete, T. Dolmiere, P. Ladret and M. Nicolas, 'The blur effect: Perception and estimation with a new no-reference perceptual blur metric', in *Human vision and electronic imaging XII*, International Society for Optics and Photonics, vol. 6492, 2007, pp. 6492–11 (p. 12).
- [38] N. Dalal and B. Triggs, 'Histograms of oriented gradients for human detection', in *International Conference on Computer Vision & Pattern Recognition (CVPR 2005)*, IEEE Computer Society, vol. 1, 2005, pp. 886–893 (p. 85).
- [39] B. V. Dasarathy, 'Nearest neighbor (NN) norms: NN pattern classification techniques', 1991 (p. 77).
- [40] 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*, vol. 32, no. 1, pp. 56–68, 2011 (pp. 15, 23–28, 32, 34, 37, 38, 40, 49, 55, 56, 58, 60, 67, 86, 92, 125, 131, 134, 141, 144, 145, 153).
- [41] B. Dijkstra, Y. Kamsma and W. Zijlstra, 'Detection of gait and postures using a miniaturised triaxial accelerometer-based system: Accuracy in community-dwelling older adults.', *Age and Ageing*, vol. 39, no. 2, pp. 259–262, 2010 (p. 9).
- [42] M. Dimiccoli, M. Bolaños, E. Talavera, M. Aghaei, S. G. Nikolov and P. Radeva, 'SR-clustering: Semantic regularised clustering for egocentric photo streams segmentation', *Computer Vision and Image Understanding*, vol. 155, pp. 55–69, 2017 (pp. 12, 67, 105).
- [43] F. Dirfaux, 'Key frame selection to represent a video', in *Proceedings International Conference on Image Processing (ICIP 2000)*(Cat. No. 00CH37101), IEEE, vol. 2, Sep. 2000, pp. 275–278 (p. 15).
- [44] A. R. Doherty, D. Byrne, A. F. Smeaton, G. J. Jones and M. Hughes, 'Investigating keyframe selection methods in the novel domain of

- passively captured visual lifelogs', in *Proceedings of the International Conference on Content-based Image and Video Retrieval (CIVR '08)*, ACM, Jul. 2008, pp. 259–268 (pp. 56, 59).
- [45] 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*, vol. 44, no. 3, pp. 320–323, 2013 (p. 8).
 - [46] A. R. Doherty, P. Kelly, J. Kerr, S. Marshall, M. Oliver, H. Badland and C. Foster, 'Use of wearable cameras to assess population physical activity behaviours: An observational study', *The Lancet*, vol. 380, S35, 2012 (p. 10).
 - [47] A. D. Doulamis, N. D. Doulamis and S. D. Kollias, 'A fuzzy video content representation for video summarization and content-based retrieval', *Signal Processing*, vol. 80, no. 6, pp. 1049–1067, 2000 (p. 16).
 - [48] M. Douze and H. Jégou, 'The yael library', in *22nd ACM International Conference on Multimedia*, ACM, Nov. 2014, pp. 687–690 (p. 65).
 - [49] N. Ejaz, I. Mehmood and S. W. Baik, 'Efficient visual attention based framework for extracting key frames from videos', *Signal Processing: Image Communication*, vol. 28, no. 1, pp. 34–44, 2013 (pp. 15, 23, 24, 27, 32, 56).
 - [50] N. Ejaz, T. B. Tariq and S. W. Baik, 'Adaptive key frame extraction for video summarization using an aggregation mechanism', *Journal of Visual Communication and Image Representation*, vol. 23, no. 7, pp. 1031–1040, 2012 (pp. 56, 60).
 - [51] E. Elhamifar and M. C. D. P. Kaluza, 'Online summarization via submodular and convex optimization', in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR2017)*, Hawaii, Jul. 2017, pp. 1818–1826 (pp. 14, 76, 118, 120, 122, 126, 150, 155).
 - [52] E. Elhamifar, G. Sapiro and S. S. Sastry, 'Dissimilarity-based sparse subset selection', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 11, pp. 2182–2197, Nov. 2016 (pp. 131, 142).
 - [53] M. Furini, F. Geraci, M. Montangero and M. Pellegrini, 'STIMO: STill and MOving video storyboard for the web scenario', *Multimedia Tools and Applications*, vol. 46, no. 1, p. 47, 2010 (pp. 15, 24, 40, 56, 60).

- [54] S. Garcia, J. Derrac, J. R. Cano and F. Herrera, 'Prototype selection for nearest neighbor classification: Taxonomy and empirical study', *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 3, pp. 417–435, 2011 (pp. 76, 77).
- [55] C. Gianluigi and S. Raimondo, 'An innovative algorithm for key frame extraction in video summarization', *Journal of Real-Time Image Processing*, vol. 1, no. 1, pp. 69–88, 2006 (p. 15).
- [56] B. Gong, W.-L. Chao, K. Grauman and F. Sha, 'Diverse sequential subset selection for supervised video summarization', in *Advances in Neural Information Processing Systems (NIPS2014)*, 2014, pp. 2069–2077 (pp. 16, 24, 27, 32, 39, 41, 49, 56, 60).
- [57] Y. Gong and X. Liu, 'Generating optimal video summaries', in *Proceedings IEEE International Conference on Multimedia and Expo. (ICME2000). Latest Advances in the Fast Changing World of Multimedia (Cat. No. 00TH8532)*, IEEE, vol. 3, Jul. 2000, pp. 1559–1562 (p. 56).
- [58] —, 'Video summarization using singular value decomposition', in *Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2000) (Cat. No. PR00662)*, IEEE, vol. 2, Jun. 2000, pp. 174–180 (p. 60).
- [59] G. Guan, Z. Wang, S. Lu, J. Da Deng and D. D. Feng, 'Keypoint-based keyframe selection', *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 4, pp. 729–734, 2013 (pp. 15, 56).
- [60] M. Gygli, H. Grabner, H. Riemenschneider and L. Van Gool, 'Creating summaries from user videos', in *European Conference on Computer Vision (ECCV 2014)*, Springer, 2014, pp. 505–520, isbn: 978-3-319-10584-0 (pp. 14, 16, 24, 25, 27, 60, 74–76, 86, 102, 154).
- [61] M. Gygli, H. Grabner and L. Van Gool, 'Video summarization by learning submodular mixtures of objectives', in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 3090–3098 (pp. 14, 17, 24, 27, 74–76, 102).
- [62] Y. Hadi, F. Essannouni and R. O. H. Thami, 'Video summarization by k-medoid clustering', in *Proceedings of the 2006 ACM Symposium on Applied Computing*, ACM, 2006, pp. 1400–1401 (p. 15).

- [63] A. Hanjalic and H. Zhang, 'An integrated scheme for automated video abstraction based on unsupervised cluster-validity analysis', *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 8, pp. 1280–1289, 1999 (pp. 15, 56).
- [64] P. Hart, 'The condensed nearest neighbor rule (corresp.)', *IEEE Transactions on Information Theory*, vol. 14, no. 3, pp. 515–516, 1968 (p. 77).
- [65] J. A. Harvey, D. A. Skelton and S. F. Chastin, 'Acceptability of novel life logging technology to determine context of sedentary behavior in older adults', *AIMS Public Health*, vol. 3, no. 1, p. 158, 2016 (p. 10).
- [66] K. He, X. Zhang, S. Ren and J. Sun, 'Deep residual learning for image recognition', in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016)*, Jun. 2016, pp. 770–778 (p. 105).
- [67] L. Herranz and J. M. Martínez, 'An efficient summarization algorithm based on clustering and bitstream extraction', in *IEEE International Conference on Multimedia and Expo, ICME 2009*, IEEE, 2009, pp. 654–657 (pp. 15, 56, 60).
- [68] S. Hodges, L. Williams, E. Berry, S. Izadi, J. Srinivasan, A. Butler, G. Smyth, N. Kapur and K. Wood, 'Sensecam: A retrospective memory aid', in *International Conference on Ubiquitous Computing (UbiComp 2006)*, Springer, 2006, pp. 177–193 (pp. 8, 9, 117).
- [69] J. Huang, R. Kumar, M. Mitra, W.-J. Zhu and R. Zabih, 'Image indexing using color correlograms', in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 1997, pp. 762–768 (p. 61).
- [70] M. Huang, A. B. Mahajan and D. F. DeMenthon, 'Automatic performance evaluation for video summarization', Maryland University College Park Institution for Advanced Computer Studies, ADA448064, 2004 (pp. 23, 41).
- [71] P. Isola, J. Xiao, D. Parikh, A. Torralba and A. Oliva, 'What makes a photograph memorable?', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1469–1482, 2014 (p. 79).
- [72] N. Jankowski and M. Grochowski, 'Comparison of instances selection algorithms i. algorithms survey', in *International Conference on*

Artificial Intelligence and Soft Computing (ICAISC 2004), Springer, 2004, pp. 598–603 (p. 76).

- [73] H. Jegou, F. Perronnin, M. Douze, J. Sánchez, P. Perez and C. Schmid, ‘Aggregating local image descriptors into compact codes’, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 9, pp. 1704–1716, 2012 (p. 65).
- [74] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama and T. Darrell, ‘Caffe: Convolutional architecture for fast feature embedding’, in *Proceedings of the 22nd ACM International Conference on Multimedia*, ACM, 2014, pp. 675–678 (pp. 67, 131).
- [75] R. M. Jiang, A. H. Sadka and D. Crookes, ‘Hierarchical video summarization in reference subspace’, *IEEE Transactions on Consumer Electronics*, vol. 55, no. 3, 2009 (p. 15).
- [76] A. Jinda-Apiraksa, J. Machajdik and R. Sablatnig, *A keyframe selection of lifelog image sequences*, Erasmus Mundus M. Sc, 2012 (pp. 14, 15).
- [77] —, ‘A keyframe selection of lifelog image sequences.’, in *In proceedings of the IAPR conference on Machine Vision Applications (IAPR MVA 2013)*, Kyoto, 2013, pp. 33–36 (pp. 23, 26–28, 32, 56, 67).
- [78] N. Jojic, A. Perina and V. Murino, ‘Structural epitome: A way to summarize one’s visual experience’, in *Advances in Neural Information Processing Systems (NIPS 2010)*, 2010, pp. 1027–1035 (p. 56).
- [79] H. Kagaya and K. Aizawa, ‘Highly accurate food/non-food image classification based on a deep convolutional neural network’, in *International Conference on Image Analysis and Processing (ICIAP 15)*, Springer, 2015, pp. 350–357 (p. 105).
- [80] H.-B. Kang, ‘Video abstraction techniques for a digital library’, in *Distributed multimedia databases: techniques & applications*, Idea Group Publishing, 2002, pp. 120–132 (p. 13).
- [81] S. Kannappan, Y. Liu and B. Tiddeman, ‘A pertinent evaluation of automatic video summary’, in *23rd International Conference on Pattern Recognition (ICPR 2016)*, IEEE, 2016, pp. 2240–2245 (pp. 23, 26–28, 35, 37, 49).
- [82] E. Kasutani and A. Yamada, ‘The MPEG-7 color layout descriptor: A compact image feature description for high-speed image/video

- segment retrieval', in *Proceedings 2001 International Conference on Image Processing (ICIP 01)*, IEEE, vol. 1, Thessaloniki, Greece, Greece, Oct. 2001, pp. 674–677 (p. 130).
- [83] J. Kerr, S. J. Marshall, S. Godbole, J. Chen, A. Legge, A. R. Doherty, P. Kelly, M. Oliver, H. M. Badland and C. Foster, 'Using the sensecam to improve classifications of sedentary behavior in free-living settings', *American Journal of Preventive Medicine*, vol. 44, no. 3, pp. 290–296, 2013 (pp. 8, 10, 117).
- [84] A. Khosla, R. Hamid, C.-J. Lin and N. Sundaresan, 'Large-scale video summarization using web-image priors', in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2013, pp. 2698–2705 (pp. 14–16, 23, 24, 26–28, 32, 34, 56, 58, 74–76, 102).
- [85] A. Krizhevsky, I. Sutskever and G. E. Hinton, 'Imagenet classification with deep convolutional neural networks', in *Advances in neural information processing systems (NIPS 2012)*, 2012, pp. 1097–1105 (p. 67).
- [86] C. A. Latkin and A. R. Knowlton, 'Social network assessments and interventions for health behavior change: A critical review', *Behavioral Medicine*, vol. 41, no. 3, pp. 90–97, 2015 (p. 10).
- [87] H. V. Le, S. Clinch, C. Sas, T. Dingler, N. Henze and N. Davies, 'Impact of video summary viewing on episodic memory recall: Design guidelines for video summarizations', in *Proceedings of the 2016 CHI conference on human factors in computing systems*, ACM, San Jose, USA, 2016, pp. 4793–4805 (pp. 9, 103, 117, 153).
- [88] H.-C. Lee and S.-D. Kim, 'Rate-driven key frame selection using temporal variation of visual content', *Electronics Letters*, vol. 38, no. 5, pp. 217–218, 2002 (p. 16).
- [89] —, 'Iterative key frame selection in the rate-constraint environment', *Signal Processing: Image Communication*, vol. 18, no. 1, pp. 1–15, 2003 (p. 16).
- [90] M. L. Lee and A. K. Dey, 'Lifelogging memory appliance for people with episodic memory impairment', in *Proceedings of the 10th International Conference on Ubiquitous Computing*, ser. UbiComp '08, Seoul, Korea:

- ACM, 2008, pp. 44–53. doi: 10 . 1145 / 1409635 . 1409643. [Online]. Available: <http://doi.acm.org/10.1145/1409635.1409643> (p. 117).
- [91] 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 (CVPR 2012)*, IEEE, 2012, pp. 1346–1353 (pp. 60, 66, 86, 91, 108, 110).
- [92] Y. J. Lee and K. Grauman, ‘Predicting important objects for egocentric video summarization’, *International Journal of Computer Vision*, vol. 114, no. 1, pp. 38–55, 2015, issn: 1573-1405. doi: 10 . 1007 / s11263-014-0794-5. [Online]. Available: <https://doi.org/10.1007/s11263-014-0794-5> (pp. 11, 14, 16, 23, 56, 60, 74–76, 102).
- [93] Y. Li, L. Wang, T. Yang and B. Gong, ‘How local is the local diversity? reinforcing sequential determinantal point processes with dynamic ground sets for supervised video summarization’, in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 151–167 (p. 1).
- [94] Y. Li and B. Merialdo, ‘Vert: Automatic evaluation of video summaries’, in *Proceedings of the 18th ACM international conference on Multimedia*, ACM, 2010, pp. 851–854 (p. 27).
- [95] S. W. Lichtman, K. Pisarska, E. R. Berman, M. Pestone, H. Dowling, E. Offenbacher, H. Weisel, S. Heshka, D. E. Matthews and S. B. Heymsfield, ‘Discrepancy between self-reported and actual caloric intake and exercise in obese subjects’, *New England Journal of Medicine*, vol. 327, no. 27, pp. 1893–1898, 1992 (p. 9).
- [96] A. Lidon, M. Bolaños, M. Dimiccoli, P. Radeva, M. Garolera and X. Giro-i-Nieto, ‘Semantic summarization of egocentric photo stream events’, in *Proceedings of the 2nd Workshop on Lifelogging Tools and Applications*, ACM, Mountain View, California, USA, 2017, pp. 3–11 (pp. 11, 14, 16, 23, 25, 56, 60, 74, 75, 102).
- [97] C.-Y. Lin, ‘Rouge: A package for automatic evaluation of summaries’, *Text Summarization Branches Out*, 2004 (p. 29).
- [98] C. Liu, ‘Beyond pixels: Exploring new representations and applications for motion analysis’, Dept. of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, USA, 2009 (p. 12).

- [99] C. Liu, J. Yuen and A. Torralba, 'Sift flow: Dense correspondence across scenes and its applications', *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 5, pp. 978–994, 2011 (p. 26).
- [100] G. Liu, X. Wen, W. Zheng and P. He, 'Shot boundary detection and keyframe extraction based on scale invariant feature transform', in *Eighth IEEE/ACIS International Conference on Computer and Information Science*, IEEE, Jun. 2009, pp. 1126–1130 (pp. 32, 60).
- [101] T. Liu, H.-J. Zhang and F. Qi, 'A novel video key-frame-extraction algorithm based on perceived motion energy model', *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 10, pp. 1006–1013, 2003 (p. 60).
- [102] D. G. Lowe, 'Distinctive image features from scale-invariant keypoints', *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004 (p. 65).
- [103] Z. Lu and K. Grauman, 'Story-driven summarization for egocentric video', in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2013, pp. 2714–2721 (pp. 12, 14, 16, 23, 60, 74–76, 102).
- [104] M. Lux and O. Marques, 'Visual information retrieval using java and LIRE', *Synthesis Lectures on Information Concepts, Retrieval, and Services*, vol. 5, no. 1, pp. 1–112, 2013 (p. 65).
- [105] K. M. Mahmoud, 'An enhanced method for evaluating automatic video summaries', *arXiv preprint arXiv:1401.3590*, 2014 (pp. 34, 56).
- [106] K. M. Mahmoud, M. A. Ismail and N. M. Ghanem, 'VSCAN: An enhanced video summarization using density-based spatial clustering', in *International conference on image analysis and processing (ICIAP 2013)*, Springer, 2013, pp. 733–742 (p. 15).
- [107] K. Mahmoud, N. Ghanem and M. Ismail, 'Vgraph: An effective approach for generating static video summaries', in *Proceedings of the IEEE International Conference on Computer Vision (ICCV) Workshops*, 2013, pp. 811–818 (pp. 23, 26–28, 32, 34, 37).
- [108] B. S. Manjunath, J. .-. Ohm, V. V. Vasudevan and A. Yamada, 'Colour and texture descriptors', *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 6, pp. 703–715, 2001 (pp. 62, 63).

- [109] B. S. Manjunath and W.-Y. Ma, 'Texture features for browsing and retrieval of image data', *IEEE Transactions on pattern analysis and machine intelligence*, vol. 18, no. 8, pp. 837–842, 1996 (p. 64).
- [110] A. L. Marshall, J. N. Rachele, J. M. Lee-anne, J. Lai and L. V. Jones, 'Sit versus stand: Can sitting be accurately identified using MTI accelerometer data?: 2006', *Medicine & Science in Sports & Exercise*, vol. 42, no. 5, p. 475, 2010 (p. 9).
- [111] B. W. Matthews, 'Comparison of the predicted and observed secondary structure of t4 phage lysozyme', *Biochimica et Biophysica Acta (BBA)-Protein Structure*, vol. 405, no. 2, pp. 442–451, 1975 (p. 146).
- [112] S. Mei, G. Guan, Z. Wang, S. Wan, M. He and D. D. Feng, 'Video summarization via minimum sparse reconstruction', *Pattern Recognition*, vol. 48, no. 2, pp. 522–533, 2015 (pp. 14, 16, 23, 27, 32, 49, 56, 76, 118, 120, 122, 126, 146).
- [113] G. A. Miller, 'WordNet: A lexical database for english', *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995 (p. 105).
- [114] G. Miller and C. Fellbaum, *Wordnet: An electronic lexical database*, 1998 (p. 105).
- [115] A. G. del Molino, C. Tan, J.-H. Lim and A.-H. Tan, 'Summarization of egocentric videos: A comprehensive survey', *IEEE Transactions on Human-Machine Systems*, vol. 47, no. 1, pp. 65–76, 2017 (pp. 1, 13, 18, 19, 25, 86, 154).
- [116] A. G. del Molino, B. Mandal, L. Li and L. J. Hwee, 'Organizing and retrieving episodic memories from first person view', in *IEEE International Conference on Multimedia & Expo Workshops (ICMEW 2015)*, IEEE, Jun. 2015, pp. 1–6 (p. 101).
- [117] D. N. Monekosso and P. Remagnino, 'Behavior analysis for assisted living', *IEEE Transactions on Automation science and Engineering*, vol. 7, no. 4, pp. 879–886, 2010. doi: 10.1109/TASE.2010.2049840 (p. 117).
- [118] A. G. Money and H. Agius, 'Video summarisation: A conceptual framework and survey of the state of the art', *Journal of Visual Communication and Image Representation*, vol. 19, no. 2, pp. 121–143, 2008 (p. 1).

- [119] P. Mundur, Y. Rao and Y. Yesha, 'Keyframe-based video summarization using delaunay clustering', *International Journal on Digital Libraries*, vol. 6, no. 2, pp. 219–232, 2006 (pp. 15, 40, 56).
- [120] A. Nagasaka, 'Automatic video indexing and full-video search for object appearances', in *Proc. IFIP 1992 2nd Working Conf. Visual Database Systems*, 1992 (p. 15).
- [121] M. Nishiyama, T. Okabe, I. Sato and Y. Sato, 'Aesthetic quality classification of photographs based on color harmony', in *Conference on Computer Vision and Pattern Recognition (CVPR 2011)*, IEEE, Jun. 2011, pp. 33–40 (p. 59).
- [122] G. O'Loughlin, S. J. Cullen, A. McGoldrick, S. O'Connor, R. Blain, S. O'Malley and G. D. Warrington, 'Using a wearable camera to increase the accuracy of dietary analysis', *American journal of preventive medicine*, vol. 44, no. 3, pp. 297–301, 2013 (pp. 9, 117).
- [123] Y.-I. Ohta, T. Kanade and T. Sakai, 'Color information for region segmentation', *Computer graphics and image processing*, vol. 13, no. 3, pp. 222–241, 1980 (p. 30).
- [124] T. Ojala and M. Pietikäinen, 'Unsupervised texture segmentation using feature distributions', *Pattern recognition*, vol. 32, no. 3, pp. 477–486, 1999 (p. 64).
- [125] T. Ojala, M. Pietikäinen and T. Mäenpää, 'Multiresolution gray-scale and rotation invariant texture classification with local binary patterns', *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 7, pp. 971–987, 2002 (pp. 64, 85).
- [126] A. Oliva and A. Torralba, 'Modeling the shape of the scene: A holistic representation of the spatial envelope', *International journal of computer vision*, vol. 42, no. 3, pp. 145–175, 2001 (pp. 62, 131).
- [127] M. Otani, Y. Nakashima, E. Rahtu, J. Heikkilä and N. Yokoya, 'Video summarization using deep semantic features', in *Asian Conference on Computer Vision (ACCV 2016)*, Springer, 2016, pp. 361–377 (pp. 15, 18, 56, 58, 60).
- [128] S.-H. Ou, C.-H. Lee, V. S. Somayazulu, Y.-K. Chen and S.-Y. Chien, 'On-line multi-view video summarization for wireless video sensor

- network', *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, no. 1, pp. 165–179, Feb. 2015 (pp. 14, 76, 118, 120, 123, 127, 146).
- [129] A. N. Papadopoulos and Y. Manolopoulos, *Nearest Neighbor Search:: A Database Perspective*. Springer Science & Business Media, 2006 (p. 76).
- [130] K. Pauly-Takacs, C. J. Moulin and E. J. Estlin, 'Sensecam as a rehabilitation tool in a child with anterograde amnesia', *Memory*, vol. 19, no. 7, pp. 705–712, 2011 (p. 8).
- [131] E. Pękalska, R. P. Duin and P. Paclík, 'Prototype selection for dissimilarity-based classifiers', *Pattern Recognition*, vol. 39, no. 2, pp. 189–208, 2006 (p. 76).
- [132] P. Piasek, K. Irving and A. F. Smeaton, 'Sensecam intervention based on cognitive stimulation therapy framework for early-stage dementia', in *5th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth) and Workshops*, IEEE, May 2011, pp. 522–525 (p. 8).
- [133] H. Pirsiavash and D. Ramanan, 'Detecting activities of daily living in first-person camera views', in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 12)*, IEEE, Providence, RI, USA, Jun. 2012, pp. 2847–2854. doi: 10.1109/CVPR.2012.6248010 (pp. 145, 154, 155).
- [134] D. Potapov, M. Douze, Z. Harchaoui and C. Schmid, 'Category-specific video summarization', in *European conference on computer vision (ECCV 2014)*, Springer, 2014, pp. 540–555 (p. 16).
- [135] G. L. Priya and S. Domnic, 'Shot based keyframe extraction for ecological video indexing and retrieval', *Ecological Informatics*, vol. 23, pp. 107–117, 2014 (pp. 25, 56, 60).
- [136] F. Ragusa, V. Tomaselli, A. Furnari, S. Battiato and G. M. Farinella, 'Food vs non-food classification', in *Proceedings of the 2nd International Workshop on Multimedia Assisted Dietary Management*, ACM, 2016, pp. 77–81 (p. 105).
- [137] Z. Rasheed and M. Shah, 'Scene detection in hollywood movies and tv shows', in *Proceedings IEEE Computer Society Conference on*

- Computer Vision and Pattern Recognition*, vol. 2, Madison, WI, Jun. 2003, pp. 343–343 (pp. 14, 15, 76, 118, 120, 123, 127).
- [138] P. Ratsamee, Y. Mae, A. Jinda-Apiraksa, M. Horade, K. Kamiyama, M. Kojima and T. Arai, 'Keyframe selection framework based on visual and excitement features for lifelog image sequences', *International Journal of Social Robotics*, vol. 7, no. 5, pp. 859–874, 2015 (pp. 56, 67).
 - [139] H. Sakoe and S. Chiba, 'Dynamic programming algorithm optimization for spoken word recognition', *IEEE transactions on acoustics, speech, and signal processing*, vol. 26, no. 1, pp. 43–49, 1978 (p. 37).
 - [140] J. F. Sallis and B. E. Saelens, 'Assessment of physical activity by self-report: Status, limitations, and future directions', *Research quarterly for exercise and sport*, vol. 71, no. sup2, pp. 1–14, 2000 (p. 9).
 - [141] K. van de Sande, T. Gevers and C. Snoek, 'Evaluating color descriptors for object and scene recognition', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1582–1596, Sep. 2010. doi: 10.1109/TPAMI.2009.154 (p. 62).
 - [142] K. Schwarz, P. Wieschollek and H. P. Lensch, 'Will people like your image? learning the aesthetic space', in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, IEEE, Mar. 2018, pp. 2048–2057 (p. 59).
 - [143] G. Shakhnarovich, T. Darrell and P. Indyk, *Nearest-neighbor methods in learning and vision: theory and practice (neural information processing)*. The MIT press, 2006 (p. 76).
 - [144] A. Sharghi, B. Gong and M. Shah, 'Query-focused extractive video summarization', in *Proceedings of the European Conference on Computer Vision (ECCV 16)*, Springer, 2016, pp. 3–19 (pp. 14, 17, 74, 102).
 - [145] A. Sharghi, J. S. Laurel and B. Gong, 'Query-focused video summarization: Dataset, evaluation, and a memory network based approach', in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 17)*, Jul. 2017, pp. 2127–2136 (pp. 14, 17–19, 60, 74, 101, 102).

- [146] W. A. Shewhart, *Economic control of quality of manufactured product*. Van Nostrand Company, 1931 (pp. 124, 152).
- [147] K. Simonyan and A. Zisserman, 'Very deep convolutional networks for large-scale image recognition', *arXiv preprint arXiv:1409.1556*, 2014 (pp. 31, 65, 131).
- [148] A. Singla, L. Yuan and T. Ebrahimi, 'Food/non-food image classification and food categorization using pre-trained googlenet model', in *Proceedings of the 2nd International Workshop on Multimedia Assisted Dietary Management*, ACM, 2016, pp. 3–11 (p. 105).
- [149] M. Song and H. Wang, 'Highly efficient incremental estimation of gaussian mixture models for online data stream clustering', in *SPIE 5803, Intelligent Computing: Theory and Applications III*, vol. 5803, Mar. 2005, pp. 174–184 (pp. 76, 120, 121, 123, 127).
- [150] T. T. de Souza Barbieri and R. Goularte, 'KS-SIFT: A keyframe extraction method based on local features', in *IEEE International Symposium on Multimedia*, IEEE, Dec. 2014, pp. 13–17 (p. 60).
- [151] A. Spector, L. Thorgrimsen, B. Woods, L. Royan, S. Davies, M. Butterworth and M. Orrell, 'Efficacy of an evidence-based cognitive stimulation therapy programme for people with dementia: Randomised controlled trial', *The British Journal of Psychiatry*, vol. 183, no. 3, pp. 248–254, 2003 (p. 8).
- [152] E. Spyrou, G. Tolia, P. Mylonas and Y. Avrithis, 'Concept detection and keyframe extraction using a visual thesaurus', *Multimedia Tools and Applications*, vol. 41, no. 3, pp. 337–373, 2009 (pp. 56, 60).
- [153] X. Sun and M. S. Kankanhalli, 'Video summarization using r-sequences', *Real-time imaging*, vol. 6, no. 6, pp. 449–459, 2000 (p. 60).
- [154] M. J. Swain and D. H. Ballard, 'Color indexing', *International journal of computer vision*, vol. 7, no. 1, pp. 11–32, 1991 (p. 63).
- [155] E. Talavera, M. Dimiccoli, M. Bolaños, 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 (p. 12).

- [156] H. Tamura, S. Mori and T. Yamawaki, 'Textural features corresponding to visual perception', *IEEE Transactions on Systems, man, and cybernetics*, vol. 8, no. 6, pp. 460–473, 1978 (p. 64).
- [157] I. Triguero, J. Derrac, S. Garcia and F. Herrera, 'A taxonomy and experimental study on prototype generation for nearest neighbor classification', *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 1, pp. 86–100, 2012 (p. 77).
- [158] B. T. Truong and S. Venkatesh, 'Video abstraction: A systematic review and classification', *ACM transactions on multimedia computing, communications, and applications (TOMM)*, vol. 3, no. 1, p. 3, 2007 (pp. 12, 15, 21, 24, 25, 76, 118, 120, 124, 128).
- [159] D. Umberson and J. Karas Montez, 'Social relationships and health: A flashpoint for health policy', *Journal of health and social behavior*, vol. 51, no. 1 suppl. S54–S66, 2010 (p. 10).
- [160] V. Valdés and J. M. Martínez, 'On-line video abstract generation of multimedia news', *Multimedia Tools and Applications*, vol. 59, no. 3, pp. 795–832, Aug. 2012 (pp. 14, 76).
- [161] P. Varini, G. Serra and R. Cucchiara, 'Egocentric video summarization of cultural tour based on user preferences', in *Proceedings of the 23rd ACM international conference on Multimedia*, ser. MM '15, ACM, Brisbane, Australia, 2015, pp. 931–934 (pp. 12, 23, 60).
- [162] —, 'Personalized egocentric video summarization of cultural tour on user preferences input', *IEEE Transactions on Multimedia*, vol. 19, no. 12, pp. 2832–2845, 2017 (pp. 14, 16, 18, 19, 60, 74, 101, 102).
- [163] A. Vedaldi and K. Lenc, 'Matconvnet – convolutional neural networks for matlab', in *Proceeding of the ACM Int. Conf. on Multimedia*, 2015 (p. 66).
- [164] J. Vermaak, P. Pérez, M. Gangnet and A. Blake, 'Rapid summarisation and browsing of video sequences.', in *Proceedings of the British Machine Vision Conference (BMVC 2002)*, Citeseer, 2002, pp. 1–10 (pp. 15, 30, 60).
- [165] M. Wang, R. Hong, G. Li, Z.-J. Zha, S. Yan and T.-S. Chua, 'Event driven web video summarization by tag localization and key-shot

- identification', *IEEE Transactions on Multimedia*, vol. 14, no. 4, pp. 975–985, 2012 (p. 60).
- [166] D. B. West *et al.*, *Introduction to Graph Theory*. Upper Saddle River: Prentice Hall, 2001, vol. 2 (p. 35).
- [167] D. R. Wilson and T. R. Martinez, 'Reduction techniques for instance-based learning algorithms', *Machine learning*, vol. 38, no. 3, pp. 257–286, 2000 (pp. 76, 77).
- [168] D. L. Wilson, 'Asymptotic properties of nearest neighbor rules using edited data', *IEEE Transactions on Systems, Man, and Cybernetics*, no. 3, pp. 408–421, 1972 (p. 77).
- [169] W. Wolf, 'Key frame selection by motion analysis', in *IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, IEEE, vol. 2, May 1996, pp. 1228–1231 (p. 16).
- [170] E. Woodberry, G. Browne, S. Hodges, P. Watson, N. Kapur and K. Woodberry, 'The use of a wearable camera improves autobiographical memory in patients with alzheimer's disease', *Memory*, vol. 23, no. 3, pp. 340–349, 2015. [Online]. Available: <https://doi.org/10.1080/09658211.2014.886703> (p. 117).
- [171] J. Wu and J. M. Rehg, 'CENTRIST: A visual descriptor for scene categorization', *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 33, no. 8, pp. 1489–1501, 2011 (p. 130).
- [172] B. Xiong and K. Grauman, 'Detecting snap points in egocentric video with a web photo prior', in *European conference on computer vision (ECCV 2014)*, Springer, 2014, pp. 282–298 (pp. 11, 56, 79).
- [173] 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 (CVPR 15)*, 2015, pp. 4525–4533 (pp. 14, 15, 18, 19, 60, 74, 101, 102).
- [174] J. Xu, L. Mukherjee, Y. Li, J. Warner, J. M. Rehg and V. Singh, 'Gaze-enabled egocentric video summarization via constrained submodular maximization', in *Proceedings of the IEEE The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 2235–2244 (pp. 14, 17, 19, 24, 27, 74, 75).

- [175] T. Yao, T. Mei and Y. Rui, 'Highlight detection with pairwise deep ranking for first-person video summarization', in *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR 2016)*, Jun. 2016, pp. 982–990 (pp. 14, 16, 23, 24, 27, 74, 75, 102).
- [176] M. M. Yeung and B. Liu, 'Efficient matching and clustering of video shots', presented at the Proceedings. International Conference on Image Processing, IEEE, vol. 1, Washington, DC, USA, Oct. 1995, pp. 338–341 (p. 15).
- [177] S. Yeung, A. Fathi and L. Fei-Fei, 'Videoset: Video summary evaluation through text', *arXiv preprint arXiv:1406.5824*, 2014 (pp. 28, 154, 155).
- [178] X.-D. Yu, L. Wang, Q. Tian and P. Xue, 'Multi-level video representation with application to keyframe extraction', in *Proceedings 10th International Multimedia Modelling Conference.*, IEEE, Jan. 2004, pp. 117–123 (pp. 15, 56, 60).
- [179] Y. Yuan, T. Mei, P. Cui and W. Zhu, 'Video summarization by learning deep side semantic embedding', *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 1, pp. 226–237, 2017 (p. 58).
- [180] 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*, vol. 24, no. 9-10, pp. 1523–1532, 2003. doi: doi.org/10.1016/S0167-8655(02)00391-4 (p. 15).
- [181] K. Zhang, W.-L. Chao, F. Sha and K. Grauman, 'Video summarization with long short-term memory', in *European conference on computer vision (ECCV 2016)*, Springer, 2016, pp. 766–782 (p. 16).
- [182] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba and A. Oliva, 'Learning deep features for scene recognition using places database', in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence and K. Q. Weinberger, Eds., Curran Associates, Inc., 2014, pp. 487–495 (p. 131).
- [183] X. Zhu, X. Wu, J. Fan, A. K. Elmagarmid and W. G. Aref, 'Exploring video content structure for hierarchical summarization', *Multimedia Systems*, vol. 10, no. 2, pp. 98–115, 2004 (p. 56).

- [184] Y. Zhuang, Y. Rui, T. S. Huang and S. Mehrotra, 'Adaptive key frame extraction using unsupervised clustering', in *Proceedings 1998 International Conference on Image Processing. ICIP98 (Cat. No. 98CB36269)*, IEEE, vol. 1, 1998, pp. 866–870 (pp. 15, 56).