

***HYPERLAPSE*** SEMÂNTICO PARA VÍDEOS EM  
PRIMEIRA PESSOA: UMA ABORDAGEM  
MULTI-IMPORTÂNCIA BASEADA EM  
CODIFICAÇÃO ESPARSA



MICHEL MELO DA SILVA

***HYPERLAPSE*** SEMÂNTICO PARA VÍDEOS EM  
PRIMEIRA PESSOA: UMA ABORDAGEM  
MULTI-IMPORTÂNCIA BASEADA EM  
CODIFICAÇÃO ESPARSA

Tese apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do grau de Doutor em Ciência da Computação.

ORIENTADOR: ERICKSON RANGEL DO NASCIMENTO  
COORIENTADOR: MARIO FERNANDO MONTENEGRO CAMPOS

Belo Horizonte  
Setembro de 2019



MICHEL MELO DA SILVA

**SEMANTIC HYPERLAPSE: A SPARSE CODING  
BASED AND MULTI-IMPORTANCE APPROACH  
FOR FIRST-PERSON VIDEOS**

Thesis presented to the Graduate Program  
in Computer Science of the Universidade  
Federal de Minas Gerais in partial fulfill-  
ment of the requirements for the degree of  
Doctor in Computer Science.

ADVISOR: ERICKSON RANGEL DO NASCIMENTO  
CO-ADVISOR: MARIO FERNANDO MONTENEGRO CAMPOS

Belo Horizonte  
September 2019

© 2019, Michel Melo da Silva.  
Todos os direitos reservados

**Ficha catalográfica elaborada pela Biblioteca do ICEx - UFMG**

Silva, Michel Melo da.

S586s Semantic hyperlapse: a sparse coding based and multi-importance approach for first-person videos / Michel Melo da Silva — Belo Horizonte, 2019.  
xxxii, 84 p.: il.; 29 cm.

Tese (doutorado) - Universidade Federal de Minas Gerais – Departamento de Ciência da Computação.

Orientador: Erickson Rangel do Nascimento  
Coorientador: Mário Fernando Montenegro Campos

1. Computação – Teses. 2. Visão por computador.  
3. Sistemas Multimídia. 4. Semântica – Processamento de dados. I. Orientador. II. Coorientador. III. Título.

CDU 519.6\*82.10(043)



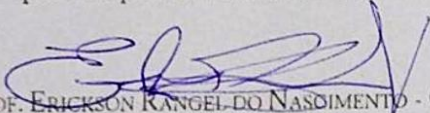
UNIVERSIDADE FEDERAL DE MINAS GERAIS  
INSTITUTO DE CIÊNCIAS EXATAS  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

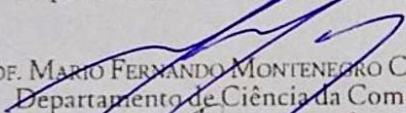
## FOLHA DE APROVAÇÃO

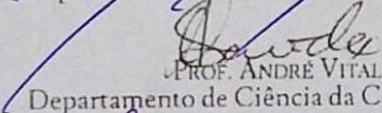
Semantic Hyperlapse: a Sparse Coding based and Multi-Importance  
Approach for First-Person Videos

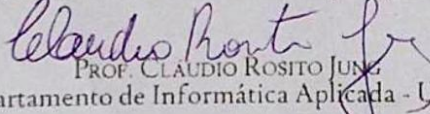
**MICHEL MELO DA SILVA**

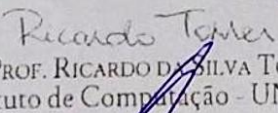
Tese defendida e aprovada pela banca examinadora constituída pelos Senhores:

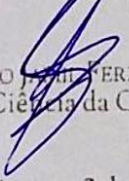
  
PROF. ERICKSON KANGEL DO NASCIMENTO - Orientador  
Departamento de Ciência da Computação - UFMG

  
PROF. MARIO FERNANDO MONTENEGRO CAMPOS - Coorientador  
Departamento de Ciência da Computação - UFMG

  
PROF. ANDRÉ VITAL SAÚDE  
Departamento de Ciência da Computação - UFLA

  
PROF. CLAUDIO ROSITO JUNG  
Departamento de Informática Aplicada - UFRGS

  
PROF. RICARDO DA SILVA TORRES  
Instituto de Computação - UNICAMP

  
PROF. SILVIO JOSÉ FERZOLI GUIMARÃES  
Departamento de Ciência da Computação - PUC/MG

Belo Horizonte, 2 de Julho de 2019.





*Dedico este trabalho, bem como todo o tempo e esforços empenhados para a conclusão do mesmo, à minha família e amigos que ombrearam junto a mim nessa jornada. Em especial, dedico este marco à minha mãe Odilia Antônia de Melo Silva, meu pai Everardo Antônio Silva, meu irmão Maykel Melo Silva e minha noiva Taciany da Silva Pereira que juntos comemoraram as minhas vitórias e suportaram os momentos turbulentos.*



# Agradecimentos

Este trabalho não seria o mesmo se não fosse as muitas mãos que ajudaram a rumar ao resultado obtido. Mesmo sendo difícil listar todos os envolvidos e correndo o risco de pecar por não mencionar alguém em específico, não poderia deixar de citar certas pessoas que foram cruciais para a conclusão desta etapa. Inicialmente, agradeço ao meu orientador professor — Erickson R. Nascimento — pelo empenho na orientação, por todo o tempo dedicado às incontáveis e intermináveis reuniões, pelo esforço despendido para direcionar a pesquisa, pela preocupação em transformar o estudante em um profissional, e pelas palavras amigas. Estendo o agradecimento ao meu co-orientador — professor Mário F. M. Campos — que confiou na minha capacidade e por todo auxílio ao trabalho desenvolvido.

Agradeço aos colegas Washington, Felipe, João, Edson, Alan e a toda equipe *Semantic Hyperlapse* pela dedicação e empenho na execução dos trabalhos. Não esquecendo os membros do laboratório de Visão Computacional e Robótica (VeRLab), funcionários e técnicos do Departamento de Ciência da Computação (DCC) e do Programa de Pós-Graduação em Ciência da Computação (PPGCC) da Universidade Federal de Minas Gerais (UFMG). A qualidade do trabalho não seria a mesma sem o auxílio financeiro das agências CAPES e CNPq que possibilitou a minha dedicação exclusiva ao doutorado. Agradeço também à FAPEMIG e Petrobras por financiar o projeto de diferentes formas, como participação em conferência e compra de equipamentos.

Por fim, as últimas palavras dessa Seção são para agradecer à minha família — Odília (mãe), Everardo (pai) e Maykel (irmão) — pela compreensão em relação aos momentos em que não pude estar presente, pelo esforço na minha criação e por formar a base sólida que me faz persistir frente às provas da vida. Sou grato ao companheirismo da minha futura esposa Taciany que me confortou nos momentos árdusos, me acompanhou durante a calmaria, e fez mais feliz as comemorações das vitórias.



*“If I have seen further, it is by standing on the shoulders of giants.”*

(Isaac Newton, 1676)



# Resumo

O surgimento de câmeras pessoais portáteis de baixo custo, combinado com a alta qualidade dos sensores e a quase ilimitada capacidade de armazenamento em sites de compartilhamento de vídeos despertou um crescente interesse pelos vídeos em primeira pessoa. Tais vídeos são geralmente compostos de gravações de longa duração sem qualquer edição, capturadas por um dispositivo acoplado ao corpo do gravador, o que os tornam tediosos e visualmente desagradáveis de assistir. Com isso, surgiu a necessidade de prover acesso rápido à informação neles contida. Para suprir essa necessidade, esforços vem sendo aplicados para o desenvolvimento de técnicas como *Hyperlapse*, na qual o objetivo é acelerar o vídeo em primeira pessoa criando um vídeo reduzido visualmente agradável de se assistir, e *Hyperlapse Semântico*, que além de acelerar o vídeo, cria ênfase em trechos importantes, dado algum critério de semântica previamente definido. Contudo, o método estado da arte em *Hyperlapse Semântico*, *Semantic Fast-Forward and Stabilized Egocentric Video (FFSE)*, negligencia o grau de importância da informação relevante, considerando apenas se a mesma é importante ou não. Outras limitações do método FFSE são o número de parâmetros, a escalabilidade no número de características visuais, e a mudança brusca nos fatores de aceleração entre segmentos de vídeo consecutivos. Nesta tese, propomos uma metodologia livre de parâmetros baseada em Codificação Esparsa para acelerar vídeos em primeira pessoa de forma adaptativa e enfatizar as partes relevantes através de uma abordagem multi-importância. O uso da abordagem proposta resultou na criação de vídeos reduzidos mantendo uma maior quantidade de informação semântica, com menos transições bruscas nas taxas de aceleração, e mais suaves em relação ao resultado do método FFSE.

**Palavras-chave:** Vídeo em primeira pessoa, Aceleração semântica, Codificação esparsa, Problema da Reconstrução Mínima Esparsa.





# Abstract

The emergence of low-cost, high-quality personal wearable cameras combined with the unlimited storage capacity of video-sharing websites have evoked a growing interest in First-Person Videos. Such videos are usually composed of long-running unedited streams captured by a device attached to the user body, which makes them tedious and visually unpleasant to watch. Consequently, rise the need to provide quick access to the information therein. To address this need, efforts have been applied to the development of techniques such as Hyperlapse and Semantic Hyperlapse, which aims to create visually pleasant shorter videos and emphasize semantic portions of the video respectively. The state-of-the-art Semantic Hyperlapse method FFSE, negligees the level of importance of the relevant information, by only evaluating if it is significant or not. Other limitations of FFSE are the number of input parameters, the scalability in the number of visual features to describe the frames, the abrupt change in the speed-up rate of consecutive video segments. In this dissertation, we propose a parameter-free Sparse Coding based methodology to adaptively fast-forward First-Person Videos, that emphasize the semantic portions applying a multi-importance approach. Experimental evaluations show that the proposed method creates shorter version video retaining more semantic information, with fewer abrupt transitions of speed-up rates, and more stable final videos than the output of FFSE.

**Palavras-chave:** First-person Videos, Semantic Fast-forward, Sparse Coding, Minimum Sparse Reconstruction Problem.



# List of Figures

1.1	Wearable devices evolution along time. . . . .	1
1.2	Illustration of the Always-On operation mode and the wearable camera's first-person Point of View. . . . .	2
1.3	Illustration of the main reasons of the problems related to First-Person Video. . . . .	4
2.1	Video Fast-Forward methodologies steps. . . . .	14
3.1	Proposed Multi-Importance and Sparse Coding-based Semantic Hyperlapse Methodology. . . . .	22
3.2	<i>Ad hoc</i> Semantic Analysis. . . . .	24
3.3	<i>CoolNet</i> Convolutional Neural Network architecture. . . . .	25
3.4	Semantic Temporal Segmentation using Multi-Importance approach. . . . .	27
3.5	Example of the search space of the Speed-up Optimization function. . . . .	29
3.6	Limitation of current frame description approaches. . . . .	31
3.7	Sparse Sampling Methodology. . . . .	32
3.8	Proposed Smooth Frame Transitions Methodology. . . . .	35
3.9	Methodology of the Stabilization process for accelerated videos. . . . .	37
3.10	Possible cases after the application of homography transformations during the video stabilization. . . . .	39
4.1	Samples of the controlled Semantic Dataset. . . . .	42
4.2	Sample and labels of the Multimodal Dataset. . . . .	44
4.3	Graph-based Adaptive Frame Sampling approaches. . . . .	46
4.4	Visual instability result of the user study. . . . .	50
4.5	Analysis of accelerated videos regarding the amount of semantic information retained. . . . .	52
4.6	Semantic Profile curve of the <i>CoolNet</i> for every frame of a sample video. . . . .	54
4.7	Smoothing Speed-up transition by the Fill Gap processing. . . . .	55
4.8	Experimental evaluation regarding the Visual Instability. . . . .	56

4.9	Experimental evaluation regarding the Temporal Discontinuity. . . . .	57
4.10	Processing time analysis of related to the input video length of graph and sparse based approaches. . . . .	60
4.11	Comparison between weighted and non-weighted sparse-based frame sampling.	62
4.12	Comparison of abrupt camera detection using OF versus CDC. . . . .	63
4.13	Effect of the Smoothing Frame Transition step in the appearance cost profile of a video. . . . .	64
4.14	Improvement regarding the visual instability metric by applying the Video Stabilization process. . . . .	70

# List of Tables

2.1	Summarization of methodologies to accelerate videos. . . . .	20
4.1	Information about videos compositing the Semantic Dataset. . . . .	42
4.2	Information about videos compositing the Multimodal Dataset. . . . .	45
4.3	Comparison between the sparse and graph based approaches in the unconstrained Dataset of Multimodal Semantic Egocentric Videos (DoMSEV). . . . .	58
4.4	Continuation of the comparison between the sparse and graph based approaches in the unconstrained DoMSEV. . . . .	59
4.5	Average time processing per frame to perform the complete fast-forward pipelines. . . . .	61
4.6	Evaluation of the frame sampling modeling applying only the Smoothing Frame Transition (STF) step and applying the complete framework with Fill Gap step. . . . .	65
4.7	Evaluation of the frame sampling modeling by Locality-constrained Linear Coding (LLC) and regular sparse coding methods Orthogonal Matching Pursuit (OMP) and Lasso (SC). . . . .	67
4.8	Evaluation of the frame sampling describing the video frames through handcrafted features proposed in Section 3.2.1 against using Deep features (AlexNet layer <code>fc7</code> ). . . . .	68



# List of Algorithms

1	Sparse-based frame sampling Lambda value adjustment . . . . .	34
2	Egocentric Accelerated Video Stabilizer . . . . .	38





# List of Acronyms

**CNN** Convolutional Neural Network. 16, 22–24, 32, 51, 69

**DoMSEV** Dataset of Multimodal Semantic Egocentric Videos. xxi, 39, 41, 42, 55–57, 69

**ES** EgoSampling. 42, 50, 54

**FFSE** Semantic Fast-Forward and Stabilized Egocentric Video. xv, xvii, 43, 50, 54, 55, 68

**FOE** Focus of Expansion. 47

**FPS** Frames Per Second. 40, 46

**MSH** Microsoft Hyperlapse. 43, 50, 54, 55, 67

**PSO** Particle Swarm Optimization. 27, 45

**RMSE** Root Mean Square Error. 52, 54, 55, 69

**ROI** Region of Interest. 21, 22

**SVM** Support Vector Machine. 10, 14



# List of Symbols

- $a_k$  Area of the ROI  $k$  normalized by the area of the image. 22
- $\alpha$  activation vector on the sparse-sampling modeling. 29–32
- $A_{i,j}$  Term of Appearance related to the transition of the frame  $i$  to  $j$ . 44, 45
- $c$  Number of features used to represent the frame. 29
- $c_k$  Classifier Confidence about the ROI  $k$ . 22, 46
- $cp\%$  Percentage central of the image defining the crop area. 36, 46
- $D$  Sparse Sampling Dictionary  $\in \mathbb{R}^{c \times n}$ . 29, 30
- $\mathbf{d}$  Feature vector descriptor  $\in \mathbb{R}^c$  of a frame. 29, 32
- $\mathcal{D}$  Set of dropped frames. 36, 37
- $\Delta$  Distance in frames between the  $M_{pos}$  and  $M_{pre}$ . 35
- $\delta$  Distance in frames between the  $M_{pre}$  and frame  $i$ . 35
- $dp\%$  Percentage central of the image defining the drop area. 36, 46
- $\eta$  Term to avoid dividing by zero. 37, 46
- $F_d$  Speed-up rate required by user. 11, 25, 26, 33, 47, 49, 54, 64
- $\hat{f}_i$  Reconstruction of the  $i$ -th frame of the video. 35–37
- $F_{ns}$  Speed-up applied to the non-semantic segment. 26, 27
- $F_s$  Speed-up applied to the semantic segment. 26, 27
- $f_i$   $i$ -th frame of the video. 21, 33, 35, 37, 48

$w$  Frame weight. 30

$\gamma$  Length of the segment used in the stabilization process. 35, 46

$G_\sigma(x)$  Value of the Gaussian function centered at frame with standard deviation  $\sigma$  in the position  $x$ . 37

$H$  Frame height. 21

$H_{i_1, i_2}$  Matrix defining the homography transformation ( $H_{i_1, i_2}$ ) from images  $i_1$  to  $i_2$ . 35

$I_{i, j}$  Term of Instability related to the transition of the frame  $i$  to  $j$ . 44

$L_{ns}$  Length in frames of the non-semantic segment. 25–28

$L_s$  Length in frames of the semantic segment. 25–28

$\lambda$  Regularization parameters. 26, 27, 29–31, 44, 45

$M_{pos}$  Master frame immediately posterior to the frame. 35

$M_{pre}$  Master frame immediately previous to the frame. 35

$M_k$  Master frame of the  $k$ -th video segment. 35

$n$  Video length in number of frames. 29, 38, 48, 49

$p$  Covered percentage of the crop area. 37

$p_{ns}$  Percentage of Non-Semantic information of the input video. 28

$p_s$  Percentage of Semantic information of the input video. 26, 27

$R$  Number of inliners obtained with RANSAC. 37

$S$  Semantic Score. 21, 37, 45

$\mathcal{S}$  Set of optimal frames to compose the hyperlapse video. 29, 33–36, 38, 45

$S_{i, j}$  Term of Semantic related to the transition of the frame  $i$  to  $j$ . 44, 45

$\tau_{max}$  Number of sequential frames connect by a edge to the frame in the graph modeling step. 44–46

$V_{i,j}$  Term of Velocity related to the transition of the frame  $i$  to  $j$ . 44

$\mathbf{v}$  Video story representation  $\in \mathbb{R}^f$  used in the Sparse Coding Modeling. 29, 30

$w_{i,j}$  Weight of the graph edge connecting the node  $i$  to  $j$ . 44

$W$  Frame width. 21



# Contents

<b>Agradecimientos</b>	<b>xi</b>
<b>Resumo</b>	<b>xv</b>
<b>Abstract</b>	<b>xvii</b>
<b>List of Figures</b>	<b>xix</b>
<b>List of Tables</b>	<b>xxi</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Contextualization . . . . .	3
1.2 Problem Definition . . . . .	6
1.3 Dissertation Statements . . . . .	6
1.4 Contributions. . . . .	7
<b>2 Related Work</b>	<b>11</b>
2.1 Video Summarization . . . . .	12
2.2 Hyperlapse and Video Fast-Forward . . . . .	13
2.3 Semantic Hyperlapse and Fast-Forward . . . . .	16
<b>3 Methodology</b>	<b>21</b>
3.1 Definition of Semantic Segments . . . . .	21
3.1.1 Semantic Analysis . . . . .	23
3.1.2 Temporal Segmentation . . . . .	26
3.1.3 Speed-up estimation . . . . .	27
3.2 Adaptive Frame Sampling . . . . .	30
3.2.1 Sparse-based selection . . . . .	30
3.3 Output Video Producing . . . . .	36
3.3.1 Video Stabilization . . . . .	37

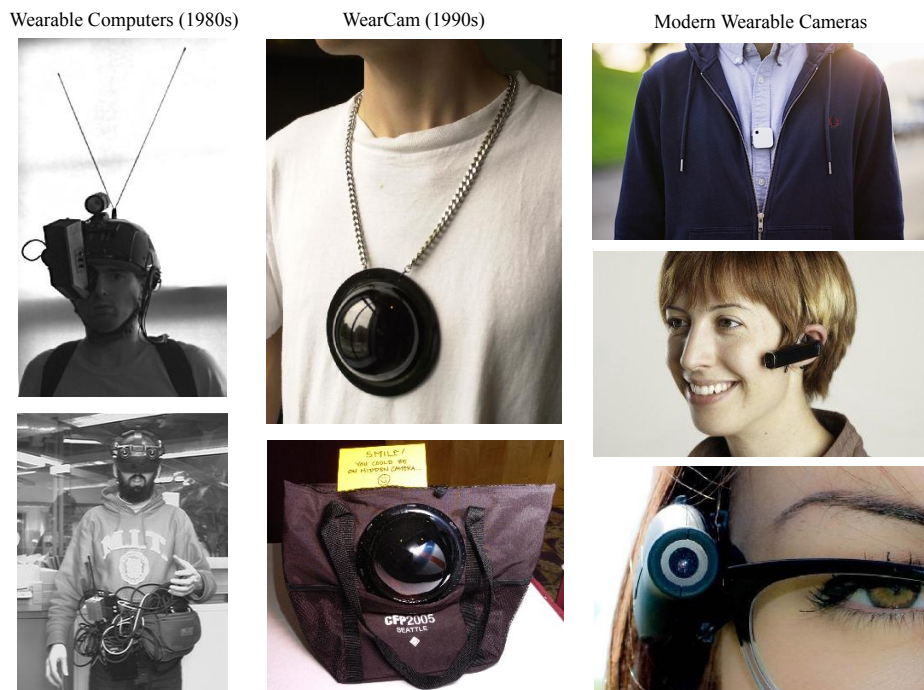
3.3.2	Video Compositing . . . . .	40
<b>4</b>	<b>Experiments</b>	<b>41</b>
4.1	Datasets . . . . .	41
4.1.1	Annotated Semantic Dataset . . . . .	41
4.1.2	Multimodal Semantic Egocentric Videos . . . . .	43
4.2	Competitors . . . . .	44
4.2.1	Graph-based selection . . . . .	46
4.3	Parameters Setup . . . . .	48
4.4	Evaluation criterion . . . . .	49
4.5	Results . . . . .	51
4.5.1	Multi Importance Semantic Analysis . . . . .	51
4.5.2	Semantic based on users' opinion . . . . .	53
4.5.3	Smoothing of Speed-up transitions . . . . .	54
4.5.4	Sparse-based Semantic Hyperlapse . . . . .	55
4.5.5	Processing Time . . . . .	57
4.6	Ablation Study . . . . .	61
4.6.1	Weighted Sparse Sampling . . . . .	62
4.6.2	Detection of abrupt camera movement . . . . .	63
4.6.3	Smooth Frame Transition . . . . .	64
4.6.4	Filling Gap Between Segments . . . . .	65
4.6.5	Comparison between Sparse Coding formulations . . . . .	66
4.6.6	Feature Scalability . . . . .	68
4.6.7	Video Stabilization . . . . .	69
4.7	Concluding Remarks . . . . .	70
<b>5</b>	<b>Conclusions</b>	<b>73</b>
5.1	Limitations . . . . .	74
5.2	Future works . . . . .	74
	<b>Bibliography</b>	<b>77</b>



# Chapter 1

## Introduction

The WearCam, introduced in the nineties, is the precursor of what we know as wearable cameras [Mann, 1998]. Despite being a functional product, the general public looked at the WearCam and all other wearable devices as function-less mechanisms with the strict purpose of acquiring specific data. Over the last couple of decades, technological advances in integrated circuits technology dropped the cost and the power consump-



**Figure 1.1.** Wearable devices evolution along time. Left column: Steve Mann, known as the first cyborg, wearing his computers in the 1980s. Middle column: WearCam introduced in the 1990s and referenced on the 15th annual conference on Computers, Freedom & Privacy 2005. Right Column: modern wearable cameras.



**Figure 1.2.** Illustration of the Always-On operation mode and the wearable camera's first-person Point of View.

tion of high-definition sensors and high-performance processors. Another advantage of these new devices is regarding the increase in the capacity of memories. Thanks to these technological advances, wearable cameras are no longer particularly designed for research or industry. Figure 1.1 depicts the evolutionary process of such devices to the general public. Cameras such as GoPro™, Narrative Clip, Looxcie, Google Glass becomes a successful worldwide product.

Differently from tape recorders, creating videos using modern wearable devices is cost-less, since the user has no concern about photographic films or cassette tapes. Moreover, the user interaction with these devices is the most distinct characteristic. Hand-held cameras bound the user capacity to interact with objects and perform actions, while fixed cameras limit the capturing area. Handling the current wearable devices is simple as pointing the camera and click the shutter button. Also, since the device is attached to the user body, it keeps the hands free to interact with objects or perform any action. Hereinafter we refer to this operation mode as Always-On. Usually, the attaching point is on the head or chest of the user, capturing unprecedented long-running Point of View footages as depicted in Figure 1.2.

Statistics about Internet usage in 2017 announce that online videos represented 70% of global traffic. Recent studies predict that this number will strike 80% by 2022 [Traffic-Inquiries, 2018]. Not only are Internet users watching more online video, but they are also recording themselves and producing a growing number of videos for sharing their day-to-day life routine. As pointed by del Molino et al. [2016], due to the number of videos and their length, the recorder may never pay attention to the

majority of recorded moments. Even highly significant moments will be lost along with everyday activities which do not merit recording.

Therefore, this massive increase in the amount of data makes arise the need for organizing and providing quick access to the information in First-Person Videos. Nowadays, this problem is on the verge of magnifying, with the technological advances in energy consumption and data storage capabilities, soon there will be cameras running all day. Manually handling such amount of information will be impractical.

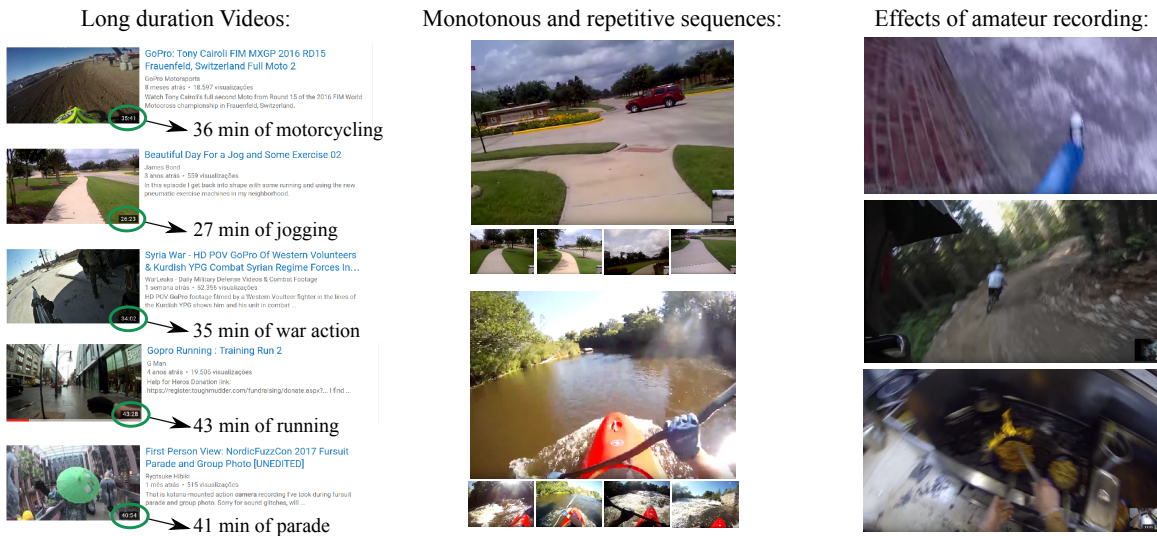
From the MylifeBits [Gemmell et al., 2002] project in early 2000 until today, the processing of video data remains as one of the most challenging tasks for lifelogging. Tasks such as acquisition, storage, and the usage of a large amount of data are particularly hard for video processing. Dealing with First-Person Videos is even more challenge, once methods developed to Third-Person Videos cannot be successfully applied to First-Person Videos [del Molino et al., 2016]. The unconcerned recording introduces side effects, such as poor illumination, jerky transitions, and blurred images, which are not treated by Third-Person Video algorithms.

Some Third-Person Video solutions were fine-tuned or completely reworked to address First-Person Video challenges. Besides good results reported to specific domains, those methods do not hold the high accuracy when applied in the wild. Examples of these solutions are: Social Interaction [Fathi et al., 2012a; Yang et al., 2016; Lee et al., 2012], Video Summarization [Potapov et al., 2014; Lu and Grauman, 2013; Gygli et al., 2014; Zhang et al., 2016], Gaze estimation [Xu et al., 2015; Fathi et al., 2012b], Video Fast-Forward [Okamoto and Yanai, 2014; Higuchi et al., 2017], visualization of 360° videos in normal field-of-view [Su et al., 2016].

## 1.1 Contextualization

The ubiquity of inexpensive shoot video devices, along with the lower costs of producing and storing videos are giving unprecedented freedom to the people to create increasingly long-running first-person videos. On the other hand, such freedom might lead the user to create lengthy and tedious videos, which are hard to watch in their entirety. Making a more in-depth analysis of the First-Person Video problems, we summarize the top-3 issues as: video length, monotonicity, and the non-use of professional practices and equipment (Figure 1.3).

Video length and the monotonicity of First-Person Video are related to the Always-On mode. By pressing the shutter button, the camera records every moment of the user activity, even the more unattractive actions, *e.g.*, to tie shoelaces before



**Figure 1.3.** Illustration of the problems related to First-Person Video (zooming to details). Left column: Videos with long duration. Middle column: monotonous videos, the thumbnails are all visually similar to the current frames. Right column: blur images and jerky scene transitions due to non-usage of professional equipment and techniques.

running, or the coach instructions before boxing. Moreover, it keeps recording until the button is pressed again, which is usually done when the activity ends. Daily activities are typically lengthy and repetitive; hence the associated video is interminable and tedious. Moving to the third listed problem, the non-use of professional equipment and techniques is more related to technical problems. The lack of proper illumination condition, worry-less camera handling, and non-usage of stabilizer solution create videos composed of blurred images and jerky scene transitions. Figure 1.3 presents examples of the cited problems.

The problems mentioned above lead to major issues: visual discomfort for the watchers [Bai and Reibman, 2016], and make difficult the process of extracting information [Poleg et al., 2015]. A video that can not have information extracted and create visual discomfort when watching is doomed to be forgotten. One manner to reverse this condition is to provide quick access to the video information.

Accelerating the video is one approach for reducing the impact of video length and monotonicity related problems. Although the fixed sampling is the most widely used technique, including in commercial video players, it produces jerky results when applied in First-Person Video [Poleg et al., 2015]. Due to the camera being attached to the user body, combined with non-usage of professional equipment and techniques, the egocentric videos incorporate the natural body movements of the camera wearer.

During daily activities, *e.g.*, walk, pedal, row, and sew, body movements are periodic, thus sampling at a fixed rate will increase its frequency turning the video unwatchable, and even nauseating. Consequently, fast-forward egocentric video had attracted the attention of researchers since 2014 when arose the first Hyperlapse works [Kopf et al., 2014; Karpenko, 2014].

Hyperlapse techniques aim at producing fast-forward videos from selecting a subset of aligned frames that maximize the visual smoothness. Despite being able to address the shake effects of fast-forwarding First-Person Videos, these techniques handle every frame as equally important which is a major weakness of these techniques [Kopf et al., 2014; Poleg et al., 2015; Joshi et al., 2015; Halperin et al., 2017; Ogawa et al., 2017]. In such long and monotonous video, some portions are undoubtedly more relevant than others, either for a visual content or context. One example is the recording of a graduation commencement, the welcome ceremony, taking diploma on the stage, and the family greetings are visually more memorable moments comparing with the commencement speech. Most of the Hyperlapse-based techniques have the characteristic of skipping stationary frames. Thus, if relevant frames are visually similar or static, hyperlapse methods could not include them in the fast-forwarded version, completely neglecting the relevant information.

A central challenge is to highlight the meaningful parts of the videos without losing the whole message that the video should convey. Although video summarization techniques [del Molino et al., 2016; Mahasseni et al., 2017] provide quick access to videos' information, they only return segmented clips or single images of the relevant moments. By not including the very last and the following frames of a clip, the summarization might lose the clip context [Plummer et al., 2017].

In the last couple of years, Semantic-based fast-forward methods for first-person videos have emerged as promising and effective approaches to deal with the tasks of visual smoothness and semantic highlighting of first-person videos. These works consider the semantic content of frames along with visual features to execute the frame sampling process [Okamoto and Yanai, 2014; Ramos et al., 2016; Yao et al., 2016; Higuchi et al., 2017; Lai et al., 2017; Lan et al., 2018]. Different acceleration rates were applied to semantic and non-semantic segments creating the emphasis effect. The result is the whole video being accelerated in a manner that segments containing semantic content are played slower, even in slow-motion, than the remainder of the video.

To reach both objectives, visual smoothness and semantic highlight, some of these techniques describe the video frames and their transitions by features, then formulate an optimization problem using the combination of these features. Consequently, the computation time and memory usage are impacted by the number of features used,

since the search space grows exponentially. Therefore, such Semantic Hyperlapse methods are not scalable regarding the number of features.

## 1.2 Problem Definition

The problem addressed by this dissertation is the selection of frames with constraints regarding visual smoothness, temporal continuity, and semantic load of the original video. We break this problem in the following specific problems related to existing Semantic Hyperlapse works:

1. Semantic information has been treated as a binary classification problem. Video segments are classified as semantic if they are composed of frames with semantic score higher than a threshold value.
2. Concept of semantics is defined in a *ad hoc* and restrict manner, by just considering the response of a defined and existing classifier combined with visual attributes.
3. Abrupt transition of acceleration rates assigned to following segments.
4. The non-scalability and the efficiency of the frame sampling processing, since they are based on an optimization modeling. Describe the frame using a high-dimensional feature vector leads to an uncontrolled scenario.

## 1.3 Dissertation Statements

In this dissertation, we aim at creating an efficient and visually pleasant multi-importance Semantic Hyperlapse for First-Person Videos preserving the video temporal continuity. The definition of visually pleasant multi-importance Semantic Hyperlapse is a video fast-forward technique designed to tackle the challenging production of smooth accelerated video without significant semantic loss, and capable of emphasizing video segments regarding their level of relevance.

The following questions, related to the problems presented in Section 1.2, guide our discussion through this dissertation:

1. How to approach the semantic definition to create multi-level of relevance to the frame content? Does this new approach lead to an improvement of amount of retained semantic in the accelerated video?

2. Is it possible to bind the semantic definition to the user preference?
3. Can Abrupt acceleration rate transitions present in semantically accelerated video be addressed by smoothing the speed-up rates assigned to consecutive segments?
4. How to make the frame sampling process scalable regarding the number of features used to describe the frames and their transitions?

We state that a Semantic Hyperlapse technique could be designed using sparse coding theory to perform the frame sampling in a time efficient manner. Following we list statements matching each specific questions listed in Section 1.3:

1. The multi-importance Semantic Hyperlapse is achieved by addressing the problem of semantic definition in a non-binary manner, creating a video semantic profile that frames with the highest scores will have the highest values. The multi-importance effect could be reached applying speed-up rates inversely proportional to semantic levels.
2. A machine learning based method trained from users' data tie the definition of semantics to the users' preferences.
3. Create an intermediary segment between two consecutive segments and apply a speed-up rate defined by the average of the two original segments smooths abrupt changes of acceleration rates.
4. Model the adaptive frame sampling step of the Hyperlapse as a Weighted Minimum Sparse Reconstruction problem and solve using a weighted sparse coding-based technique turns the frame sampling scalable in the number of features used to describe the frames. Assigning lower weights to frames, and consequently, over-sampling frames in regions of high camera movement create fast-forwarding video composed of smoothing transitions.

## 1.4 Contributions.

To evaluate the dissertation statements, we propose a novel methodology capable of:

- i. Analyzing, in a multi-level approach, the semantic content of a video to perform the semantic extraction, the temporal segmentation, and assigning of speed-ups rates to the video portions.

- ii. Refining the concept of semantic in a video considering what is important to the user.
- iii. Smoothing speed-up transitions between consecutive video segments.
- iv. Performing sparse sampling-based adaptive frame selection to address the problem related to abrupt camera motions while not increasing the processing time.

Additional contributions of our approach are: *i*) a high-dimensional descriptor to better describe the frames and the video transitions. *ii*) frame weighting processing regarding the camera movement to address the abrupt camera motions related problem. *iii*) creation of three datasets of First-Person Videos, the first dataset is composed of a set of short-running videos ( $\sim 5$  minutes) with a controlled amount of semantic information; the second dataset is a labeled 80-hour multimodal (3D Inertial Movement Unit, GPS, and RGB-D camera) set of lengthy first-person videos ( $\sim 1$  hour) covering a wide range of activities such as video actions, party, beach, tourism, and academic life. Each frame is labeled with respect to the activity, scene, recorder ID, interaction, and attention. *iv*) an exhaustive ablation analysis demonstrating the effect of applying each of the proposed methodology steps and algorithms. The third dataset is a set of frames from YouTube egocentric videos labeled with respect to two classes: videos with high number of likes, and videos recorded in boring and monotonous places. The two first datasets were used to perform the experimental evaluation, while the last dataset was used to train a Convolutional Neural Network.

The results of this dissertation were published on:

- *Michel M. Silva, Washington L. S. Ramos, Felipe C. Chamone, João P. K. Ferreira, Mario F. M. Campos, Erickson R. Nascimento. Making a long story short: A Multi-Importance fast-forwarding egocentric videos with the emphasis on relevant objects*, Journal of Visual Communication and Image Representation (JVCI), 2018.
- *Michel M. Silva, Washington L. S. Ramos, João P. K. Ferreira, Felipe C. Chamone, Mario F. M. Campos, Erickson R. Nascimento. A Weighted Sparse Sampling and Smoothing Frame Transition Approach for Semantic Fast-Forward First-Person Videos*, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.
- *Michel M. Silva, Washington L. S. Ramos, João P. K. Ferreira, Mario F. M. Campos, Erickson R. Nascimento. Towards Semantic Fast-Forward and*



**Stabilized Egocentric Videos**, First International Workshop on Egocentric Perception, Interaction, and Computing at European Conference on Computer Vision (ECCVW), 2016.



# Chapter 2

## Related Work

In this chapter, we first present the taxonomy used to refer techniques, then we list works related to Video Summarization, Video Fast-Forward, Hyperlapse, and Semantic Hyperlapse.

**Taxonomy.** Many Computer Vision-related techniques output a video. In this dissertation, we focused on techniques that produce a shorter video resuming the narrated story in a given input video. The following taxonomy characterizes these techniques according to the constraints imposed during the frame sampling process, which directly implies in the output video.

**Video Summarization.** The term associated with techniques that present relaxed constraints, or none, temporal continuity and visual smoothness restrictions. In general, the output video is a set of temporally disconnected video segments (skims).

**Video Fast-Forward.** This term refers to techniques that have tight temporal continuity restrictions. Output videos are composed of evenly spaced frames in time resulting of a uniform sampling approach.

**Hyperlapse.** The term used to refer techniques which perform frame sampling with balanced restrictions between visual smoothness and temporal continuity. To address both imposed constraints, these methods perform adaptive frame sampling process producing temporally continuous and visually pleasant results.

**Semantic techniques.** This term is related to techniques that perform the frame sampling process concerning an extra constraint of emphasizing segments containing semantic content. Given a defined semantic, the output is a video where

segments containing this semantic content are emphasized by a visual effect, *e.g.*, applying a lower speed-up rate, playing the segment in slow motion, applying zoom-in on the image region containing the semantics, *etc.*

In the last years, video processing to resume the story of First-Person Videos has been extensively studied, especially the video summarization problem. del Molino et al. [2016] conducted a broad study over these techniques, and one of the topics was the fundamental differences between Video Summarization and Hyperlapse techniques. Hyperlapse methods are focused on creating a visually smooth and temporally continuous fast-forward version of the input video, *i.e.*, the video is sped up entirely not removing any clips, unless there are stationary camera moments. Video summarization methods, on the other hand, are focused on creating compact visual summaries capable of presenting the most discriminative and/or the most enlightening parts of the video. These summaries do not deal with temporal continuity or visual smoothness restrictions since they are usually presented as video skims, or key-frame collection. Although the focus of this dissertation is Semantic Hyperlapse for First-Person Videos, we cover highlighted works in related areas such as Video Summarization, Video Fast-Forward, and Hyperlapse to present the big picture of the current literature.

## 2.1 Video Summarization

The ultimate goal of summarization techniques is to produce a compact version of the video keeping the essential information by either creating a static storyboard, where some selected frames resume the relevant video content [Lee et al., 2012; Song et al., 2016; Marvaniya et al., 2016], or a dynamic video skimming, where selected clips from the original stream are collated to compose the output video [Gong and Liu, 2000; Ngo et al., 2003; Zhang et al., 2016].

As far as egocentric videos are concerned, the following works have been developed recently [Lee et al., 2012; Lu and Grauman, 2013; Lin et al., 2015; Xiong et al., 2015; Yang et al., 2016]. Lee et al. [2012] exploited interaction level, gaze, and object detection frequency as egocentric properties to create a storyboard of keyframes with important people and objects. Lu and Grauman [2013] produced video skims as summaries instead of static keyframes. After splitting the video into sub-shots, they computed the mutual influence of objects and estimated the subshots importance to select the optimal chain of subshots. Lin et al. [2015] designed a context-based highlight detection algorithm based on structured Support Vector Machine (SVM) to generate video highlights. Xiong et al. [2015] proposed a summarization method that fits in the

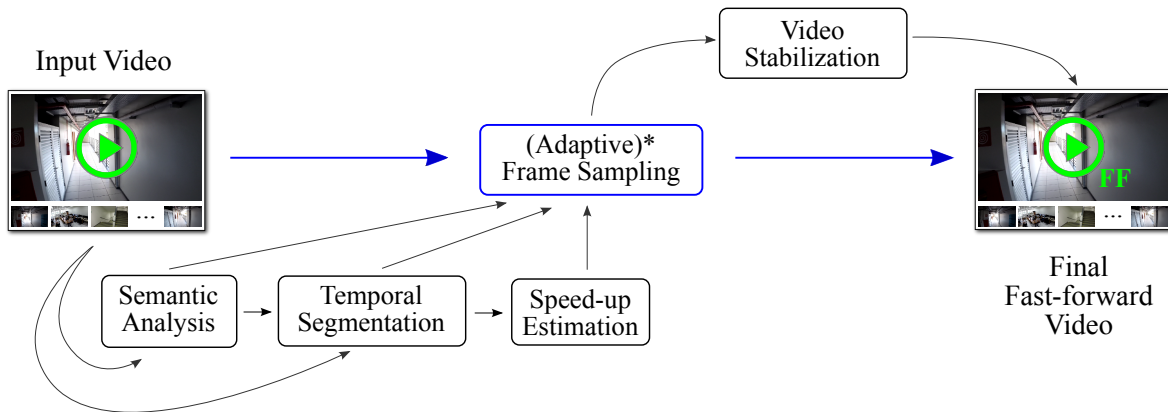
user preference performing a story-based semantic retrieval in a storyline representation of the video. Gygli et al. [2015] selected the best subset of skims by combining objectives such as interestingness, representativeness, and uniformity.

Recent approaches are based on highlight detection [Lin et al., 2015; Bettadapura et al., 2016; Yao et al., 2016] and vision-language models [Sharghi et al., 2017; Plummer et al., 2017; Panda and Roy-Chowdhury, 2017]. Bettadapura et al. [2016] proposed an approach for identifying picturesque highlights. They used composition, symmetry and color vibrancy as scoring metrics and leverage Global Positioning System (GPS) data to filter frames by the popularity of the location. Plummer et al. [2017] presented a semantically-aware video summarization. They optimize a linear combination of visual features, *i.e.*, representativeness, uniformity, interestingness, and vision-language objectives to select the best subset of video segments. Despite these techniques create summaries with relevant parts of egocentric videos, they produce, at best, temporally discontinuous video sub-shots, cutting out semantically non-relevant parts of the input video. However, these cut segments are crucial to holding the video context [Poleg et al., 2015].

Cong et al. [2012] formulated the problem of video summarization as a dictionary selection problem. They proposed a novel model to either extract keyframes or generate video skims using sparsity consistency. Zhao and Xing [2014] proposed a method based on online dictionary learning that generates key-frames collection summaries on-the-fly. They used sparse coding to eliminate repetitive events and create a representative short version of the original video. Sparse Coding has also been successfully applied to many varieties of vision tasks [Wright et al., 2009; Zhao et al., 2011; Cong et al., 2012; Zhao and Xing, 2014; Oliveira et al., 2014; Mei et al., 2014, 2015a]. The main benefit of using sparse coding for frame selection is that selecting a different number of frames does not incur an additional computational cost. This dissertation differs from sparse coding video summarization since it produce an output video handling the shakiness in the transitions via a weighted sparse frame sampling solution. Also, it is capable of dealing with the temporal gap caused by discontinuous skims.

## 2.2 Hyperlapse and Video Fast-Forward

Video Fast-Forward and Hyperlapse techniques follow a multipath pipeline as depicted in Figure 2.1. The disjunction point between these two classes of methods is the type of frame sampling performed. Fast-Forward techniques perform a Uniform Frame Sampling while Hyperlapse methods run an Adaptive Frame Selection. Uniform Sampling



**Figure 2.1.** Steps performed by Video Fast-Forward methodologies. \*Hyperlapse methods perform Adaptive Frame Sampling while Fast-Forward methods run uniform Sampling.

is a frame selection method that follows a tight temporal constraint of selecting every  $F_d$ -th frame of the video, where  $F_d$  is the required speed-up rate. We list the simplicity and time efficiency as the primary advantages of the uniform sampling. However, visual results become jerky when applied to First-Person Videos. Adaptive Selection follows the temporal continuity and also visual smoothness constraints during the frame sampling process which is guided by a maximization function, usually related to the transition stability on the produced video. Output videos of both Video Fast-forward and Hyperlapse techniques are composited of the frames selected in Frame Sampling step directly or after applying a Video Stabilization processing.

Hyperlapse strategies for First-Person Videos can be divided into two categories: 3D model approach, where methods aim at firstly creating the whole environment structure, and then finding a virtual optimal camera path through the scene to create videos composited of smooth transitions; and, 2D approach, which includes methods focused on finding an optimal set of frames based on visual smoothness criterion.

Methods applying the 3D model approach have the advantage of choice the camera poses, once it is regarding to a virtual camera. However, to calculate the virtual camera path it is necessary reconstruct the recorded surroundings first, which leads to massive computational complexity, and needs a high scene parallax. A representative member of the 3D model category is the work of Kopf et al. [2014]. Their approach consists of three stages: *i*) scene reconstruction via structure-from-motion and per-frame proxy geometries; *ii*) path planning by optimizing a 6D virtual camera path; *iii*) and image-based rendering via projection, stitching, and blending of selected input frames. Despite impressive results achieved by this technique, it requires substantial scene overlap among frames and presents high computational cost. Moreover, if the

scene parallax is small, it generates numerous artifacts. The authors used temporal segmentation due to memory restriction, segmenting the video to make the computation feasible.

The 2D based methods avoid the computational expensive 3D reconstruction process by performing adaptive sampling of the frames in the input video [Karpenko, 2014; Poleg et al., 2015; Joshi et al., 2015; Ramos et al., 2016; Halperin et al., 2017]. Different to the 3D-based works that optimize the frame selection by creating a virtual camera path, 2D methods perform the frame sampling using real camera poses only, which lead to shakier videos when compared to 3D-based results.

The work of Karpenko [2014] performs fixed frame sampling, therefore by following our taxonomy the work is a Fast-Forward technique. The authors inferred camera orientations from gyroscope data, then fed into a video filtering pipeline to estimate steady frames and stabilize the final video. Gyroscope data are gathering from the Inertial Measurement Unit (IMU) of the capture device and used to remove the unwanted hand and body movements, to posterior filtering of the body shake from the camera movements. This technique has the immediate restriction of needing data from external sensors.

Poleg et al. [2015] proposed the Adaptive Frame Selection methodology by modeling the sampling as a graph shortest path problem. The authors proposed to create a graph from the input video taking the frames as nodes and edges as the transitions from one frame to another. These transitions are modeled as a linear combination of the shakiness, speed of motion, and appearance between pairs of frames. The Adaptive Frame Sampling is performed by running a shortest path algorithm on the created graph. Their final video is composed of those frames related to nodes composing the shortest path.

Recently, Halperin et al. [2017] extended the approach present in the work of Poleg et al. [2015] with an expansion of the field of view of the output video. They created wide-view frames using the mosaicking technique on input frames of one or more egocentric videos. A stabilization process is applied to created frames by moving a cropping area compensating the frame movement. A drawback of graph-based methods is the number of parameters to be set by the user. Using default values does not produce good results, as we discuss in Section 4.

Microsoft Hyperlapse [Joshi et al., 2015] is the state-of-the-art Hyperlapse method as far as visual smoothness is concerned. The authors described the frame transitions by feature tracking techniques to recover 2D camera motion. Further, they optimally select the set of frames via dynamic-time-warping regarding the desired target speed-up and the smoothness in frame-to-frame transitions jointly. Finally, they perform a 2D

video stabilization based on homography transformation to produce the final video.

Wang et al. [2018] reinforced the importance of accelerating First-Person Videos and expanded the borders of the applications. The authors showed how to create a hyperlapse video based on multiple spatially-overlapping sources. Using multiple sources, the final video can be created synthesizing virtual routes created from paths traveled by distinct cameras. Visual pleasant result is achieved by performing graph-based adaptive frame sampling combined with video stabilization and appearance smoothing.

With the spread in the number of omnidirectional devices, Ogawa et al. [2017] and Rani et al. [2018] proposed fast-forward methods proper to  $360^\circ$  videos. Stable results were achieved by performing an adaptive frame sampling process combined with camera rotations into the omnidirectional sphere.

Although these solutions have succeeded in creating short and watchable versions of long first-person videos, they often remove segments of high relevance to the user, since the methods handle all frames as having the same semantic relevance.

## 2.3 Semantic Hyperlapse and Fast-Forward

Unlike Hyperlapse and Fast-Forward techniques, which the goal is to create a video with the required number of frames and optimize the visual smoothness in the case of Hyperlapse, Semantic Hyperlapse techniques also deal with the constraint of emphasizing video portions containing relevant content.

To the best of our knowledge, Okamoto and Yanai [2014] proposed the pioneering semantic technique by fast-forwarding a guidance video with emphasis on selected parts of the route. The authors considered street corners and pedestrian crosswalks as semantic since the path can be comprehended by the actions performed on these checkpoints. The camera motion was used to identify the street corners and an SVM classifier was trained to detect the pedestrian crosswalks. The authors temporally segmented the videos, fed each segment into the learning method and applied different speed-ups based on the classification result. Following the processing for each segment, frames were then uniformly sampled concerning to calculated speed-up. This way, the methodology dynamically controls the video playing speed. Conversely to traditional Fast-Forward method, they are not concerned about achieving a required speed-up rate in the final video.

After the arising of Hyperlapse techniques in 2014, Ramos et al. [2016] designed the first Semantic Hyperlapse approach producing visually smooth fast-forward videos from First-Person Video with emphasis on a given semantic. The proposed method



consists of creating a video profile based on the frame relevance, segmenting the video w.r.t. the created semantic profile, calculating speed-up rates for each segment in a manner that the semantic segments are played slower than the non-semantic ones, and performing graph-based adaptive frame sampling. They assigned semantic scores for each frame of the video through a liner combination over terms related to face detection and attributes such as size, confidence, and centrality.

Silva et al. [2016] extended the work of Ramos et al. [2016] by improving the temporal slicing strategy with a smarter thresholding method. Further, the authors introduced a stabilization process specially designed to fast-forward video. The main contributions of this stabilization method were to apply weighted homography transformations and image stitching using frames dropped during the sampling process. They also defined a new instability metric and performed a user study to demonstrate that the results are more consistent with the participants' opinion than the metric used so far. Finally, the authors proposed a semantically controlled and labeled dataset to evaluate fast-forward videos regarding to semantic.

As mentioned early, the main problem of graph-based frame sampling methods is the number of free parameters to be set by the user. Aiming to address this issue, Ramos [2017] proposed a bio-inspired automatic parameter setting methodology. The author extended the work of Silva et al. [2016] by proposing a two-step automatic parameter setting. The first step set the regularization terms of the equation that calculate the speed-up rates assigned to each type of segment, while the second step set the coefficients related to each term of the edge weighting during the graph modeling process. In this dissertation, we propose to create a multi-importance approach. Rather than labeling the segments as semantic or non-semantic as done so far, we aim to create a methodology to assign levels of relevance into a multiple importance scale. Thus, the segments can be emphasized according to their importance and not as a binary problem. Further, we discuss the non-scalability regarding the video length and the number of features used to represent frames and their transitions. Then, we present a sparse-based frame sampling methodology to address the scalability issues.

Yao et al. [2016] proposed to learn the relationship between paired highlights and non-highlights segments to create a summary of the video. Although the work is focused on video summarization, it has a twofold output, a video skims of sport highlight moments, and a semantic video fast-forward emphasizing sport highlight moments. The authors emphasized relevant segments by playing them in slow motion while the remaining of the video is played in a fast-forwarded manner. It is noteworthy that the authors assume the length of highlight segments smaller than the length of non-highlight segments. This assumption does not hold for videos such as the 'Biking 50p',

‘Driving 50p’ and ‘Walking 75p’ presenting in the Semantic Dataset proposed by Silva et al. [2016]. Similar to the work of Okamoto and Yanai [2014], the authors applied a temporal segmentation due to the input length of the learning method. Each created segment had assigned a speed-up rated regarding its length and predicted relevance, creating the emphasis effect. The frame selection adopted by the authors is uniform frame sampling. Therefore, by following the proposed taxonomy the work of Yao et al. [2016] is a Semantic Fast-Forward technique. When compared to the work of Yao et al. [2016], the methodology proposed in this dissertation is lighter and presents a more modular approach since we model the semantic and identify segments boundaries using classifier confidence and threshold. Furthermore, our segmentation strategy is capable of handling different configurations for the highlights lengths.

Higuchi et al. [2017] proposed a video fast-forwarding interface to assist users in tasks of finding important events on First-Person Videos. The system allows the user to select relevant egocentric cues, which are used to create an elastic timeline playing at original speed segments of the video containing the selected cues. The remaining of the video is played faster given a speed-up set by the user. The acceleration of non-relevant segments is based on uniform sampling, while the egocentric cues are ego-motion and detection of hand and people.

Lai et al. [2017] proposed a Semantic-driven Hyperlapse technique to 360° videos. The first step was to convert the video from 360° full panoramic to the normal field of view. This conversion was focused on displaying the scene regions with higher semantic content and visual saliency. The authors calculate the semantic content of a frame based on objects detected using Convolutional Neural Network (CNN) and its saliency, weighted by the user preference over a set of defined object. The semantic score is then used to control the playback rate of the accelerated video. The scope of the work of Lai et al. [2017] is slightly different from methods designed for First-Person Videos. 360° videos see the world for every point of view at each camera pose, they also do not present the user intention, *e.g.*, in an First-Person Video when an element on the scene attracts the recorder’s attention this one will change his point of view to look at it. This same behavior is not present in the 360° videos. From one 360° video, a combinatorial number of normal field of view videos can be extracted.

Lan et al. [2018] proposed a learning method which learns visually what is relevant in a video sequence and use this information to summarize an input video automatically. The highlight of this video is to process frames online, as soon as a frame is presented to the network, it is capable of determining how many frames will be jumped. It is noteworthy that those techniques do not handle the suavity constraint, generating shaky videos.

In Table 2.3, we summarize the acceleration methods presented in Sections 2.2 and 2.3 by indicating the steps performed in a set of methodology steps, the input data, and the efficiency of the sampling process. The non-semantic acceleration techniques compose the top group of the table and the semantic ones the bottom group.

Column HD shows the deficiency concerning method that performs Adaptive Frame Sampling (column AFS) of handling High Dimensional feature vectors. The only two methods capable of handling such feature vectors perform the uniform selection, not an optimization-based frame sampling. Analyzing the table, we see that the work of Lai et al. [2017] is the only Hyperlapse methodology which considers the semantic content as a multi-level problem.

In this dissertation, we aim to create a novel methodology to produce semantic hyperlapse of egocentric videos. The goal of the proposed methodology is to address issues related to the existing works such as, treat the semantic analysis as a binary problem, *ad hoc* semantic definition, and the scalability regarding the number of frames and dimension of the feature vectors used to describe the frames. We model the frame sampling step as a Minimum Sparse Reconstruction problem. To the best of our knowledge, it is the first work to address this problem using Sparse Coding based formulation. Other applications using Sparse Coding have achieved encouraging results such as Image Compression [Romberg, 2008], Image Classification [Liu et al., 2015], and Video Summarization [Mei et al., 2015b; Cong et al., 2012; Mei et al., 2014, 2015a].

**Table 2.1.** This table indicates methodology steps employed by each work focused on video acceleration. Steps are described as: (SA) Semantic Analysis, the values are: [X] not performed, [b] analyzed as binary problem, and [m] analyzed using multi scale; (TS) Temporal Segmentation; (SE) Speed-up Estimation; (AFS) Adaptive Frame Sampling; (VS) Video Stabilization. HD column indicates which method is capable of handling High-dimensional Descriptor. SP column stands for Sampling Performance, where the values are: [-] not available (method which does not perform adaptive frame sampling); [p] poor (more than 1s per frame); and [r] regular (more than 1ms per frame), [g] great (less than 1ms per frame). In the Input Video, FPV and FPVs stand for First-Person Video and its plural form, respectively.

Methods	Methodology Steps					Input Video	HD	SP
	SA	TS	SE	AFS	VS			
Karpenko [2014]	X	X	X	X	✓	FPV	X	-
Kopf et al. [2014]	X	✓	X	✓	✓	FPV	X	p
Poleg et al. [2015]	X	X	X	✓	X	FPV	X	r
Joshi et al. [2015]	X	X	X	✓	✓	FPV	X	g
Halperin et al. [2017]	X	X	X	✓	✓	FPVs	X	r
Ogawa et al. [2017]	X	X	X	✓	X	360°	X	r
Wang et al. [2018]	X	X	✓	✓	✓	FPVs	X	r
Rani et al. [2018]	X	X	X	✓	X	360°	X	r
Okamoto and Yanai [2014]	m	✓	✓	X	X	FPV	X	-
Ramos et al. [2016]	b	✓	✓	✓	X	FPV	X	r
Silva et al. [2016]	b	✓	✓	✓	✓	FPV	X	r
Yao et al. [2016]	m	✓	✓	X	X	FPV	✓	-
Higuchi et al. [2017]	b	✓	X	X	X	FPV	X	-
Ramos [2017]	b	✓	✓	✓	✓	FPV	X	r
Lai et al. [2017]	m	✓	✓	✓	✓	360°	X	p
Lan et al. [2018]	b	X	✓	X	X	Any	✓	-
<i>Ours</i>	m	✓	✓	✓	✓	FPV	✓	g

# Chapter 3

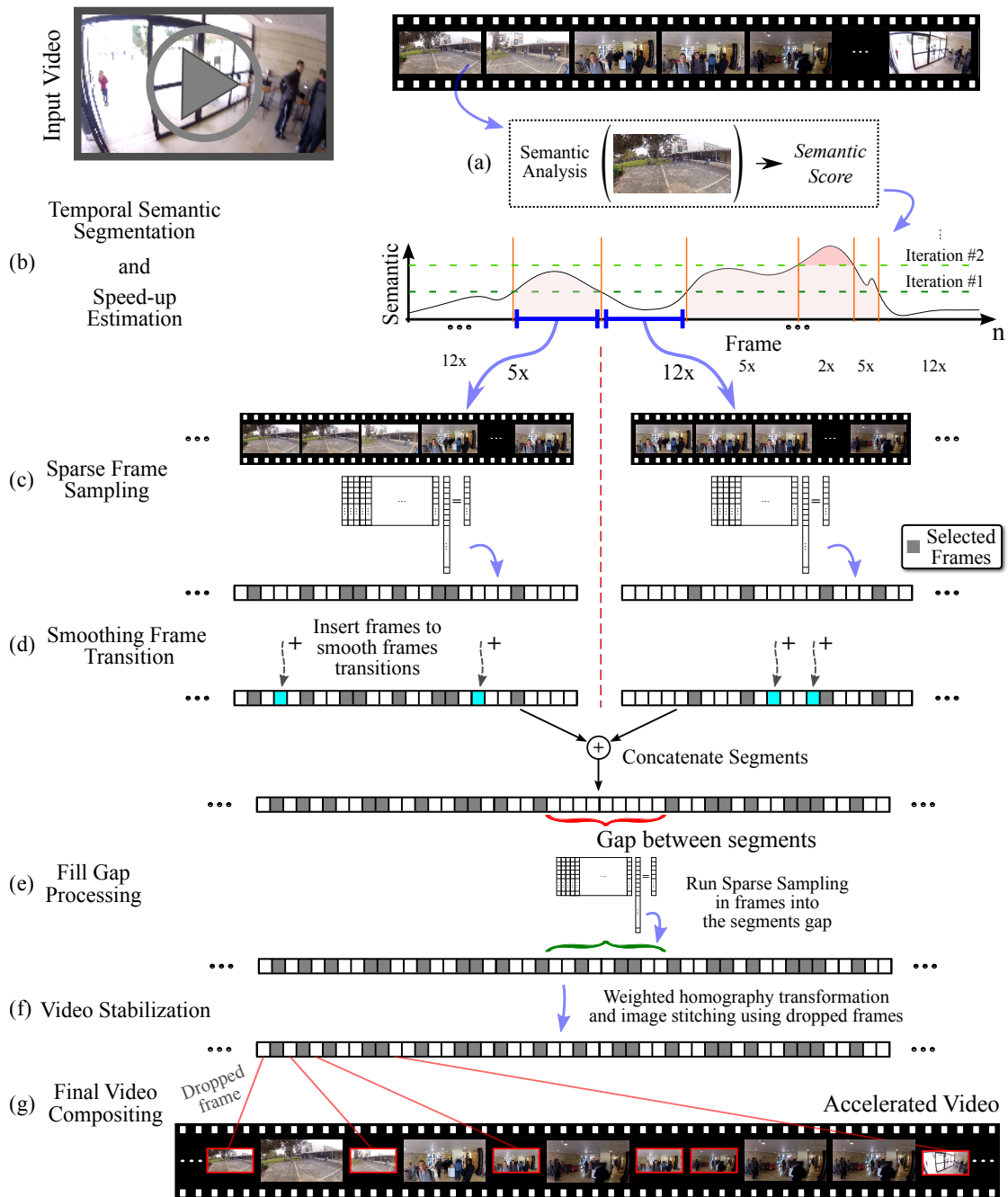
## Methodology

In this chapter, we describe the proposed methodology to create smooth and continuous fast-forward videos emphasizing the semantic contents of the original video in reduced processing time. Our method consists of three primary phases: *i*) Definition of Semantic Segments; *ii*) Adaptive Frame Sampling; and *iii*) Output Video Producing.

Figure 3.1 depicts a schematic diagram of our methodology. Each frame of the input First-Person Video is analyzed regarding its content creating a semantic video profile – Figure 3.1-a. We segment the video into semantic and non-semantic segments from the created semantic profile, and then, refine the semantic clips to create the Multi-Importance approach. The Speed-up Estimation is performed to create an emphasis effect proportional to the video segment relevance – Figure 3.1-b (these three first steps define the first phase of our methodology). The sparse sampling-based Adaptive Frame Selection is modeled as an Minimum Sparse Reconstruction problem – Figure 3.1-c. The Smoothing Frame Transition step deals with discontinuities created by the Minimum Sparse Reconstruction solution – Figure 3.1-d. Fill Gap Processing is an additional step to avoid temporal discontinuities between video segments – Figure 3.1-e. Steps (c), (d), and (e) compose the second phase of our methodology that selects the frames to compose the accelerated video. Finally, we stabilize the created semantic hyperlapse using the selected and non-selected frames – Figure 3.1-f – producing the output video – Figure 3.1-g.

### 3.1 Definition of Semantic Segments

In this section, we cover the steps Semantic Analysis and Temporal Segmentation and Speed-up Estimation depicted in Figure 3.1 (a) and (b), respectively. These methodology steps describe how the frame content analysis is performed to create the semantic



**Figure 3.1.** Overview of steps compositing the proposed sparse sampling-based semantic hyperlapse methodology. Each frame is analyzed concerning its content (a) creating a semantic profile of the input video. Following we perform a semantic temporal segmentation of the created video profile (b). For each segment, a speed-up rate is calculate regarding its relevance, and is performed a frame selection process composed of the steps Sparse Frame Sampling (c) and Smoothing Frame Transitions (d). The selected frames in each segment are then concatenated, and the step of Fill Gap Processing is applied to tackle temporal discontinuities (e). Finally, we stabilize (f) and compose the final semantic hyperlapse video (g).

video profile, how this created profile is segmented, and the speed-up estimation for each segment. We also discuss the proposed multi-importance temporal semantic segmentation and speed-up estimation steps.

### 3.1.1 Semantic Analysis

Numerous questions arise when a work mention the term semantic, *e.g.*, “*What is semantic information?*”, “*How do you define it?*”, and “*Why do you consider something as semantic?*”. These are legitimate inquiries, once semantics is an open-minded concept that can take every meaning, *e.g.*, recognition of car objects or actions, identification of people, places, scenes, sound patterns, or behavior anomalies.

In this dissertation, we first bound the definition of semantic in an *ad hoc* manner to make clear the evaluation process present in Section Experiment (4.5.1). Following, we demonstrate how to exploit the unrestricted scope of the term semantic and extrapolate its definition to fit the users’ preference.

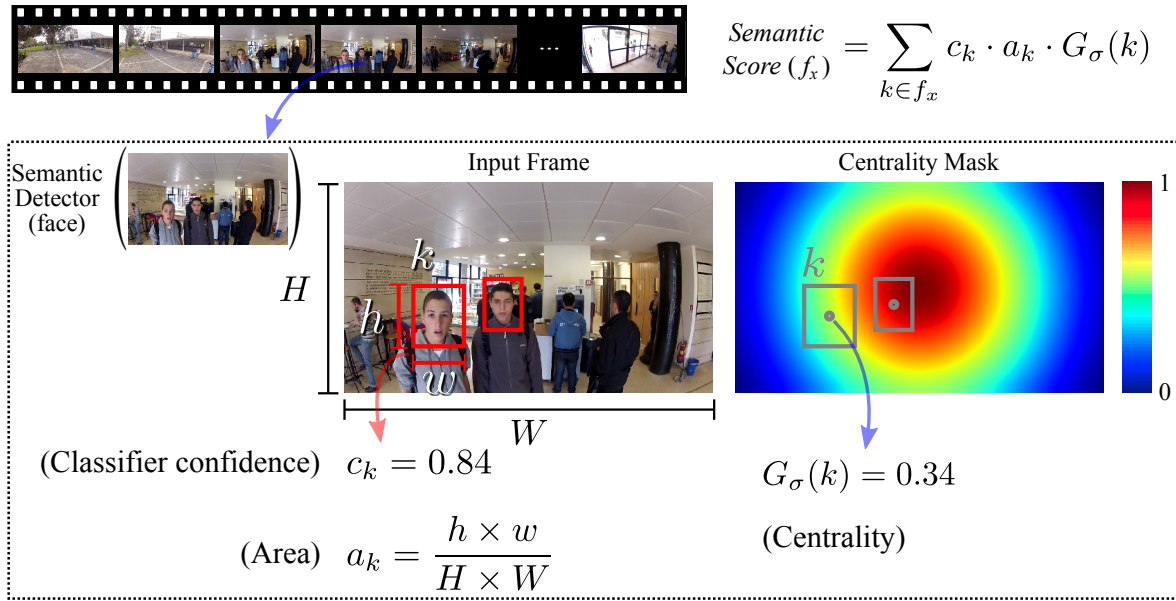
#### 3.1.1.1 Ad hoc definition

The *ad hoc* definition is used to facilitate the evaluation process and guide the understanding of visual results. Thus, we employ state-of-the-art methods achieving accuracy comparable to humans in simple tasks, *e.g.*, face and pedestrian detection.

This definition was initially proposed and formulated in the first Semantic Hyperlapse work [Ramos et al., 2016]. The semantic information was encoded by the score function  $S : \mathbb{R} \rightarrow \mathbb{R}$  composed of three components: i) the confidence of the extracted information, which is given by the face or pedestrian detector; ii) the centrality of the region returned by the semantic detector, as the input is an egocentric video, the authors considered the central area of the frame as having a higher relevance to the viewer; and iii) the size of the semantic region, representing a higher probability of interaction, once larger areas mean closer objects generally – Figure 3.2.

Let  $k$  be the  $k$ -th Region of Interest (ROI) returned by the semantic detector (red boxes in Figure 3.2) for the frame  $f_x$  of dimensions  $W \times H$ . To quantify the centrality of ROI, we use a Gaussian mask centered at the frame  $f_x$  with standard deviation  $\sigma = \min(W/2, H/2)$  – right image in Figure 3.2. Higher values are assigned to objects closer to the central point of the frame. The semantic score is given by:

$$S_x = \sum_{k \in f_x} c_k \cdot a_k \cdot G_\sigma(k), \quad (3.1)$$



**Figure 3.2.** Semantic Score calculated in an *Ad Hoc* manner. The scoring function is based on the classifier confidence about the extracted information, the size, and the centrality of the semantic region returned by the detector (The values reported are symbolic).

where  $a_k$  is the normalized area size in pixels of the  $k$ -th ROI and  $c_k$  is the normalized confidence returned by the classifier for ROI  $k$ . By using the classifier confidence, the relevance is assigned proportionally to the reliability of the semantic information. The last step is to apply a threshold value on  $c_k$  to filter false positive detections,

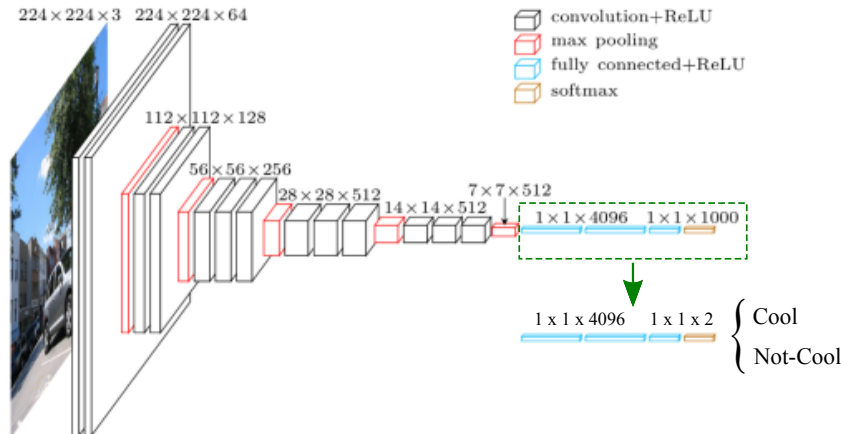
$$c_k = \begin{cases} c_k, & \text{if } c_k > \text{threshold}_{c_k} \\ 0, & \text{otherwise.} \end{cases} \quad (3.2)$$

### 3.1.1.2 Users' Preferences

The use of *ad hoc* definition of semantics is restricted and applications dependent such as suspect identification for law enforcement. Aiming to spread the application spectrum of Semantic Hyperlapse techniques to general usage, we propose a user-based Semantic Extractor focused on learning the users' preferences from personal data.

We propose a Convolutional Neural Network (CNN) model to predict the probability of users to visually enjoy an input frame based on data gathering from a video web-sharing platform. Once the goal of this network is to rate the "coolness" of an image, we analyze the input frame in its entirety, similar to the scene recognition problem. Therefore, the proposed architecture is based on the VGG16, a well-known CNN to scene recognition. The network weights were instantiated using the model trained





**Figure 3.3.** *CoolNet* Convolutional Neural Network architecture fine-tuned from VGG-16. The original VGG 1000-output layer of the Fully Connected Network was replaced by a 2-output layer related to the “Cool” and “Not-Cool” classes – highlighted by the dashed green box. (Image adapted from <https://www.cs.toronto.edu/~frossard/post/vgg16/vgg16.png>)

on MIT Places205 dataset [Zhou et al., 2014], and the final 1,000-output layer of the Fully Connected Network used to classify an image into the 1,000 VGG classes was replaced by a 2-output layer with random weights. The CNN was then fine-tuned in our domain to classify the input frame into “Cool” or “Not-Cool”, creating the *CoolNet* – Figure 3.3.

**Dataset.** To create the training dataset representing the users’ interest, we gathered videos and their respective statistics such views, likes, and dislikes, from the most accessed video web-sharing platform. Once the scope of dissertation is egocentric videos, we collected videos from the YouTube8M Dataset [Abu-El-Haija et al., 2016] using the query “GoPro”. Returned videos were ranked according to Equation 3.3 and frames of the 150-top-ranked videos were selected to composite the *Cool* class.

$$Video\_scoring = \frac{views}{(dislikes/likes)}. \quad (3.3)$$

Analyzing the selected videos, we found out most of them are related to radical sports and gorgeous landscapes. Therefore, to compose the negative class, from the list of queries in YouTube8M Dataset, we picked labels with the opposite concept of nature and sports, *i.e.*, “Home Video”, “Mobile Home”, “Office”, and “House”. Then, frames of the 150 tops ranked videos following Equation 3.3 composite the “Not Cool” class. Finally, after removing the intros, editing effects, and blurred frames, the final dataset

contains a total of 940,030 labeled images. This dataset is one of the contributions of this dissertation, and it is publicly available.

**Training.** Starting from the model trained on MIT Places205 dataset, we performed the fine-tuning in our dataset. Each frame is presented to the network with probability 1 if it is in the class “Cool”, or 0 otherwise, and the network works to predict this label correctly. The fine-tuning process follows the VGG authors guide-lines for training.

The validation process were performed using 80% of the dataset for training, and 20% for testing. After running a random search to tune the learning parameters, we set  $1 \times 10^{-6}$  for *base\_lr* and  $5 \times 10^{-4}$  for *weight\_decay*. The final network’s accuracy on test data was 98.03%.

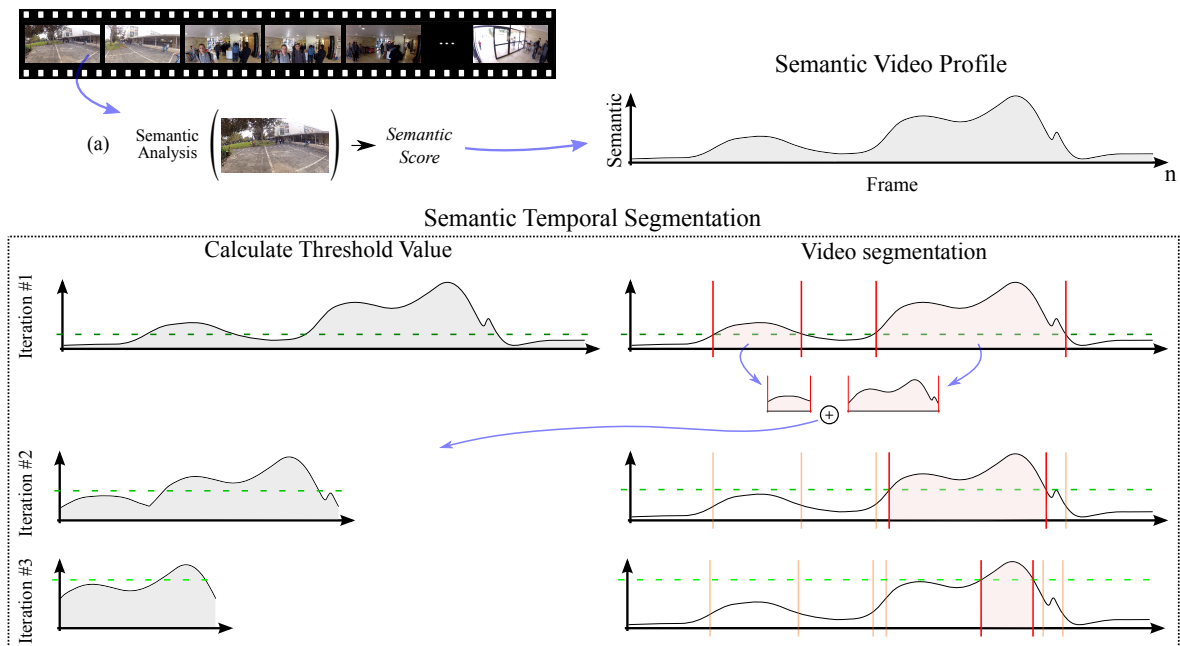
**Semantic Score.** Once the network is fine-tuned in our domain, the Semantic Score of a frame is given by the CNN confidence about the image be in the class “Cool”. The training process performed is focused on analyzing the preferences of an average Internet user since it gathers general data. However, this methodology can be extended to a single-user-focused analysis, by gathering individual data.

### 3.1.2 Temporal Segmentation

In early Semantic Hyperlapse for First-Person Videos [Ramos et al., 2016; Silva et al., 2016; Ramos, 2017], the sequence of computed Semantic Scores for each frame defined the semantic profile of the video – Figure 3.4. From the semantic profile, a histogram with semantic values was created, and a threshold value was calculated using the Otsu thresholding method [Otsu, 1979] – dashed green lines in Figure 3.4. The semantic profile of the video was then segmented using the calculated threshold in a manner that every frame above this value is labeled as semantic. Sets of consecutive frames labeled as semantic composite the semantic segments while the remaining frames composite the non-semantic segments.

We extended this binary temporal semantic segmentation approach by refining the segments taken as semantic in a multi-level relevance scale – Figure 3.1-b. This way, we treat the problem of assigning relevance as a Multi-Importance approach and not a binary classification.

We run the refinement process as depicted in Figure 3.4. First, the video is segmented into semantic and non-semantic segments using the Otsu method over the histogram of semantic content. When processing the  $i$ -th iteration, the segments assigned as non-semantic in iteration  $i - 1$  are excluded, and a new profile is created by



**Figure 3.4.** Proposed methodology to segment temporally the semantic video profile iteratively to a Multi-Importance approach.

concatenating the semantic segments; then we re-execute the Otsu threshold over the histogram of semantic information of the new profile and segment the original video using the calculated threshold. The iteration processing stops when the highest value of the semantic profile is lower than  $T$  times the threshold value returned by the Otsu method.

### 3.1.3 Speed-up estimation

The widely used technique to create emphasis effect in fast-forward videos is to assign a lower rate to relevant segments when compared to remaining of the video.

We follow this emphasizing method with an additional constraint of keeping the speed-up of the whole fast-forward video closer the desired speed-up. Differently to the work of Yao et al. [2016], we do not have restrictions regarding the length of the segment. To address the constrain regarding the overall speed-up rate of the whole video, the non-semantic segments have to be played faster than the desired speed-up  $F_d$  since the semantic segments are emphasized applying a lower rate. Given an input desired speed-up to the video, it is not a trivial task to estimate the speed-ups assigned to the semantic and non-semantic segments, since the total duration of these segments may vary a lot, and the final speed-up calculated over the final video should be closer to the desired value  $F_d$ .

Let  $F_d$  be the speed-up rate required by the user,  $L_s$  the total number of frames of all semantic segments and  $L_{ns}$  the number of frames of non-semantic segments. We compute the semantic speed-up  $F_s$  and the non-semantic speed-up  $F_{ns}$  by minimizing the energy function:

$$D(F_{ns}, F_s) = \left| \frac{L_s + L_{ns}}{F_d} - \left( \frac{L_s}{F_s} + \frac{L_{ns}}{F_{ns}} \right) \right|. \quad (3.4)$$

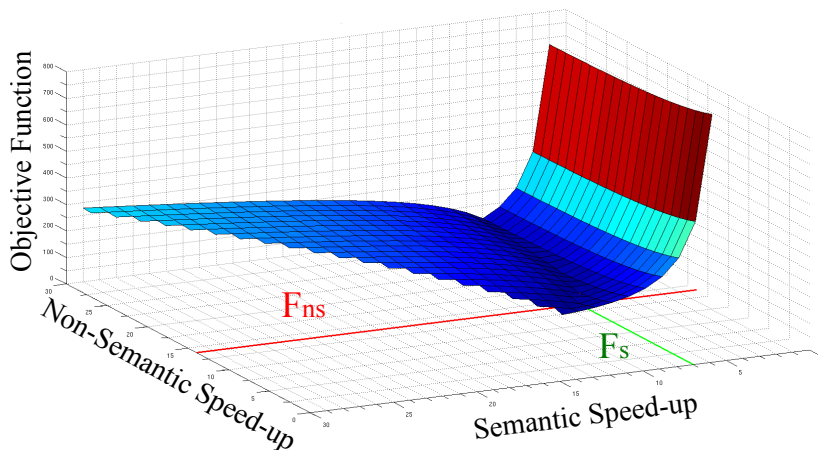
Equation 3.4 has multiple solutions since for every  $F_s$  value there is a  $F_{ns}$  leading the result to 0. Therefore, we solve this equation by an optimization problem in which we minimize the energy function  $D(F_{ns}, F_s)$  in a manner that value of the speed-up assigned to semantic segments  $F_s$  is as low as possible and the difference between speed-up rates assigned to non-semantic  $F_{ns}$  and semantic segments  $F_s$  is as close as possible. In this dissertation, we deal only with integer values to speed-up, and as the aim is to emphasize relevant segments, the following three restrictions are imposed: (i)  $F_s \leq F_d$  because we want emphasis in semantic parts; (ii)  $F_{ns} \geq F_d$ , since we want to achieve the desired speed-up in the fast-forward video and; (iii)  $F_s \geq p_s \cdot F_d$ , where  $p_s = L_s / (L_s + L_{ns})$ , to avoid an excessive number of frames. Thus, the optimization model as follows:

$$\begin{aligned} F_s^*, F_{ns}^* &= \arg \min_{F_s, F_{ns}} D(F_{ns}, F_s) + \lambda_{dif} |F_{ns} - F_s| + \lambda_{f_s} |F_s| \\ \text{s.t. } F_s &\leq F_d \\ F_{ns} &\geq F_d \\ F_s &\geq p_s F_d, \end{aligned} \quad (3.5)$$

where  $\lambda_{dif}$  and  $\lambda_{f_s}$  are the regularization parameters used to control the importance of keeping the speed-up rates closer or setting lower values to  $F_s$ , respectively. An example of the search space is depicted in Figure 3.5.

Following the proposed idea of a multi-importance approach, we handle the different levels of semantic content defined in Section 3.1.2 by estimating speed-up for semantic segments according to their relevance.

Our multi-importance strategy is implemented by following the iterative process defined in Section 3.1.2. During the first iteration, the video is segmented based on the threshold value calculated using the Otsu method, and Equation 3.5 is calculated. From the second iteration and beyond, non-semantic segments defined in the last iteration are ignored, and a new profile is created by concatenating all semantic segments defined in the last iteration. At this point, we set the desired speed-up value  $F_d$  in Equation 3.5



**Figure 3.5.** Example of the search space related to the speed-up optimization function described in Equation 3.5.

to the semantic speed-up  $F_s$  calculated in the iteration  $i - 1$ , and new values for  $F_s$  and  $F_{ns}$  are calculated regarding the length of the new semantic and non-semantic segments refined in iteration  $i$ . This process follows up to the stopping point defined in Section 3.1.2.

The idea of this processing is to accelerate the less important segments with the first non-semantic speed-up, higher than the value required by the user, while the speed-up rates assigned to the semantic segments will be refined in a manner that as higher is the relevance of a segment, lower is the value assigned.

**Parameter Settings.** Equation 3.5 has two parameters ( $\lambda_{dif}$  and  $\lambda_{fs}$ ) highly related to the input video, demanding user knowledge and effort to configure them. It is highly probable that the user will stop before finding the best set of parameters, which could lead to poor results. We set these parameters automatically using the Particle Swarm Optimization (PSO) algorithm, heading an optimal result and also excluding the user of the process’s pipeline, as presented in the work of Ramos [2017].

PSO algorithm is an iterative method that groups particles by arranging them randomly in the search space [Kennedy and Eberhart, 1995]. At every iteration, the particles positions (parameters values) are updated to follow the local and global best particles. The solution is given by a fitness equation defined according to the problem. To solve the optimization Equation 3.5, we define the following fitness equation:

$$\text{fitness}_{\lambda_{dif}, \lambda_{fs}} = c \cdot \left| \widehat{F}_s - \frac{F_d + p_s \cdot F_d}{2} \right| + |\widehat{F}_d - F_d| + p_{ns} \cdot |\widehat{F}_s - \widehat{F}_{ns}|, \quad (3.6)$$

which estimates  $\lambda_{dif}$  and  $\lambda_{fs}$  of Equation 3.5. The  $\widehat{F}_s$  and  $\widehat{F}_{ns}$  are the best values of

$F_s$  and  $F_{ns}$  in the finite and discrete search space when replacing  $\lambda_{dif}$  and  $\lambda_{f_s}$  with the particle position. The value  $p_s = L_s/(L_s + L_{ns})$  is the semantic percentage of the video,  $p_{ns} = L_{ns}/(L_s + L_{ns})$  is non-semantic percentage,  $c = 2$  is a constant value to control the importance of selecting a lower semantic speed-up, and  $\widehat{F}_d$  is the calculated video speed-up when applying the proposed semantic and non-semantic speed-ups given the length of each segment, as described in the following Equation:

$$\widehat{F}_d = \frac{L_s + L_{ns}}{L_s/\widehat{F}_s + L_{ns}/\widehat{F}_{ns}}. \quad (3.7)$$

## 3.2 Adaptive Frame Sampling

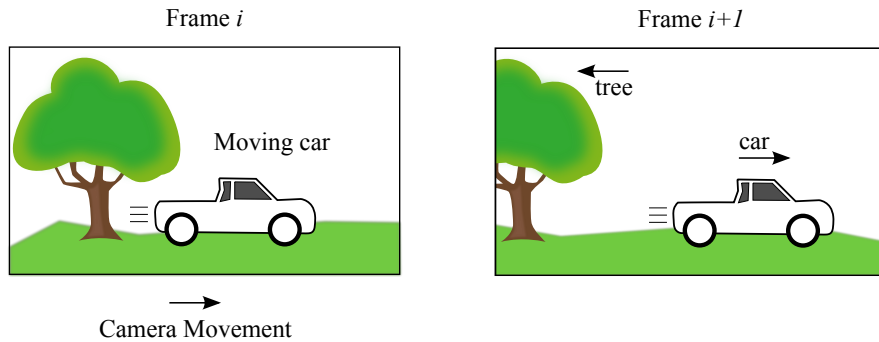
Adaptive Frame Sampling is the second phase of our three-phase methodology. The goal of this step is to select a set of frames to produce a visually smooth and temporally continuous accelerated video.

We proposed a sparse-based approach to perform the frame selection step in our Semantic Hyperlapse methodology. The sparse-based approach models the frame sampling as a Minimum Sparse Reconstruction problem, and applies two extra steps, the Smoothing Frame Transitions and Fill Gap process.

### 3.2.1 Sparse-based selection

In general, hyperlapse techniques solve the adaptive frame selection problem searching the optimal configuration (*e.g.*, shortest path in a graph or dynamic programming) in a space of representation where different types of features are combined to represent frames or transitions between frames. Once the search space dimensionality is corresponding to the length of the features vectors, most of these techniques use a single value to represent complex characteristics of frames, such as frame movement or appearance, and describe the whole frame using four features maximum [Joshi et al., 2015; Poleg et al., 2015; Ramos et al., 2016; Silva et al., 2016; Halperin et al., 2017; Ramos, 2017]. Otherwise, the use of a higher number or high-dimensional features leads to exponential growth in the search space of the optimization problem. Therefore, these methods are not scalable in the number of features used to describe the frame and transitions.

We look at the use of a single value to describe complex characteristics as a limitation. One example is the frame movement that is usually described as the mean value of the optical flow magnitudes related to image pixels. Figure 3.6 depicts a frame transition captured during a camera turning in a scene composed of fixed and moving



**Figure 3.6.** Synthetic consecutive frames depicting a car is moving from left to right and a camera turning to the right. A complex scene dynamics is depicted on the second image, the car shifted to the right, once the car moved faster than the camera, and the tree to the right, due to the camera movement.

objects. This figure illustrates the inconsistency of using a single value to describe the movement of both fixed object moving left, opposite to the camera turn direction, and a moving object going right. The same limitation is observed when describing the frame appearance, usually calculated using the difference between color histograms related to consecutive frames. However, the difference will result in a small value since the same elements compose both images. Recent works present state-of-the-art results in different applications using high-dimensional feature vectors [Otani et al., 2017; Lal et al., 2019; Fu et al., 2019].

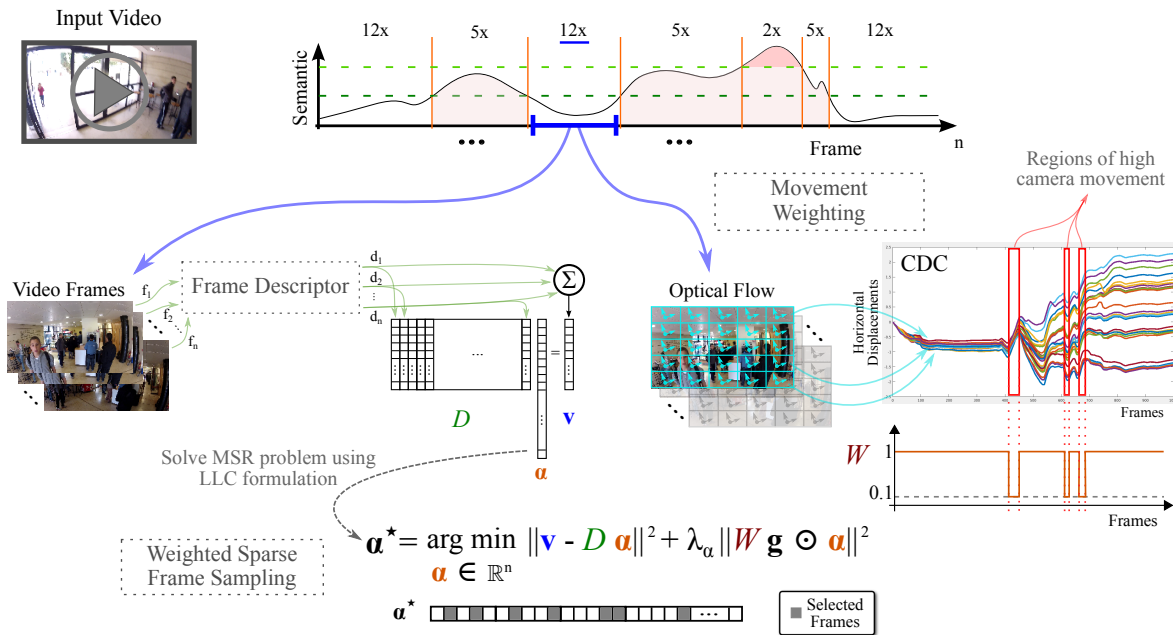
In this dissertation, we introduce the sparse-based Adaptive Frame Sampling method formulated as a Minimum Sparse Reconstruction problem – Figure 3.7. The goal is to create a formulation to the adaptive frame sampling step in which we can describe the frames and their transitions in more details with no impact in the processing time.

Let  $D = [\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3, \dots, \mathbf{d}_n] \in \mathbb{R}^{c \times n}$  be a segment of the original video with  $n$  frames represented in our feature space. Each entry  $\mathbf{d}_i \in \mathbb{R}^c$  stands for the feature vector of the  $i$ -th frame. Let the video story  $\mathbf{v} \in \mathbb{R}^c$  be defined as the sum of the frame features of the whole segment, *i.e.*,

$$\mathbf{v} = \sum_{i=1}^n \mathbf{d}_i. \quad (3.8)$$

The goal is to find an optimal subset  $\mathcal{S} = [\mathbf{d}_{s_1}, \mathbf{d}_{s_2}, \mathbf{d}_{s_3}, \dots, \mathbf{d}_{s_m}] \in \mathbb{R}^{c \times m}$ , where  $m \ll n$  and  $\{s_1, s_2, s_3, \dots, s_m\}$  belongs to the set of frames in the segment.

Let the vector  $\boldsymbol{\alpha} \in \mathbb{R}^n$  be an activation vector indicating whether  $\mathbf{d}$  is in the set  $\mathcal{S}$  or not. The problem of finding the values for  $\boldsymbol{\alpha}$  that lead to a small reconstruction



**Figure 3.7.** Sparse-based Adaptive Frame Sampling methodology. For each segment created in the temporal semantic profile segmentation, frame-wise processes are performed to compute both weights, based on the camera movement, and frame descriptors. The sampling step is modeled as a Minimum Sparse Reconstruction (MSR) problem, in a manner that frames related to activated positions of the sparse vector composite the final video.

error of  $\mathbf{v}$ , can be formulated as a Locality-constrained Linear Coding (LLC) [Wang et al., 2010] problem as follows:

$$\arg \min_{\alpha \in \mathbb{R}^n} \|\mathbf{v} - D \alpha\|^2 + \lambda_\alpha \|\mathbf{g} \odot \alpha\|^2, \quad (3.9)$$

where  $\mathbf{g}$  is the Euclidean distance of each dictionary entry  $\mathbf{d}_i$  to the segment representation  $\mathbf{v}$ , and  $\odot$  is an element-wise multiplication operator. The  $\lambda_\alpha$  is the regularization term of the locality of the vector  $\alpha$ .

The benefit of using LLC formulation instead of the traditional Sparse Coding (SC) models, such as Orthogonal Matching Pursuit and Lasso, is twofold: *i*) the LLC provides local smooth sparsity; and *ii*) it can be solved by an analytic solution, which results in a smoother final fast-forward video in a lower computational cost.

**Weighted Sampling.** Abrupt camera motions are challenging issues for fast-forwarding video techniques. They might lead to the creation of shaky and nauseating videos. To tackle this issue, we used a weighted Locality-constrained Linear Coding



formulation, where each dictionary entry has a weight assigned to it:

$$\boldsymbol{\alpha}^* = \arg \min_{\boldsymbol{\alpha} \in \mathbb{R}^n} \|\mathbf{v} - D \boldsymbol{\alpha}\|^2 + \lambda_{\alpha} \|W \mathbf{g} \odot \boldsymbol{\alpha}\|^2, \quad (3.10)$$

where  $W$  is a diagonal matrix built from the weight vector  $\mathbf{w} \in \mathbb{R}^n$ , *i.e.*,  $W \triangleq \text{diag}(\mathbf{w})$ .

Let  $C \in \mathbb{R}^{g \times n}$  be the set of  $g$  Cumulative Displacement Curves [Poleg et al., 2014] of a video composed of  $n$  frames. Each curve represents the cumulative sum of the horizontal displacements of the Optical Flow magnitudes computed in a cell over the  $5 \times 5$  grid windows of the video frame, as depicted in Figure 3.7. Let  $C' \in \mathbb{R}^{g \times n}$  be the set of derivative of each curve  $C$  w.r.t. time. We assume frame  $i$  to be within an interval of abrupt camera motion if all curves  $C'$  present the same sign (positive/negative) at the point  $i$ , which represents a head-turning movement [Poleg et al., 2014] – red boxes in CDC curve in Figure 3.7. We assign a lower weight for these motion intervals to enforce them to be composed of a larger number of frames. We empirically set the weights to  $\mathbf{w}_i = 0.1$  and  $\mathbf{w}_i = 1.0$  for the frame features inside and outside the interval, respectively.

This weighting formulation provides a flexible solution, in which we create different weights for frames based on the camera movements. When frames into regions of high camera movement are sampled, they do not increase the sparsity/locality term significantly, once their distance to the dictionary basis is multiplied by a lower weight. However, regarding the reconstruction term, the first part of Equation 3.10, they contribute similarly to any other frame since the weighting factor is not applied in this part of the equation. Contextualizing in frame sampling, the solution of the weighted formulation leads to an oversampling in regions of high camera movement minimizing the reconstruction term with no impact in the locality term.

**Speed-up Control.** All frames related to the activated positions of the vector  $\boldsymbol{\alpha}^*$  will be selected to compose the final video. Since  $\lambda_{\alpha}$  controls the sparsity, it also manages the speed-up rate of the output video. The zero-value  $\lambda_{\alpha}$  enables the activation of all frames leading to a complete reconstruction. To achieve the desired speed-up, we perform an iterative search starting from zero, as depicted in Algorithm 1. The function *NumberOfFrames*( $\lambda$ ) (Algorithm 1 line 4) solves Equation 3.10 using  $\lambda$  as the value of  $\lambda_{\alpha}$  and returns the number of activations in  $\boldsymbol{\alpha}^*$ .

**Frame Description.** Once our solution is able to handle high-dimensional frame descriptions, we propose to describe the  $i$ -th frame through the feature vector  $\mathbf{d}_i \in \mathbb{R}^{446}$  by concatenating the following terms. The  $\mathbf{hof}_m \in \mathbb{R}^{50}$  and  $\mathbf{hof}_o \in \mathbb{R}^{72}$  are histograms

---

**Algorithm 1** Sparse-based frame sampling Lambda value adjustment

---

**Require:** Desired length of the final video  $VideoLength$ .**Ensures:** The  $\lambda_\alpha$  value to reach the desired number of frames.

```

1: function LAMBDA_ADJUSTMENT( $VideoLength$ )
2:    $\lambda_\alpha \leftarrow 0$  ,  $step \leftarrow 0.1$  ,  $nFrames \leftarrow 0$ 
3:   while  $nFrames \neq VideoLength$  do
4:      $nFrames \leftarrow NumberOfFrames(\lambda_\alpha + step)$ 
5:     if  $nFrames \geq VideoLength$  then
6:        $\lambda_\alpha \leftarrow \lambda_\alpha + step$ 
7:     else
8:        $step \leftarrow step/10$ 
9:     end if
10:  end while
11: end function

```

---

of optical flow magnitudes and orientations of the  $i$ -th frame, respectively. The appearance descriptor,  $\mathbf{a} \in \mathbb{R}^{144}$ , is composed of the mean, standard deviation, and skewness values of HSV color channels of the windows in a  $4 \times 4$  grid of the frame  $i$ . To define the content descriptor,  $\mathbf{c} \in \mathbb{R}^{80}$ , we first use the CNN YOLO [Redmon and Farhadi, 2016] to detect the objects in the frame  $i$ ; then, we create a histogram with these objects over the 80 classes of the YOLO architecture. Finally, the sequence descriptor,  $\mathbf{s} \in \mathbb{R}^{100}$ , is an one hot vector, with the  $\text{mod}(i, 100)$ -th feature activated indicating in which portion of the video the frame is.

### 3.2.1.1 Smoothing Frame Transitions

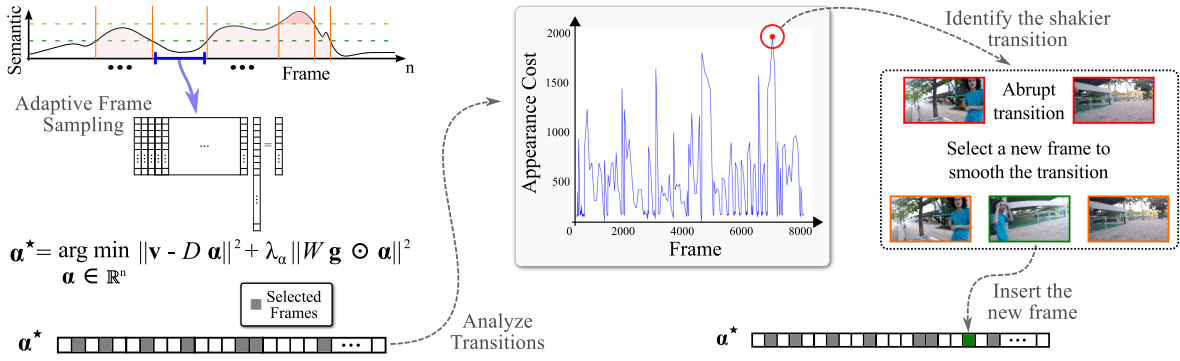
A solution  $\alpha^*$  does not ensure a final continuous fast-forward video. Occasionally, the solution might provide a low reconstruction error of small and highly detailed segments of the video. Thus, by creating a better reconstruction with a limited number of frames,  $\alpha^*$  might ignore stationary moments or visually similar views and output videos comparable to results of summarization methods.

We address this issue by dividing the frame sampling into two steps. First, we run the weighted sparse sampling to reconstruct the video using a required speed-up multiplied by a factor  $SpF$ . The resulting video contains  $1/SpF$  of the desired number of frames. Then, we iteratively insert frames into the shakier transitions (Figure 3.8) until the video achieves the exact number of frames.

Let  $I(F_x, F_y)$  be the instability function defined by

$$I(f_x, f_y) = AC(f_x, f_y) \times (p_y - p_x - F_d). \quad (3.11)$$

The function  $AC(f_x, f_y)$  calculates the Earth Mover's Distance [Pele and Werman,



**Figure 3.8.** For each segment of the video, we evaluate the Appearance Cost using Equation 3.11 over all selected frames. Then, we smooth shakier transitions iteratively by inserting frames from the original video.

2009] between the color histograms of the frames  $f_x$  and  $f_y$ . The second term of the instability function is related to the speed-up deviation. Let  $p_i$  be the index of the  $i$ -th frame in the original video. We calculate the speed-up deviation by the difference of the required speed-up  $F_d$  and the distance between frames  $f_x$  and  $f_y$ , *i.e.*,  $p_y - p_x$ . We identify a shakier transition using the Equation 3.12:

$$i^* = \arg \max_{i \in \mathbb{R}^m} I(f_{s_i}, f_{s_{i+1}}). \quad (3.12)$$

The frames  $f_{s_{i^*}}$  to  $f_{s_{i^*+1}}$ , *i.e.*, solution of Equation 3.12, compose the transition with the most visually dissimilar frames and a distance between them higher than the required speed-up – red circle in Figure 3.8.

After identifying the shakier transition, from the subset with frames of the original video ranging from  $f_{s_{i^*}}$  to  $f_{s_{i^*+1}}$ , we choose the frame  $f_{j^*}$  that minimizes the instability of the frame transition as follows:

$$j^* = \arg \min_{j \in \mathbb{R}^n} I(f_{s_{i^*}}, f_j)^2 + I(f_j, f_{s_{i^*+1}})^2. \quad (3.13)$$

For each iteration, the frame  $f_{j^*}$  selected to smooth the shakier transition is added in the set of selected frames  $\mathcal{S}$ . Equations 3.12 and 3.13 can be solved by exhaustive search, since the interval is small.

### 3.2.1.2 Fill Gap between segments

Using LLC formulation over video segment may lead to temporal discontinuities between some of the segments. These discontinuities occur due to the frame selection being performed to each segment disregarding the others. Once the last selected frame

of one segment is far from the first selected frame of the following video segment, it turns in a visual gap when the final video is composited. Section 3.2.1.1 presents a valid solution by inserting frames and tackling the visual discontinuities created inside the segments. However, it has no effect on frames transitions between the segments.

Abrupt speed-up difference between video segments is an additional issue present in most semantic fast-forward methods in the literature. This abrupt difference is caused by the selection of speed-up rates assigned to video segments. Generally, it occurs when one segment containing a significant amount of semantic information is followed by, or follows, a non-semantic segment. This case would cause abrupt difference on the speed-up rates assigned to each segment, *e.g.*, in experiment “Driving\_50p” a semantic segment with speed-up  $1\times$  is followed by a  $14\times$  non-semantic segment. In this section, we present a solution that addresses both the visual gap and the abrupt speed-up difference issues.

To address the visual gap issue, we first calculate the instability index (Equation 3.11) between the last frame of a segment  $A$  and the first frame of its consecutive segment  $B$ . If the instability index is higher than the average instability over all transitions of segment  $A$ , then we create a new segment delimited by the last frame of segment  $A$  and the first frame of the segment  $B$  (Figure 3.1-e). This newly created segment is then used to smooth the speed-up transition and fill the visual gap. To tackle the abrupt speed-up difference issue, we define the speed-up rate for the new segment as the average value between the speed-ups of  $A$  and  $B$ . Finally, we fill the visual gap by running the Weighted Sparse Frame Sampling and Smoothing Frame Transitions, defined in Sections 3.2.1 and 3.2.1.1 respectively, using the smoother calculated speed-up.

### 3.3 Output Video Producing

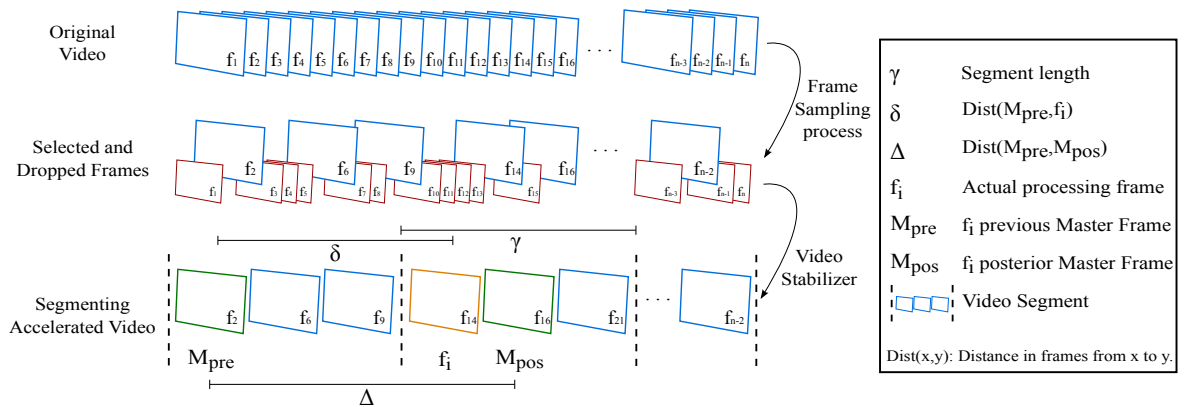
In the last phase of our methodology, we have the set  $\mathcal{S}$  of candidate frames to composite the final video. These frames are candidates because they yet can be dropped during the video stabilization, the first step of the Producing Output Video processing. Video stabilization step refines the set  $\mathcal{S}$  by stabilizing the selected frames, and if necessary, replacing frames which yield to homograph transformations that distort the frames or make impossible to reconstruct them. After this refinement, all frames in set  $\mathcal{S}$  will compose the final video.

### 3.3.1 Video Stabilization

As studied and discussed in the work of Kopf et al. [2014], traditional video stabilization algorithms do not perform at their best when applied to egocentric videos. The poor performance can be assigned to the difficulty of tracking motion between successive frames due to abrupt changes in camera pose, which is intensified in the accelerated version. In this dissertation, we propose to stabilize the final video similar to the stabilization method present in our previous work Silva et al. [2016], which is particularly designed for accelerated videos. Following in this Section, we describe the stabilizer in details.

As the stabilizer is designed to accelerated videos, its input is the output set  $\mathcal{S}$  of selected frames from the sampling phase. Algorithm 2 details the stabilization process. Initially, we split the video into segments of fixed size  $\gamma$ , and select one master frame for each segment (Figure 3.9). A master frame  $M_k$  of the  $k$ -th segment is the frame  $f$  that maximizes the number of *inliers* obtained with RANSAC when computing the homography transformation from images into the  $k$ -th segment to  $f$ .

The next step is to smooth out the frame transitions by using the computed master frames. The key idea is to create smooth frame transitions by setting target image planes on masters  $M_{pre}$  and  $M_{pos}$ , then modifying the image planes of frames between these masters in a manner to smooth the transition from  $M_{pre}$  to  $M_{pos}$  image planes. As shown in line 5 of Algorithm 2, for each frame  $f_i$  of the accelerated video, we compute a frame  $\hat{f}_i$  of the stabilized video using  $\hat{f}_i = H_{f_i, M_{pre}}^{1-w} \cdot H_{f_i, M_{pos}}^w \cdot f_i$ . The matrices  $H_{f_i, M_{pre}}$  and  $H_{f_i, M_{pos}}$  transform the frame  $f_i$  to the previous master frame



**Figure 3.9.** Stabilization methodology. The top row depicts the original video in a frame sequence. The middle row shows the selected and dropped frames in the sampling process (larger blue and smaller red frames, respectively). The last row presents an example of the accelerated video segmentation, master frames, and terms  $\gamma$ ,  $\Delta$  and  $\delta$ .

---

**Algorithm 2** Egocentric Accelerated Video Stabilizer

---

**Require:** Set of frames  $\mathcal{S}$  in the accelerated Video; Set  $\mathcal{D}$  of dropped frames during the sampling process; The *crop\_area* and *drop\_area*.**Ensures:** The set of stabilized frames  $\mathcal{V}$ .

```

1: function VIDEOSTABILIZER( $\mathcal{S}, \mathcal{D}$ )
2:    $\mathcal{V} \leftarrow \{\}$ 
3:   for all  $f_i \in \mathcal{S}$  do
4:      $w \leftarrow (\delta \cdot (2 \cdot \gamma) / \Delta)$ 
5:      $\hat{f}_i \leftarrow H_{f_i, M_{pre}}^{1-w} \cdot H_{f_i, M_{pos}}^w \cdot f_i$ 
6:     while  $\hat{f}_i \cap \text{crop\_area} < \text{crop\_area}$  do
7:       if  $f_i \cap \text{drop\_area} = \text{drop\_area}$  and ExistUnusedFrames( $\mathcal{D}$ ) then
8:          $\hat{f}_i \leftarrow \text{Stiching}(\hat{f}_i, \text{GetUnusedFrame}(\mathcal{D}))$ 
9:       else
10:         $f_d \leftarrow \text{SelectNewFrame}(\mathcal{D}, f_i)$ 
11:         $w \leftarrow (\delta \cdot (2 \cdot \alpha) / \Delta)$  ▷ Recalculate distances using  $f_d$  as  $f_i$ .
12:         $\hat{f}_i \leftarrow H_{f_i, M_{pre}}^{1-w} \cdot H_{f_i, M_{pos}}^w \cdot f_d$ 
13:      end if
14:    end while
15:     $\mathcal{V} \leftarrow \mathcal{V} + \{\hat{f}_i \cap \text{crop\_area}\}$ 
16:  end for
17: end function

```

---

$M_{pre}$  and to the posterior master frame  $M_{pos}$ , respectively. The  $\delta$  value is number of frames from  $f_i$  to  $M_{pre}$  and  $\Delta$  is the number of frames between  $M_{pre}$  and  $M_{pos}$ . Like in the work of Hsu et al. [2012], we weight both homography transformations according to the distance to the master frames.

Black areas may be created after applying the homography transformations due to abrupt camera motions and the large elapsed time between consecutive frames in the accelerated videos. Thus, we define two areas centered at the frame to decide when a frame should be reconstructed: the drop area delimited by the red line in Figure 3.10, and equals to  $dp\%$  size of the frame; and the crop area delimited by the green line in Figure 3.10, and equals to  $cp\%$  size of the frame ( $cp > dp$ ).

The drop area  $dp\%$  is the center of the image, where the viewer focuses on the majority of the time, then it is not allowed to have any black or reconstructed areas in this region. The region between the drop and crop areas is related to the peripheral vision, thus we consider permitted to have artifacts but not black areas. In the final video, we remove the regions outside of crop area, therefore, having black areas within them does not cause any issue. The defined reconstruction process ensure every  $\hat{f}_i$  frame covers the crop area.

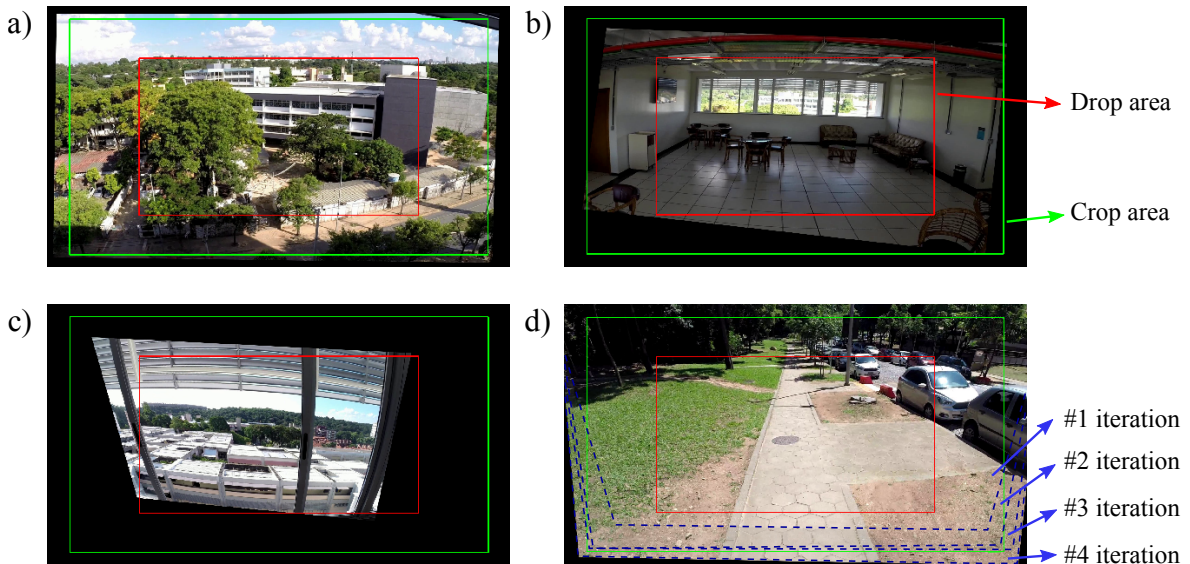
The location of the black areas created by the application of the homography

transformation in the image leads to three possible cases with related actions: *i*) black areas outside of the crop area – Figure 3.10-a – no further action is required; *ii*) black areas outside of the drop area and inside the crop area – Figure 3.10-b – image reconstruction needed; and *iii*) black area inside the drop area – Figure 3.10-c – discard the image and select a new one in the set of dropped frames.

There are two conditions for reconstructing a frame: *i*)  $\hat{f}_i$  does not create black regions in the central area; and *ii*) there are unused frames in the dropped set  $\mathcal{D}$  for stitching. If both these conditions hold, we perform the stitching using  $\hat{f}_i$  and a new frame from  $\mathcal{D}$ . If one of the conditions is false, we discard  $\hat{f}_i$  and select a new frame  $f_d$  from  $\mathcal{D}$  and recalculate the distances and homography matrices. Once the crop area is covered, the intersection between this area and the frame  $\hat{f}_i$  compose the *i*-th frame in the stabilized video.

If the  $\hat{f}_i$  does not cover the crop area in the final of the processing, we select a new frame  $f_d^*$  belonging to the interval  $[f_{i-1}, f_{i+1}]$  in the set of dropped frames  $\mathcal{D}$  (Algorithm 2 line 10) and that maximizes the equation:

$$f_d^* = \arg \max_{f_d} ( G_\sigma(p) \cdot ( R(f_d, f_{i-1}) + R(f_d, f_{i+1}) ) \cdot (\eta + S(f_d)) ), \quad (3.14)$$



**Figure 3.10.** Possible cases after the application the homography transformations in relation to crop and drop areas. a) Image covers the crop area – no further action required. b) Image covers the drop area but does not cover the crop area – reconstruction is needed. c) Image does not cover crop area neither the drop area – discard the image and select a new one in the unset frames set. d) Result of the reconstruction process – each dashed line illustrate one algorithm iteration result.

where  $G_\sigma(x)$  is a Gaussian function with mean one and standard deviation  $\sigma$  in the position  $x$ ,  $p$  is the percentage of the crop area covered by  $d_j$ ,  $R(\cdot)$  is the number of inliers obtained with RANSAC and  $S(\cdot)$  is the semantic score given by Equation 3.1.  $\eta$  is used to avoid multiplying by zero.

Frames in set  $\mathcal{V}$  are the final to composite the output video.

### 3.3.2 Video Compositing

All selected frames of each segment created by the semantic temporal segmentation are inserted in the set of selected frames  $\mathcal{S}$  to compose the final video. This set is refined using the egocentric video stabilization described in the Section 3.3.1 in which frames are transformed or even replaced to create a smooth final video creating the new set  $\mathcal{V}$ . The output is produced by the frames in set  $\mathcal{V}$  sorted by the index of these frames in the original video. The final accelerated video speed-up ( $F_f$ ) is given by the ratio between the number of frames in the original video ( $n$ ) and the number of frames in set  $\mathcal{V}$ :

$$F_f = \frac{n}{|\mathcal{V}|}. \quad (3.15)$$



# Chapter 4

## Experiments

We present the experimental evaluation of the proposed methodology, which includes describing the datasets used, the parameters configuration, weights setting, methods, and metrics chosen for quantitative comparison and the result discussion. We also discuss what makes some information a good semantic for general propose, how to learn and extract it, and how to infer it based on the users' preference.

### 4.1 Datasets

We use two datasets to run the experimental evaluation of the proposed methodology. Due to the limited amount of available controlled and labeled data regarding specific semantic content, were created two sets of videos using wearable devices. The aim of the first Dataset is to evaluate the amount of semantic content kept in the accelerated video and how it is related with the semantic load of the original video. Therefore, the Semantic Dataset is composed of smaller videos ( $\sim 5$  minutes per video) with controlled percentage of frames containing semantic content. On the other hand, the Dataset of Multimodal Semantic Egocentric Videos (DoMSEV) is composed of lengthy ( $\sim$  one hour per video) and unconstrained videos combined with multimodal data. The DoMSEV dataset is used to stress the methodology testing the scalability and running times.

#### 4.1.1 Annotated Semantic Dataset

This dataset is publicly available<sup>1</sup> and it was first presented in the work of Silva et al. [2016]. The videos composing this dataset were recorded focusing on managing the

---

<sup>1</sup><https://www.verlab.dcc.ufmg.br/semantic-hyperlapse/epic2016-dataset/>



**Figure 4.1.** Examples of the proposed controlled Semantic Dataset. Frames in the first row represent the videos of the Biking category. Frames in the second row represent the videos of the Walking category. Frames in the third row represent the videos of the Driving category.

total of frames containing semantic information. Hereinafter this dataset is referred to as Semantic Dataset, and it is composed of 11 sequences recorded while performing everyday activities, such as Biking, Driving, and Walking. Table 4.1 presents the complete information for each video composing the Semantic dataset and Figure 4.1 depict random samples. All the sequences are labeled as: 0p, for videos with approximately no semantic information (Biking 0p, Driving 0p, and Walking 0p); 25p, for the videos

**Table 4.1.** Information about videos composing the proposed Semantic controlled Dataset. Duration is the length of the video before the acceleration. In Camera column, Hero stands for the GoPro<sup>®</sup> line product.

Videos	Info					
	Semantic (%)	Duration (mm:ss)	Mount	Camera	Image Resolution	FPS
Biking_0p	0%	4:59	Helmet	Hero3+	1280 × 720	60
Biking_25p	25%	9:29	Helmet	Hero3+	1920 × 1080	30
Biking_50p	50%	7:29	Helmet	Hero3+	1280 × 720	60
Biking_50p_2	50%	4:08	Helmet	Hero3+	1280 × 720	60
Driving_0p	0%	5:15	Head	Hero3+	1920 × 1080	30
Driving_25p	25%	4:26	Head	Hero3+	1920 × 1080	30
Driving_50p	50%	5:45	Head	Hero3+	1920 × 1080	30
Walking_0p	0%	4:34	Head	Hero3+	1920 × 1080	30
Walking_25p	25%	6:06	Head	Hero3+	1920 × 1080	30
Walking_50p	50%	6:25	Head	Hero3+	1920 × 1080	30
Walking_75p	75%	8:36	Head	Hero3+	1920 × 1080	30

containing relevant semantic information in 25% of its frames (Biking 25p, Driving 25p, and Walking 25p); 50p, for the ones with around a half of their frames has some semantic content (Biking 50p, Biking 50p2, Driving 50p, and Walking 50p); and 75p, for the videos with 75% of their frames containing relevant semantic (Walking 75p). It is worth noting that even when video belongs to the class 0p, it still contains semantics on its frames. The reason of being classified as 0p is mainly because it does not have a minimum number of frames with high semantic score. The semantic information labeled in this dataset is faces for Walking videos, and pedestrian for Biking and Driving ones.

### 4.1.2 Multimodal Semantic Egocentric Videos

Aside from the Semantic Dataset, to test the running times and scalability of egocentric methodologies, we proposed an unrestricted 80-hour Dataset of Multimodal Semantic Egocentric Videos (DoMSEV). The videos compositing this dataset cover a wide range of activities such as shopping, recreation, daily life, attractions, party, beach, tourism, sports, entertainment, and academic life. The recording conditions vary in light (from sunny day to night, and also artificial lights), scenes (indoor/outdoor), places (from calm natural environment to crowded urban spaces), camera mounting (head, helmet, and chest), capturing device (RGB-D sensor and commercial egocentric cameras), and recorders varying in gender, age, height, and preferences. All details mentioned earlier are annotated for the videos.

The multimodal data was recorded using either a GoPro Hero™ camera or a built setup composed of a 3D Inertial Measurement Unit (IMU) attached to the Intel Realsense™ R200 RGB-D camera. Figure 4.2 shows the setup used, a few examples of frames from the videos, and the fields used to label the video and the frames. Table 4.2 exhibits the videos information, videos entitled ‘A\_c’ were recorded simultaneous to the video ‘A’ (not including suffix ‘\_c’) but with the Point of View set in the user’s chest.

The recorders labeled the videos informing the scene where the segments were taken (*e.g.*, indoor, urban, crowded environment, *etc.*), the activity performed (*e.g.*, walking, standing, browsing, driving, biking, eating, cooking, observing, in conversation, *etc.*), if something caught their attention and when they interacted with some object. Example of labels are depicted in Figure 4.2. Also, we create a profile for each recorder representing their preferences over the 80 classes of the YOLO classifier [Redmon and Farhadi, 2016] and the 48 visual sentiment concepts defined by Sharghi et al. [2017]. To create the recorders’ profile, we asked them to indicate their interest in each



**Figure 4.2.** Sample and labels of the Dataset of Multimodal Semantic Egocentric Videos (DoMSEV). Top-left: setup used to record videos with RGB-D camera and IMU. Top-right: frame samples from DoMSEV. Bottom: Annotated information for videos and frames. The symbol  $\circ$  indicates the possible values for the respective annotation.

class and concepts in a scale from 0 to 10.

Table 4.2 summarizes the diversity of sensors, camera mounting, length of the videos, and activities that can be found in the dataset. 3D model for printing the built setup, and the dataset are publicly available <sup>2</sup>.

## 4.2 Competitors

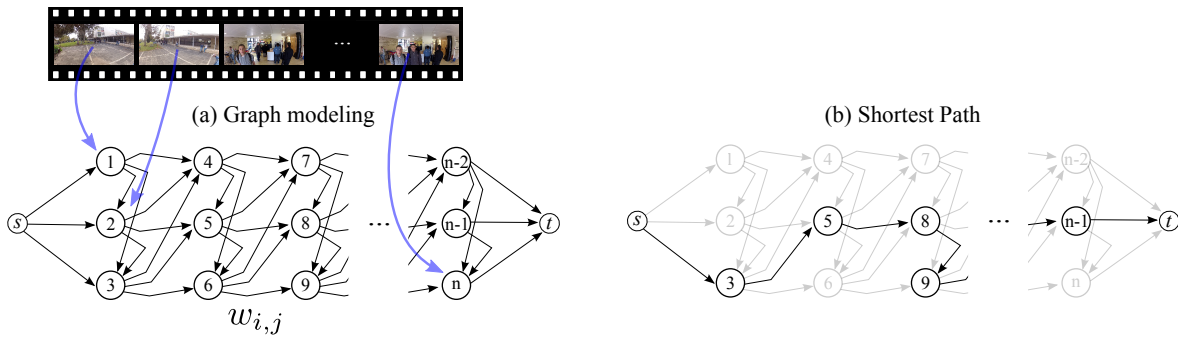
In this Section, we present the competitors used to perform the experimental evaluation of the proposed methodology. The first methodology is the graph-based adaptive frame selection EgoSampling (ES) proposed by [Poleg et al., 2015], which is a Hyperlapse tech-

<sup>2</sup><https://www.verlab.dcc.ufmg.br/semantic-hyperlapse/cvpr2018-dataset/>

**Table 4.2.** Information about videos in the proposed Multimodal dataset. Duration is the length of the video before the acceleration. In Camera column, RS200 stands for RealSense™R200 by Intel® and Hero is a GoPro® line product. Following abbreviation was used in Videos column due to the space limitation: Academic Life (Aca), Attraction (Att), Beach (Bea), Daily Life (Dai), Entertainment (Ent), Party (Par), Recreation (Rec), Shopping (Sho), Sport (Spo), and Tourism (Tou).

Videos	Duration (hh:mm:ss)	Info			GPS IMU Depth	Videos	Duration (hh:mm:ss)	Info			GPS IMU Depth	
		Mount	Camera					Mount	Camera			
Aca_01	00:26:10	head	Hero4		✓	✓	Ent_01	00:14:14	head	Hero4	✓	✓
Aca_02	00:45:08	chest	Hero5		✓	✓	Ent_02	00:18:50	chest	Hero5	✓	✓
Aca_03	00:36:38	helmet	Hero4				Ent_03	01:01:50	chest	Hero5	✓	✓
Aca_04	01:04:12	head	Hero5				Ent_04	01:09:06	helmet	RS200	✓	✓
Aca_05	00:33:11	head	Hero5		✓	✓	Ent_05	01:00:54	helmet	RS200	✓	✓
Aca_06	01:39:24	head	Hero5		✓	✓	Ent_05_c	00:55:25	chest	Hero5	✓	
Aca_07	00:45:02	helmet	Hero5		✓	✓	Ent_06	01:21:54	helmet	RS200	✓	✓
Aca_08	01:11:33	head	Hero5		✓	✓	Ent_06_c	01:36:48	chest	Hero5	✓	✓
Aca_09	01:02:53	helmet	RS200		✓	✓	Ent_07	01:19:47	helmet	RS200	✓	✓
Aca_10	02:04:33	head	Hero5		✓	✓	Ent_07_c	02:02:08	chest	Hero5	✓	✓
Aca_11	01:02:04	hand	Hero4				Par_01	01:02:32	chest	Hero5	✓	✓
Aca_12	01:03:31	chest	Hero5		✓		Rec_01	01:19:05	helmet	Hero4		
Aca_13	00:47:14	helmet	RS200		✓	✓	Rec_02	01:30:40	head	Hero5	✓	✓
Aca_13_c	00:43:37	chest	Hero5		✓	✓	Rec_03	00:57:39	helmet	Hero4		
Att_01	01:25:55	helmet	Hero4				Rec_04	02:15:15	helmet	Hero5	✓	✓
Att_02	01:31:10	chest	Hero5		✓	✓	Rec_05	01:11:45	chest	Hero5	✓	✓
Att_03	01:31:05	head	Hero5		✓	✓	Rec_06	01:03:42	head	Hero5	✓	✓
Att_04	01:11:21	head	Hero5		✓	✓	Rec_07	01:47:44	helmet	Hero4		
Att_05	00:57:10	head	Hero5		✓	✓	Rec_08	01:44:15	shoulder	Hero5	✓	✓
Att_06	00:46:54	head	Hero5		✓	✓	Rec_09	00:48:36	helmet	Hero4		
Att_07	01:30:25	chest	Hero4				Rec_10	00:49:02	helmet	Hero4		
Att_08	00:32:41	chest	Hero5		✓	✓	Rec_11	00:46:04	chest	Hero5	✓	✓
Att_09	01:03:02	helmet	RS200		✓	✓	Rec_12	00:59:01	helmet	Hero4		
Att_09_c	00:52:43	chest	Hero4		✓		Sho_01	00:54:06	helmet	Hero5	✓	✓
Att_10	00:59:09	helmet	RS200		✓	✓	Sho_02	00:50:27	chest	Hero4		
Att_11	01:17:20	helmet	RS200		✓	✓	Spo_01	00:51:56	head	Hero5	✓	✓
Att_11_c	01:08:46	chest	Hero5		✓	✓	Spo_02	00:43:20	head	Hero5	✓	✓
Att_12	01:28:03	chest	Hero5		✓	✓	Spo_03	02:22:21	head	Hero5	✓	✓
Att_13	00:35:21	helmet	RS200		✓	✓	Spo_04	01:01:39	chest	Hero4		
Att_14	00:40:35	helmet	RS200		✓	✓	Tou_01	00:55:35	chest	Hero4		
Att_14_c	00:46:35	chest	Hero5		✓	✓	Tou_02	02:22:52	head	Hero5	✓	✓
Bea_01	00:39:32	head	Hero3				Tou_03	00:41:40	helmet	RS200	✓	✓
Bea_02	01:41:39	head	Hero3				Tou_04	01:46:38	helmet	RS200	✓	✓
Dai_01	01:16:45	head	Hero5		✓	✓	Tou_05	00:59:43	head	Hero5	✓	✓
Dai_02	01:33:39	head	Hero5		✓	✓	Tou_06	01:25:17	chest	Hero4		
Dai_03	01:12:34	head	Hero5		✓	✓	Tou_07	01:05:03	head	Hero5	✓	✓
							Tou_08	01:01:03	head	Hero5	✓	✓

nique following the taxonomy proposed in this dissertation. The second competitor is the state-of-the-art Hyperlapse methodology Microsoft Hyperlapse (MSH) [Joshi et al., 2015] concerning the visual smoothness of the accelerated video. We also compare with the graph-based Semantic Hyperlapse Semantic Fast-Forward and Stabilized Egocentric Video (FFSE) [Silva et al., 2016], the state-of-the-art method regarding amount of semantics kept in the final video.



**Figure 4.3.** Graph-based Adaptive Frame Sampling. (a) Graph modeling considering  $\tau_{max}$  equals 3, meaning that every frame is connected to its three consecutive frames. (b) Example of a shortest path, frames related to nodes in this solution compose the final video.

We also extended the methodology proposed in the work of Silva et al. [2016] changing the temporal segmentation and speed-up definition by our proposed Multi-Importance approach. This method is referred as Ours/Graph hereinafter, and it is used to evaluate and validate the proposed Multi-Importance semantic analysis. Following we present the graph modeling and details about the parameter setting.

### 4.2.1 Graph-based selection

Graph-based solution for adaptive frame sampling was first proposed in the work of Poleg et al. [2015] and successfully applied in the works of Ramos et al. [2016]; Silva et al. [2016]; Ramos [2017]; Halperin et al. [2017]; Wang et al. [2018].

For each segment created in the Temporal Segmentation step described in Section 3.1.2, we model a Directed Acyclic Graph with frames as nodes and frame transitions as edges. Every nodes is connected using weighted directed edges to its  $\tau_{max}$  nodes, related to the subsequent frames – Figure 4.3-(a). The weight  $w_{i,j}$  of the edge connecting the  $i$ -th node to  $j$ -th node is given by the linear combination of terms related to frame transition, as shown in Equation 4.1. These terms are: instability  $I_{i,j}$ , appearance  $A_{i,j}$ , velocity  $V_{i,j}$ , and semantic  $S_{i,j}$ .

$$w_{i,j} = (\lambda_I \cdot I_{i,j} + \lambda_V \cdot V_{i,j} + \lambda_A \cdot A_{i,j} + \lambda_S \cdot S_{i,j}) \cdot \left\lceil \frac{(j-i)}{F} \right\rceil. \quad (4.1)$$

The last component of this equation is a weighting factor that enhances transitions between frames with lower distance and  $F$  is the speed-up rate applied to the graph which the edge belongs. The coefficients  $\lambda$  are regularization factors for the respective cost terms.

The  $I_{i,j}$  term models the instability of the transition from the  $i$ -th frame to the  $j$ -th frame computing the motion direction of each frame by estimating the Focus of Expansion. The term  $V_{i,j}$  controls the sense of speed in the output video by skipping more frames where the camera motion is low and skipping less when the motion is high. A video with consecutive dissimilar images indicates that the camera is unstable. Therefore, a visually pleasant video is composed of visually similar frames; this similarity is modeled as  $A_{i,j}$  calculating the Earth Mover Distance [Pele and Werman, 2009] between the color histograms of the  $i$ -th and  $j$ -th images. We refer the reader to the work of Poleg et al. [2015] to an in-depth description of each component. The term  $S_{i,j}$  is used to penalize the transitions that are not composed of frames with relevant semantic information, and computed as:

$$S_{i,j} = \frac{1}{S_i + S_j + \epsilon}, \quad (4.2)$$

where  $S_x$  is the semantic score defined in Equation 3.1 for the  $x$ -th frame, and  $\epsilon$  avoids dividing by zero when both scores are null.

The adaptive frame sampling is described as the problem of finding the shortest path on this modeled graph. For this, a source node is connect to the  $\tau_{max}$  first nodes, and a target node is connected to the last  $\tau_{max}$  frames – Figure 4.3. Weights connecting both source to nodes, and nodes to target, are set to 0. All frames compositing the shortest path on the graph modeled to the video segment are added to the set of selected frames  $\mathcal{S}$  to produce the final video.

**Parameter Setting.** A drawback of the graph based Frame Sampling methods is the number of parameters to be set by the user. Equation 4.1 has a total of four parameters highly related to the input video, and with a large search space, once these values assume continuous values. Similar to the parameter setting of the speed-up regularization terms, the configuration of four parameters demands user knowledge and effort. In this case, it is highly probable that the user will not find the right parameters. Using fixed parameters as done in the work of Poleg et al. [2015] does not lead to the best result, as can be confirmed by analyzing their results (EgoSampling) in Section 4.5.

To address this issue, we set the values of the terms  $\lambda_I$ ,  $\lambda_V$ ,  $\lambda_A$  and  $\lambda_S$  in Equation 4.1 by applying an automatic and user-free parameter setting using the Particle Swarm Optimization (PSO) algorithm. The fitness function used is defined by Equa-

tion 4.3:

$$\text{fitness}_{\lambda_I, \lambda_V, \lambda_A, \lambda_S} = \frac{J}{Max_J} + \left| \frac{\widehat{L} - E_L}{E_L} \right| + \frac{\widehat{S}^* - Semantics}{\widehat{S}^*}, \quad (4.3)$$

where  $J$  is the jitter of the generated fast-forward video,  $Max_J$  is the maximum possible jitter for the video,  $E_L$  is the expected number of frames,  $\widehat{L} = L/\widehat{F}_d$  is the final video length,  $L$  is the original video length, and  $\widehat{S}^*$  is the maximum value for the semantic score of the fast-forward video.

The *Semantics* value represents the semantic content of the generated fast-forward video. It is the sum of the semantic score computed by Equation 3.1 using all frames. We compute the jitter as the magnitude of the mean deviation of the Focus of Expansion locations along the selected frames. The maximum possible jitter is the jitter of a hypothetical video in which for every frame the Focus of Expansion is as far as possible from the previous.

### 4.3 Parameters Setup

Parameters of our methodology were empirically set following a careful procedure to achieve satisfactory overall results.

**Ad hoc Semantic Analysis** – Section 3.1.1 – For the experiments on evaluating the semantic content, we used the NPD Face Detector [Liao et al., 2016] and a pedestrian detector [Dollár, 2016] as the semantic extractors. Values 60 and 100 were used for  $\text{threshold}_{c_k}$  in Equation 3.2 as minimum confidence to accept a face detection and pedestrian detection, respectively.

**Temporal Segmentation** – Section 3.1.2 – We filtered the semantic profile using a Gaussian function with standard deviation  $\sigma = 5 \cdot FPS$ , where FPS stands for Frames Per Second. Segments smaller than 5 seconds were discarded by connecting them to adjacent segments since short ranges would result in a flash on accelerated videos.

**Graph-based Frame Sampling** – Section 4.2.1 – We set the number of nodes outgoing edges  $\tau_{max}$  as 100. In the calculation of the edge semantic cost term (Equation 4.2),  $\epsilon$  was set as 1.

**Sparse-based Frame Sampling** – Section 3.2.1 – We used  $SpF = 2$  during the Smoothing Frame Transitions, in a manner that half of the frames compositing the



final video were sampled to reconstruct well the context of the original video, and the other half to smooth the transitions.

**Video Stabilization** – Section 3.3.1 – The size of the segments for selecting the master frames was defined as  $\gamma = 4$ . The drop area  $da$  was set to  $dp\% = 50\%$  of the frame and the crop area  $ca$  was set as  $cp\% = 90\%$  of the frame total area. In Equation 3.14, we used  $\eta = 0.5$  and  $\sigma = 10$ .

## 4.4 Evaluation criterion

We perform the experimental evaluation of the proposed methodology by a quantitative analysis regarding four aspects: amount of semantic information retained in the accelerated video, deviation of achieved speed-up based on the required value, visual instability, and temporal discontinuity of the output video.

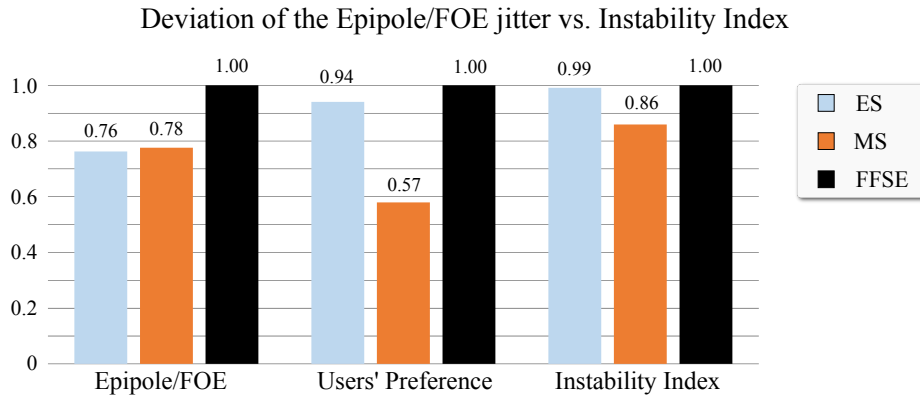
**Semantics.** For the Semantic evaluation, use the *ad hoc* definition of semantics that states the relevant information as pedestrian in videos with high speed camera forward movement, *e.g.*, videos recorded when Biking or Driving, and face in the remaining sequences. The semantic index of the accelerated video is given by the ratio between the sum of the semantic content in each frame of the final video and the maximum possible semantic value for the original accelerated using the required speed-up rate  $F_d$  as defined in Equation 4.4:

$$SV = \frac{\sum_{i=1}^{m_c} S(f_{s_i})}{\sum_{i=1}^{m_r} S(f_{t_i})}, \quad (4.4)$$

where  $m_c$  is the number of frames in the accelerated video,  $m_r$  is the number of frames needed to create a accelerated video with the required speed-up  $F_d$ ,  $f_{s_i}$  stands for the  $i$ -th frame of the accelerated video, and  $f_{t_i}$  stands for the top- $i$  ranked frame of the original video regarding the semantic content [Ramos et al., 2016].  $S(f)$  stands for the semantic content of the frame  $f$  and is defined by Equation 3.1.

Semantic value is larger for videos composed of frames that have semantic regions with a higher confidence assigned by the classifier, a larger area, and located in the central part of the image.

**Visual Instability.** Most of the fast-forward methodologies either use qualitative metrics, based on subjective human evaluation [Kopf et al., 2014; Joshi et al., 2015], or the quantitative evaluation by analyzing the deviation of the epipole/Focus of Expansion (FOE) jitter [Poleg et al., 2015] in the produced video. However, we demonstrated



**Figure 4.4.** Visual instability result of the user study. (Left) Mean value of instability calculated using the Epipole Jitter deviation over all sequences for all three techniques. (Center) Mean value related to the users’ opinion concerning the visual instability of produced accelerated videos. (Right) Average value of visual instability calculated by the proposed Instability metric (Equation 4.5). The result of the proposed metric reflect better the users’ opinion than the Epipole jitter deviation.

through a user study that the deviation of the epipole jitter metric occasionally assigned better scores for shakier videos, contradicting users’ opinion. Among the possible reasons for the bad performance of the epipole jitter metric, we list the epipole estimation error and the weakness of relying on a single value. Therefore, we propose a metric to evaluate the smoothness of the final produced video concerning the whole frame by analyzing the pixel intensity standard deviation in a sliding buffer. We get inspiration on the qualitative evaluation employed by Joshi et al. [2015], where they used a side-by-side comparisons and standard deviation frames of a few consecutive images. The metric is defined as follows:

$$I = M \left( \frac{1}{n} \cdot \sum_{i=1}^n \frac{\sum_{j \in B_i} (f_j - \bar{f}_i)^2}{(n_B - 1)} \right), \quad (4.5)$$

where  $n$  is the number of frames in the video,  $B_i$  is the  $i$ -th buffer composed by  $n_B$  temporal neighbor frames,  $f_j$  is the  $j$ -th frame of the video,  $\bar{f}_i$  is the average frame of the buffer  $B_i$ ,  $M(\cdot)$  is a function that returns the mean value for the pixels of a given image and  $I$  indicates the instability index of the video. A smoother video yields a smaller  $I$  value. We use buffer size equals 7 for all experiments.

Figure 4.4 depicts the users’ opinion gathered in the user study performed concerning the visual instability of the produced videos. To create the database used in the user study, we collected the output videos produced by the acceleration techniques:

EgoSampling (ES) [Poleg et al., 2015]; Microsoft Hyperlapse (MH) [Joshi et al., 2015], and Fast-Forward Based on Semantic Extraction (FFSE) [Silva et al., 2016]. We run the techniques in 9 egocentric videos from the EgoSequences Dataset [Poleg et al., 2015], with the 10 as the required speed-up, producing accelerated videos of average 35-second length. During the user study, we asked 33 subjects to watch the (unlabeled) videos and grade the video instability with respect to its smoothness in an assessment questionnaire. Higher values indicate shakier videos. Epipole jitter results state ES produce smoother videos when compared to the MSH, which is the state-of-the-art regarding visual smoothness, as depicted by the users’ opinion. Figure 4.4 shows that the proposed Instability reflects the subjects’ preferences.

**Speed-up Deviation.** Speed-up metric is given by the absolute difference between the achieved speed-up rate and the required value  $F_d$ . The achieved speed-up is the ratio between the number of frames in the original video and in its accelerated version (Equation 3.15). In this dissertation, we used required speed-up  $F_d = 10$  for all experiments.

**Temporal Discontinuity.** To measure the video *Discontinuity*, we propose to use the Root-Mean-Square Error (RMSE) over the selected frames jumps and the required speed-up rate for that video, as follows

$$D = \sqrt{\frac{\sum_{i=2}^m ((f_{s_i} - f_{s_{i-1}}) - F_d)^2}{(m - 1)}}, \quad (4.6)$$

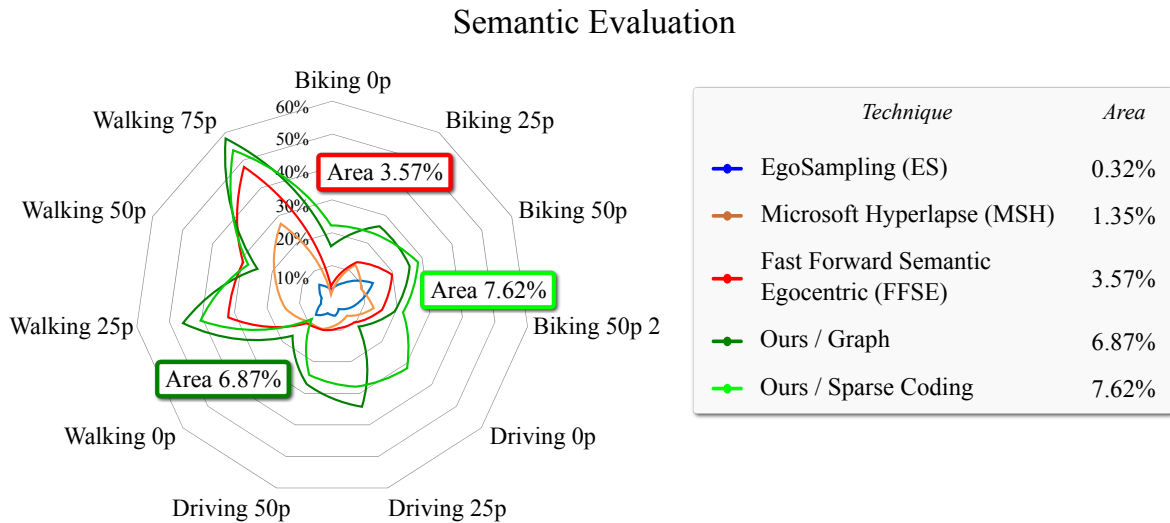
where  $f_{s_i}$  is the index of  $i$ -th selected frame on the original video,  $F_d$  is the required speed-up rate, and  $m$  is the number of frames in the accelerated video. Higher values indicate that the accelerated video contains long jumps, which creates visual gaps.

## 4.5 Results

In this section, we present the quantitative results concerning the experimental evaluation of the proposed method.

### 4.5.1 Multi Importance Semantic Analysis

In order to perform a quantitative evaluation, we use the *Ad hoc* definition of semantic presented in Section 3.1.1.1. Pedestrian detection is used as semantic information in videos with high speed camera movement, *e.g.*, Biking and Driving, and face detection



**Figure 4.5.** Semantic content retained in the accelerated videos produced by ES, MSH, FFSE, and our methods in the Semantic Dataset. The use of Multi-Importance approach leads our both graph and sparse based methodologies to keep twice the semantic information retained by the best competitor (FFSE). Higher values are better.

to videos with low speed camera, *e.g.*, Walking. To demonstrate the invariability of the results regarding the frame sampling step, the Multi-Importance approach was applied in both graph-based and sparse-based adaptive frame sampling.

Figure 4.5 shows the fraction of the semantic content retained from the maximum value that can be present in a accelerated video. We calculate this maximum by summing over the  $m$  top-ranked frames with relation to the semantic content, where  $m$  is the ratio between the original video length  $n$  and the required speed-up rate  $F_d$  (we refer the reader to Section 4.4 for more details).

The combination of the multi-importance approach with an adaptive frame selection creates accelerated videos with even more emphasis in the semantic segments, leading our methodology to outperform the competitors, excepting in “Walking 50p”. Analyzing experiments “Driving 0p” and “Driving 25p”, we manage to keep around 30% of the possible semantic information, while the best competitor takes around 10%, what means three times more semantic information retained in the final video. Sequences “Walking 75p” and “Walking 50p” are failure cases, the Multi-Importance approach did not increase the amount of semantic retained. A single semantic segment was created in these experiment due to low variation in the semantic profile of the videos.

Our methodology manages to keep around 2 times more semantic content than the best competitor (FFSE), which is also a semantic fast-forward method. In comparison to the MSH, which is the best non-semantic competitor, the average semantic

information kept is 5 times higher.

### 4.5.2 Semantic based on users' opinion

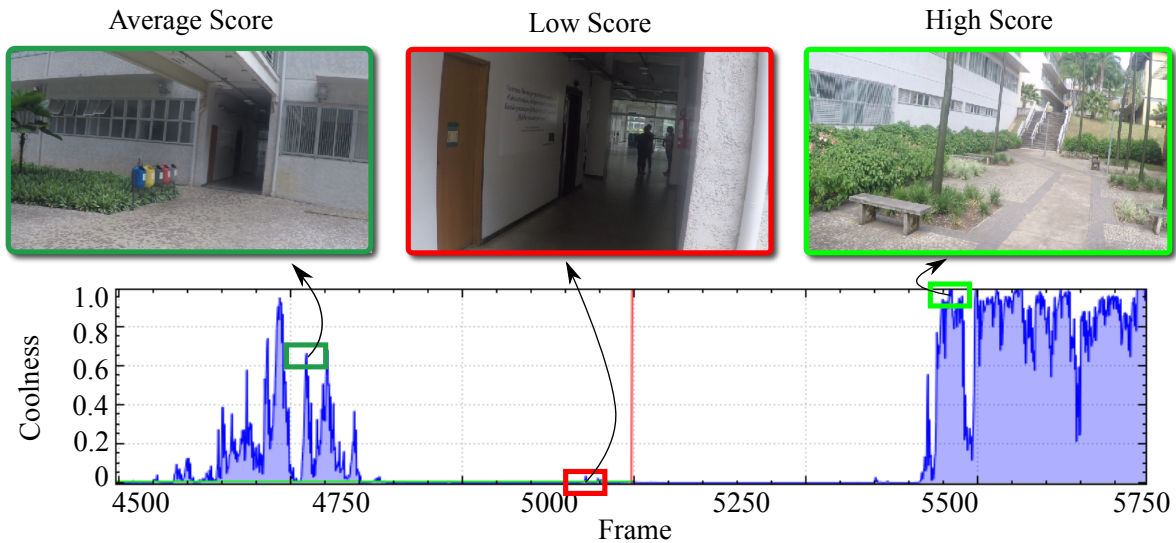
In Section 4.5.1, due to the need of establishing a ground truth for experimental evaluation and exhibition proposes, we perform the experiments using a predefined semantic: face detection for videos with slow movements and pedestrian for the others. However, semantics is more than faces or pedestrians, by definition it is everything that has significance for someone. In this Section, we present the results of the proposed methodology to assign semantic score to video frames based on the user's attention.

Our methodology assign a semantic score to frames using the preference of the user from web video statistics. We train a CNN to learn how to identify a "Cool" frame based on the images composing the videos which the user likes. Besides of being applied to general purpose, this approach can be user specific, *i.e.*, learning the interest of a single person.

After the training, the CoolNet is capable of rate the video frames based on the "Cool" concept learned from the training data. Therefore, the user does not need to specify the semantic information to produce the *Hyperlapse* video, the methodology use the CoolNet to assign relevance to frames instead.

Figure 4.6 depicts network score related to different scenes. Analyzing the assigned scores and their respective images, we inferred that the Network classifies with high score frames with nature elements, *e.g.*, forest, beach, dirty roads and gardens. This preference for natural scenarios is due to the fact that most of the images labeled as "Cool" in the gathered Dataset are related to radical sports and beautiful landscapes. Uniform frames, like indoor looking images, walls, and offices, yield to a low rating. In the left image in Figure 4.6, when the recorder passes through an inside garden, the network attributes an average rating. In the center image, the recorder is walking inside a building hall, which the net considers unattractive. In the right image, the recorder goes to an outside area with trees and gardens, which are highly rated by the net.

Even though the *CoolNet* incorporates user's preferences to estimate the relevance of each frame, it could be not enough to cover all possible semantics. We address this issue by combining semantic extractors, making a linear combination of their output. In this case, the output is a fast-forward video emphasizing segments which have either face or beautiful landscapes, for example. Additionally, since the score for each frame is given by a linear combination, we have the freedom to set which extractor has more influence. The reader is referred to <https://youtu.be/fa0r70LvH8w?t=293> for visual



**Figure 4.6.** Semantic Profile curve of the *CoolNet* for every frame of a sample video. The left image depicts an inside garden, with its medium score. The central image is a building hall, that the *CoolNet* does not consider containing large semantic content. The right image is a garden with an outdoor view, for which *CoolNet* gives the highest scores.

results of the accelerated video produced by the *CoolNet* and semantic combination.

### 4.5.3 Smoothing of Speed-up transitions

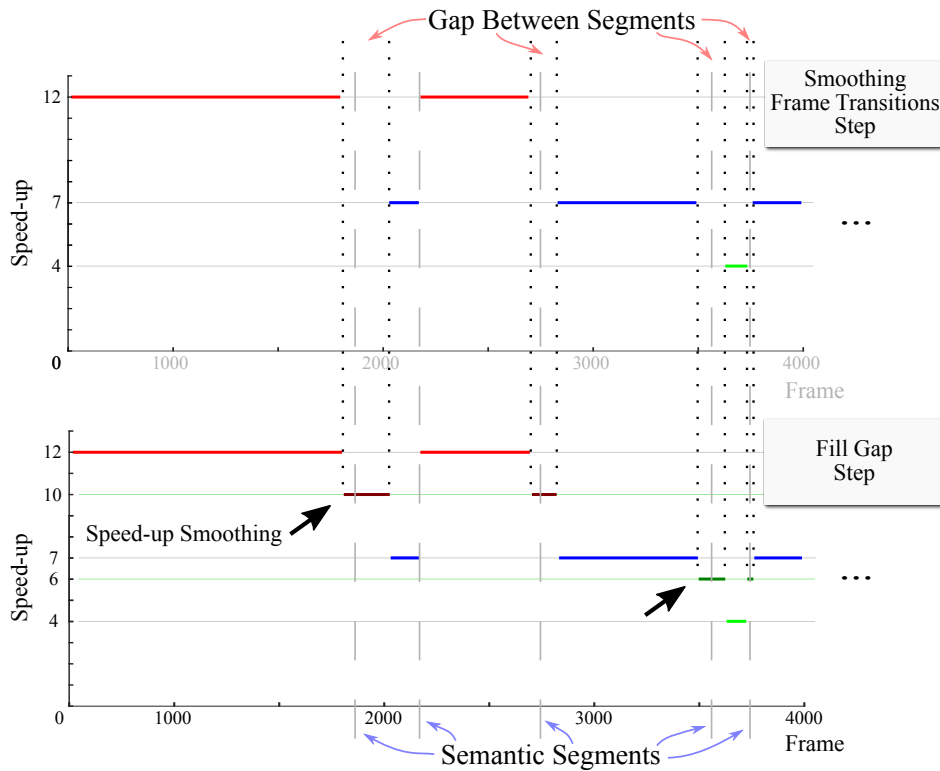
As stated in Section 1.2, one of the problems concerning semantic fast-forward methodologies is the abrupt difference of speed-ups when changing from a semantic segment to a non-semantic one, or vice versa. One drawback of the abrupt difference in speed-up transitions is to create a virtual effect on the final video. In this dissertation, we address this problem of smoothing the speed-up transitions during the Fill Visual Gap between segments processing (Section 3.2.1.2).

We evaluate the speed-up smoothing effect by calculating the Root Mean Square Error (RMSE) between the acceleration rates applied to consecutive segments, as described in Equation 4.7

$$S = \sqrt{\frac{\sum_{i \in \mathcal{SS}} (F_i - F_{i-1})^2}{(|\mathcal{VS}|)}}, \quad (4.7)$$

where  $F_i$  is the speed-up rate applied to the  $i$ -th segment of the video,  $\mathcal{VS}$  is the set of video segments. Higher values indicate that the accelerated video contains abrupt difference of speed-up transitions.

When comparing our frame sampling without applying the speed-up transitions smoothing with our full methodology, the average Root Mean Square Error (RMSE)

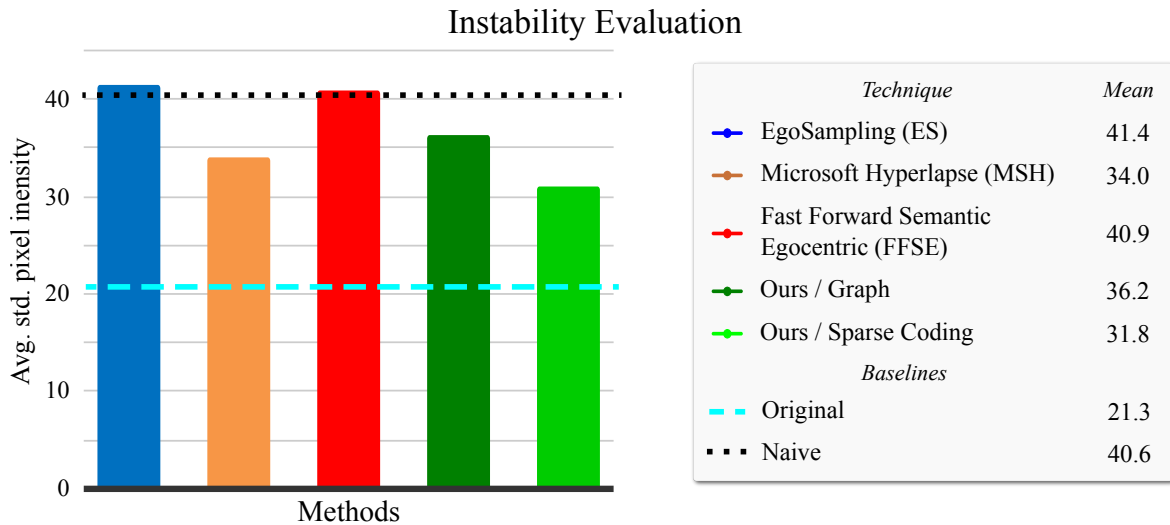


**Figure 4.7.** Smoothing Speed-up transition by the Fill Gap processing. Horizontal lines indicates the ranges where frames were effectively sampled. Top image depict the problem of temporal discontinuities related to sparse-based adaptive frame sampling. Bottom image show frames sampled in the Gap between the segments and the speed-up transition smooth.

values are 5.7 and 3.6 before and after applying the smoothing step on the Semantic Dataset, respectively. Experiments performed in the DoMSEV dataset resulted in values 8.4 and 4.5 before and after applying the speed-up transition step. Figure 4.7 depicts the speed-up rates applied in the video segments, horizontal lines indicates the ranges where frames were sampled during the frame sampling. Top image depicts the problem related with sparse-based adaptive frame sampling, gaps are created between the last frame of a segment and the first frame of the following segment. Bottom image depicts the effect of the Fill Gap step smoothing the speed-ups transitions.

#### 4.5.4 Sparse-based Semantic Hyperlapse

The reported results regarding the Semantic Evaluation show that the sparse-based frame sampling applying the Multi-Importance approach outperforms the state-of-the-art semantic hyperlapse technique (Figure 4.5). We extend the experimental evaluation of the proposed sparse-based sampling to the analysis of Visual Instability, Temporal



**Figure 4.8.** Experimental evaluation regarding the Temporal Discontinuities. Bars show the average Instability Index over all videos in the Semantic Dataset achieved by each methodology. Our sparse-based framework outperforms the competitors. Lower values indicate more stable videos.

Discontinuity, Speed-up Deviation, and Processing time.

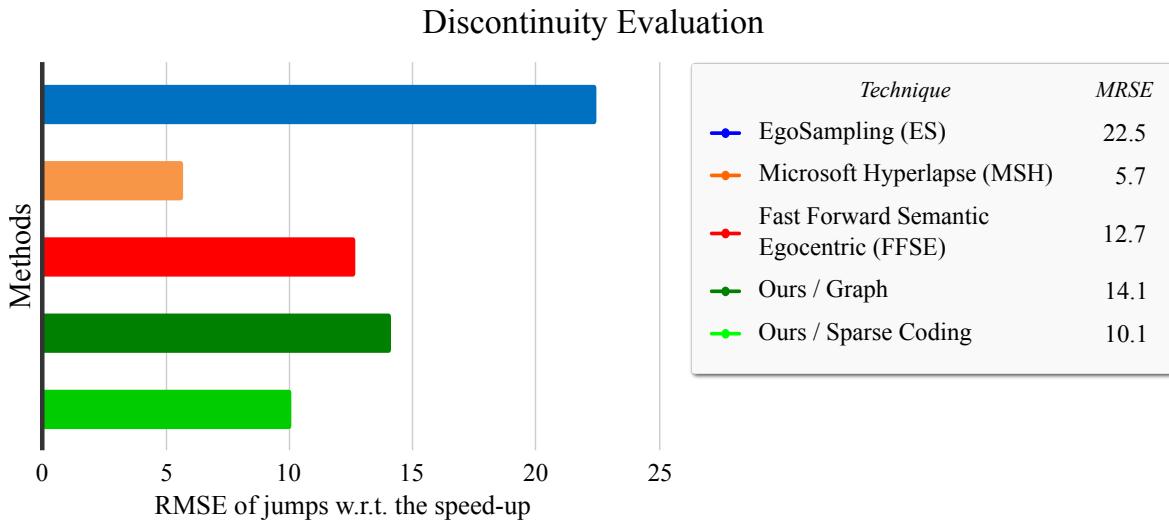
The results for Instability are presented as the mean of the instability indexes calculated over all sequences using Equation 4.5 – Figure 4.8, lower values are better. Cyan dashed line stands for the average instability index among the original videos, and the black dotted line stands for the average instability index of the naively accelerated videos. Ideally, it is better to yield an instability index as close as possible to the original video. The chart shows that our method created videos as smooth as the state-of-the-art method MSH.

Figure 4.9 shows the average Root Mean Square Error (RMSE) of the jumps regarding the required speed-up rate  $F_d$ . MSH has an advantage over other competitors since it is a non-semantic hyperlapse, which means that the whole video will be accelerated in a rate very close to the required speed-up. On the other hand, semantic hyperlapse techniques eventually will have segments accelerated with a lower speed-up to create emphasis, and with a higher play rate to compensate the lower speed-ups.

Comparing only semantic acceleration methodologies, our proposed sparse-based framework achieved the lowest value. We assign this improved results to the methodological steps Smoothing Frame Transitions and Fill Gap between Segments, since the first use the distance between frames to smooth a transition, and the second step fill the temporal gap resulting of the frame sampling. We refer the reader to Section 4.6.4 for more experiments related to the Fill Gap step.

Regarding speed-up analysis, the average absolute difference over all experiments





**Figure 4.9.** Experimental evaluation regarding the Visual Instability. Bars show the average RMSE between the frame jumps and the required speed-up rate over all videos in the Semantic Dataset. The best value is achieved by the MSH, which is a non-semantic Hyperlapse. Regarding semantic methodologies, the proposed sparse-based frameworks achieved the best value. Lower values are better.

over the Semantic Dataset was smaller than the one for the proposed graph-based methodology (0.8) and sparse-based (0.9). Other competitors performed poorly in this criteria: ES (11.0), FFSE (3.3), and MSH (1.2).

#### 4.5.5 Processing Time

Our proposed sparse-based methodology outperform the competitors analyzing the results concerning amount of retained semantic information, visual stability, proximity of achieved speed-up to the desired value, and time to run the adaptive frame sampling. To stress the methodology, we ran a detailed performance assessment in the unconstrained DoMSEV dataset comparing our methodology against the second best competitor regarding semantic retained in the final video, which is the FFSE using our Multi-Importance approach (Section 4.2.1). Results are showed in Tables 4.3 and 4.4. As can be seen in the mean values regarding the whole dataset present in the end of Table 4.4, our graph-based method outperforms by a small margin the sparse-based frame sampling in the metric Semantic retained. Analyzing the Visual Instability, sparse-based approach leads, it means that the frame sampling neglected the gain in semantic to create a more smooth video. The speed-up deviation of the sparse-based approach is tree times lower than the graph-based solution.

However, the highlighting result when comparing the sparse-based approach to

**Table 4.3.** Comparison between the sparse and graph based approaches in the unconstrained DoMSEV w.r.t. semantic retained, visual instability, speed-up achieved, and processing time of frame sampling step.

Videos	Semantic <sup>1</sup> (%)		Instability <sup>2</sup>		Speed-up <sup>2</sup>		Time <sup>2</sup> (s)	
	Graph	Sparse	Graph	Sparse	Graph	Sparse	Graph	Sparse
Aca_01	<b>32.6</b>	26.5	46.8	<b>45.1</b>	<b>0.0</b>	0.3	1,477.1	<b>5.1</b>
Aca_02	<b>41.9</b>	28.0	34.5	<b>32.8</b>	<b>0.0</b>	0.4	2,587.2	<b>8.9</b>
Aca_03	<b>32.6</b>	28.0	42.8	<b>41.4</b>	<b>0.0</b>	0.2	2,112.9	<b>9.7</b>
Aca_04	<b>28.0</b>	23.7	<b>34.5</b>	35.1	0.4	<b>0.2</b>	3,882.0	<b>10.4</b>
Aca_05	<b>28.7</b>	26.9	42.0	<b>39.6</b>	2.2	<b>0.1</b>	1,810.0	<b>6.5</b>
Aca_06	<b>38.1</b>	34.4	36.5	<b>33.1</b>	<b>0.0</b>	0.1	6,976.3	<b>12.6</b>
Aca_07	<b>26.8</b>	21.4	41.5	<b>41.0</b>	<b>0.2</b>	0.8	2,802.9	<b>6.2</b>
Aca_08	<b>24.5</b>	19.5	<b>31.5</b>	34.4	2.4	<b>0.1</b>	4,252.7	<b>18.8</b>
Aca_09	<b>24.7</b>	20.7	50.4	<b>48.4</b>	<b>0.0</b>	0.2	3,298.5	<b>8.7</b>
Aca_10	<b>25.1</b>	24.1	51.6	<b>48.1</b>	1.5	<b>0.2</b>	8,759.3	<b>32.7</b>
Aca_11	20.1	<b>22.7</b>	42.6	<b>31.1</b>	1.0	<b>0.3</b>	3,176.9	<b>10.5</b>
Aca_12	<b>32.2</b>	28.2	33.8	<b>30.8</b>	1.0	<b>0.1</b>	4,373.4	<b>21.9</b>
Aca_13	40.0	<b>46.7</b>	30.0	<b>19.9</b>	2.4	<b>0.2</b>	2,623.4	<b>21.4</b>
Aca_13_c	<b>40.5</b>	19.5	<b>20.0</b>	27.6	<b>0.1</b>	0.2	2,533.3	<b>5.8</b>
Att_01	<b>27.0</b>	22.1	38.1	<b>36.1</b>	<b>0.0</b>	0.4	5,102.3	<b>12.6</b>
Att_02	59.2	<b>66.7</b>	<b>25.6</b>	32.2	<b>0.0</b>	0.2	6,003.6	<b>19.4</b>
Att_03	46.7	<b>75.3</b>	40.3	<b>30.0</b>	4.1	<b>0.0</b>	3,762.0	<b>226.8</b>
Att_04	48.9	<b>50.9</b>	37.9	<b>35.9</b>	<b>0.5</b>	0.6	4,851.0	<b>15.6</b>
Att_05	50.6	<b>51.5</b>	36.1	<b>34.9</b>	1.0	<b>0.1</b>	3,320.9	<b>13.2</b>
Att_06	<b>26.9</b>	21.6	46.4	<b>45.9</b>	1.0	<b>0.4</b>	2,812.6	<b>9.2</b>
Att_07	<b>32.0</b>	26.3	44.8	<b>43.7</b>	<b>0.0</b>	0.8	7,011.0	<b>12.5</b>
Att_08	77.5	<b>81.4</b>	37.8	<b>36.4</b>	1.7	<b>0.3</b>	1,762.0	<b>7.0</b>
Att_09	<b>21.5</b>	12.2	52.4	<b>37.5</b>	<b>0.0</b>	0.7	3,265.1	<b>16.2</b>
Att_09_c	44.4	<b>46.5</b>	<b>39.6</b>	47.0	<b>0.3</b>	0.4	4,458.3	<b>22.7</b>
Att_10	<b>43.7</b>	35.6	45.6	<b>44.8</b>	0.3	<b>0.2</b>	4,043.8	<b>9.6</b>
Att_11	<b>39.2</b>	17.8	35.8	<b>20.6</b>	<b>0.0</b>	0.5	6,025.0	<b>9.4</b>
Att_11_c	<b>31.5</b>	27.3	<b>21.1</b>	36.5	<b>0.0</b>	0.1	5,684.4	<b>13.2</b>
Att_12	<b>53.3</b>	32.8	24.0	<b>21.6</b>	0.7	<b>0.2</b>	7,020.3	<b>13.0</b>
Att_13	<b>31.9</b>	27.3	47.8	<b>46.8</b>	0.2	<b>0.0</b>	2,460.5	<b>7.9</b>
Att_14	<b>24.2</b>	21.3	43.2	<b>28.4</b>	<b>0.1</b>	<b>0.1</b>	2,502.5	<b>6.2</b>
Att_14_c	<b>30.1</b>	24.5	<b>33.8</b>	41.0	<b>0.0</b>	0.4	2,399.6	<b>4.5</b>
Bea_01	<b>26.6</b>	18.8	32.8	<b>30.1</b>	<b>0.0</b>	0.1	5,364.7	<b>15.4</b>
Bea_02	<b>30.5</b>	28.5	<b>29.1</b>	36.3	<b>0.0</b>	0.3	21,922.4	<b>50.9</b>
Dai_01	<b>20.0</b>	16.3	49.0	<b>44.4</b>	2.6	<b>0.2</b>	5,222.0	<b>16.0</b>
Dai_02	25.5	<b>25.7</b>	46.8	<b>35.4</b>	3.5	<b>0.2</b>	5,741.3	<b>65.4</b>
Dai_03	<b>27.2</b>	23.4	33.5	<b>28.0</b>	<b>0.0</b>	<b>0.0</b>	3,868.3	<b>61.8</b>

<sup>1</sup>Higher is better.

<sup>2</sup>Lower is better.

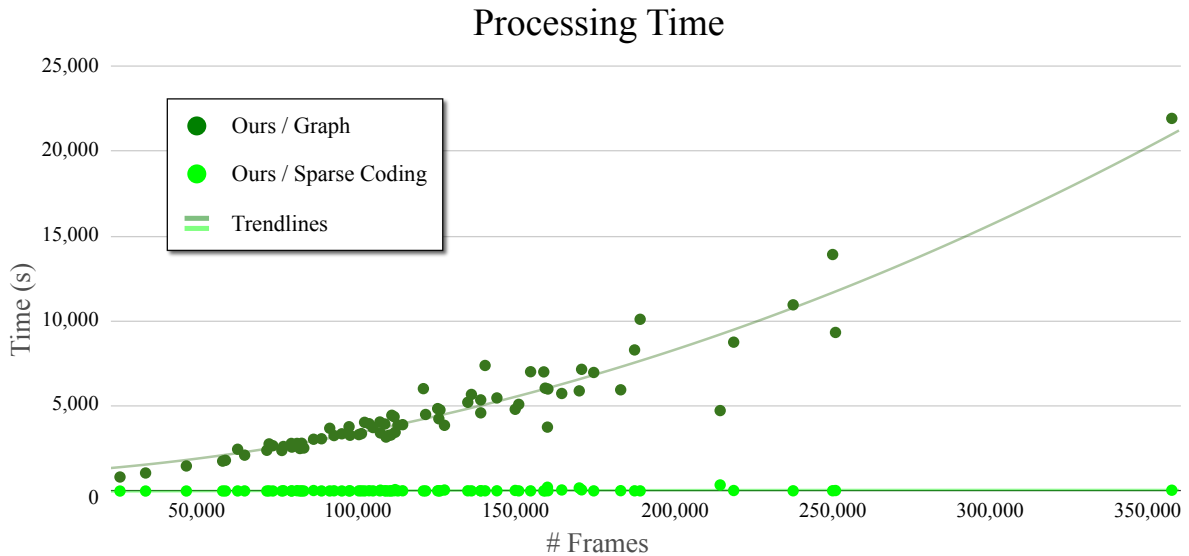
the graph-based is the processing time evaluation regarding the frame sampling process. The average time spent to the sparse-based approach to perform the frame sampling step was 27.8 s, including the steps Weighted Sparse-based Frame Sampling, Smooth Frame Transition, and Fill Gap between segments, against 4,728.4 s of the graph-based approach including the graph modeling, shortest path calculation, and automatic parameter-setting processes. These values represent an average of 0.2 ms per frame for the sparse-based solution against 0.36 ms per frame for the frame-based approach. Figure 4.10 shows the time spent during the frame sampling step related to the length of input videos for all experiments in the DoMSEV.

**Table 4.4.** Continuation of the comparison between the sparse and graph based approaches in the unconstrained DoMSEV w.r.t. semantic retained, visual instability, speed-up achieved, and processing time of frame sampling step. The *Mean* row is regarding the whole dataset, *i.e.*, the average value of the column of both Tables 4.3 and 4.4.

Videos	Semantic <sup>1</sup> (%)		Instability <sup>2</sup>		Speed-up <sup>2</sup>		Time <sup>2</sup> (s)	
	Graph	Sparse	Graph	Sparse	Graph	Sparse	Graph	Sparse
Ent_01	<b>35.9</b>	32.1	<b>24.0</b>	30.6	<b>0.0</b>	0.5	827.2	<b>2.1</b>
Ent_02	<b>64.4</b>	62.7	37.5	<b>32.2</b>	0.3	<b>0.0</b>	1,063.8	<b>3.0</b>
Ent_03	<b>31.4</b>	26.0	31.4	<b>23.9</b>	1.0	<b>0.6</b>	3,945.5	<b>17.4</b>
Ent_04	<b>32.8</b>	29.8	45.5	<b>42.7</b>	<b>0.0</b>	0.1	4,505.9	<b>10.0</b>
Ent_05	<b>33.2</b>	29.4	39.1	<b>21.4</b>	<b>0.0</b>	0.1	3,786.1	<b>8.6</b>
Ent_05_c	<b>23.9</b>	22.1	<b>25.9</b>	34.1	<b>0.0</b>	0.1	4,063.8	<b>13.2</b>
Ent_06	7.3	<b>65.1</b>	38.6	<b>19.9</b>	2.0	<b>0.5</b>	5,894.6	<b>183.0</b>
Ent_06_c	<b>27.5</b>	17.0	22.7	<b>22.1</b>	0.8	<b>0.4</b>	5,478.1	<b>16.8</b>
Ent_07	<b>83.3</b>	68.1	28.7	<b>24.1</b>	<b>0.0</b>	0.2	4,735.2	<b>360.4</b>
Ent_07_c	<b>39.6</b>	13.4	9.4	<b>8.2</b>	<b>0.0</b>	0.2	7,389.0	<b>15.4</b>
Par_01	<b>24.1</b>	19.3	<b>28.5</b>	31.0	<b>0.0</b>	0.6	3,275.9	<b>8.0</b>
Rec_01	<b>22.6</b>	18.5	40.1	<b>39.3</b>	<b>0.0</b>	0.6	4,601.6	<b>16.4</b>
Rec_02	<b>44.8</b>	38.4	<b>43.9</b>	45.3	<b>0.6</b>	<b>0.6</b>	6,054.6	<b>14.9</b>
Rec_03	72.7	<b>76.7</b>	43.6	<b>42.3</b>	<b>0.4</b>	<b>0.4</b>	3,379.9	<b>12.6</b>
Rec_04	<b>25.0</b>	22.7	<b>40.6</b>	43.2	<b>0.0</b>	0.4	10,955.5	<b>16.9</b>
Rec_05	<b>31.7</b>	24.1	28.9	<b>27.7</b>	2.5	<b>0.5</b>	4,775.1	<b>10.5</b>
Rec_06	<b>11.2</b>	10.0	48.2	<b>47.1</b>	4.5	<b>0.0</b>	3,459.6	<b>86.4</b>
Rec_07	<b>22.8</b>	19.8	39.1	<b>37.5</b>	<b>0.0</b>	0.3	10,105.4	<b>12.3</b>
Rec_08	<b>26.3</b>	25.9	38.4	<b>33.5</b>	3.7	<b>0.7</b>	5,957.0	<b>23.3</b>
Rec_09	<b>23.9</b>	20.1	30.7	<b>27.5</b>	<b>0.0</b>	<b>0.0</b>	7,162.1	<b>71.7</b>
Rec_10	67.0	<b>68.8</b>	<b>19.7</b>	24.8	<b>0.0</b>	0.2	3,048.9	<b>32.3</b>
Rec_11	65.2	<b>67.6</b>	<b>11.6</b>	<b>11.6</b>	<b>0.0</b>	0.2	2,802.9	<b>23.7</b>
Rec_12	<b>46.0</b>	40.0	18.5	<b>17.7</b>	<b>0.0</b>	0.3	3,962.0	<b>19.5</b>
Sho_01	<b>25.8</b>	21.3	43.7	<b>43.1</b>	<b>0.0</b>	0.7	3,368.7	<b>6.6</b>
Sho_02	<b>30.4</b>	24.4	43.7	<b>42.1</b>	<b>0.3</b>	0.5	3,076.1	<b>8.3</b>
Spo_01	<b>24.4</b>	22.5	36.6	<b>34.7</b>	<b>0.0</b>	0.5	3,694.5	<b>7.4</b>
Spo_02	<b>12.6</b>	11.4	53.3	<b>47.0</b>	<b>0.0</b>	0.4	2,387.6	<b>4.2</b>
Spo_03	<b>31.6</b>	22.3	37.9	<b>33.8</b>	<b>0.0</b>	0.1	13,915.1	<b>19.5</b>
Spo_04	<b>45.9</b>	40.7	32.7	<b>31.0</b>	<b>0.0</b>	0.7	2,774.9	<b>10.2</b>
Tou_01	62.9	<b>64.7</b>	31.6	<b>29.8</b>	2.1	<b>0.3</b>	3,283.4	<b>14.6</b>
Tou_02	47.2	<b>47.8</b>	54.3	<b>51.7</b>	3.2	<b>0.9</b>	9,331.0	<b>31.9</b>
Tou_03	<b>33.9</b>	31.2	<b>38.8</b>	39.3	<b>0.0</b>	0.3	2,668.3	<b>5.8</b>
Tou_04	<b>29.2</b>	25.3	56.4	<b>54.1</b>	<b>0.1</b>	0.5	8,302.5	<b>15.3</b>
Tou_05	56.8	<b>57.6</b>	33.1	<b>31.1</b>	1.2	<b>0.2</b>	3,735.8	<b>11.6</b>
Tou_06	33.6	<b>34.7</b>	<b>26.4</b>	27.1	6.2	<b>0.2</b>	4,810.6	<b>37.4</b>
Tou_07	42.7	<b>44.8</b>	<b>37.1</b>	40.4	4.5	<b>0.1</b>	3,906.1	<b>17.7</b>
Tou_08	<b>32.7</b>	29.3	<b>29.4</b>	32.2	4.4	<b>0.2</b>	3,419.4	<b>53.0</b>
<i>Total mean</i>	<b>36.0</b>	<i>33.1</i>	<i>36.6</i>	<b>34.6</b>	<i>0.9</i>	<b>0.3</b>	<i>4,728.4</i>	<b>27.8</b>

<sup>1</sup>Higher is better. <sup>2</sup>Lower is better.

It is noteworthy that the graph-based approach runs a parameter setup to adjust 4 parameters and then performs the shortest path for each segment. Our sparse-based approach runs the analytic solution for the minimum reconstruction problem followed by the frame transition smoothing step, and the fill gap between segments steps. Time represented in Y-axis in the chart indicates the execution time of graph-based approach grows exponentially with the number of frames in the input video, while the sparse-



**Figure 4.10.** Processing time analysis of related to the input video length of graph and sparse based approaches. Trendline of each set of points follows a second order polynomial form.

based approach is not influenced by the number of frames in the input video. This  $180\times$  of time speed-up regarding sparse-approach over the graph-solution and the invariance concerning the input video length is due to the analytic solution of LLC formulation.

Table 4.5 presents the average time (in milliseconds) to process each frame (resolution 720p) of the input video regarding the steps to accelerate the video for the proposed method, graph-based semantic Hyperlapse (Section 4.2.1), and Microsoft Hyperlapse (MSH) [Joshi et al., 2015]. The table is organized in Frame Description, Frame Sampling, and Video Stabilization columns.

The proposed method executes the optical flow inference, object detection using YOLO, and extracts features related to the color histograms of the image to compute the Frame Description; following, the proposed method estimate the speed-up rates to each video segment and solve the LLC formulation to execute the Frame Sampling; and finally, it performs the Video Stabilization step. The most time-consuming task during the Frame Description step is to calculate the color histogram features followed by the optical flow inference. Frame Sampling is the step sped-up in this work. Video Stabilization leads as the most time-consuming step by a large margin, spending between 12 seconds in the best case and 24 seconds in the worst case per frame. It is noteworthy that the Video Stabilization time process is highly dependent on the frame sampling and the input video itself.

The graph-based semantic Hyperlapse executes the optical flow inference, calculates the Earth Mover’s Distance (EMD) of the color histograms, and models the

**Table 4.5.** Average time processing per frame to perform the complete fast-forward pipeline of our proposed method, graph-based semantic fast-forward, and Microsoft Hyperlapse (MSH). The values are in milliseconds. The \* indicates value reported by the authors.

Method	Frame Description			Frame Sampling		Video Stabilization
	Optical Flow	YOLO	Features Extraction	Speed-up Estimation	Sampling	
Ours	177.74	33.33*	231.49	0.21	0.16	12,000 – 25,000
	Optical Flow	EMD Histograms	Edge Weights	Speed-up Estimation	Sampling	
Graph-based	177.74	61.57	19.83	0.21	31.72	14,000 – 24,000
MSH		0.02*			0.005*	

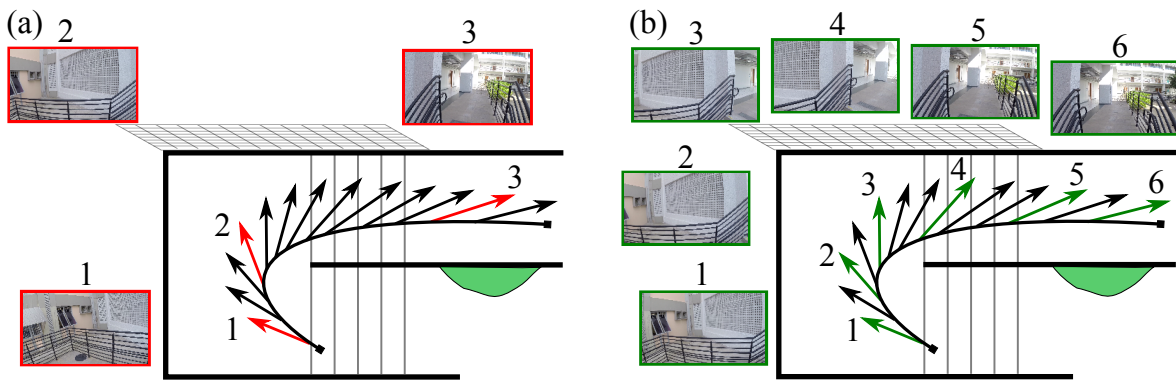
graph calculating the edge weights to compute the Frame Description; following, the graph-based method estimate the speed-up rates to each video segment, optimize the parameters settings, and find the shortest path in the created graph to execute the Frame Sampling; and finally, it executes the Video Stabilization step. The most time-consuming task during the Frame Description step is to infer the optical flow.

The authors of the Microsoft Hyperlapse work reported the values 0.02 milliseconds to create the frame description and 0.005 milliseconds to execute the frame sampling and stabilize the video. The values were measured running on a single core of a 2.67 GHz Intel Xeon X5650 PC from 2011 running Windows 8.1 (Reported information).

Experiments were conducted in a computer with an i7-6700K CPU @ 4.00GHz and 16 GB of memory. The values reported to Ours and Graph-based have no code optimization, while the MSH was highly optimized by a team of professional software engineerings to run in real time on standard personal computer and even in smart phones architectures.

## 4.6 Ablation Study

Along the ablation analysis, we discuss effect of each methodological step in the sparse-based final result, *i.e.*, application of steps Weighted Sparse Frame Sampling, the Smoothing Frame Transitions, and Fill Gap between segments during the frame sampling process. We also compare the LLC formulation to other general sparse coding formulations, and the usage of Convolutional Neural Networks (CNN) deep-features



**Figure 4.11.** The effect of applying the Weighted Sparse Sampling in an abrupt camera movement segment of a real experiment. Black arrows are the frames of the original video, red arrows are the frames selected by non-weighted sparse sampling, and the green arrows represent the frames sampled by the weighted sparse sampling. Each image is related with the respective numerated arrow.

instead of the hand-crafted features proposed in this dissertation. Finally, we evaluate the benefits of applying the Video Stabilization to produce the final accelerated video.

### 4.6.1 Weighted Sparse Sampling

As stated in Section 3.2.1, we introduce a new model based on weighted sparse sampling to address the problem of abrupt camera motions. In this model, small weights are applied to frames into temporal regions of abrupt camera motions increasing the probability of these frames being selected and, consequently, to create a smooth sequence.

To test the efficiency of the proposed solution, we evaluate the number of frames sampled by the weighted and non-weighted frame sampling in regions stated as having abrupt camera motions by the CDC-based classifier (Section 3.2.1) on the Semantic Dataset. The weighted version manages to sample, in average, three times more frames than the non-weighted version.

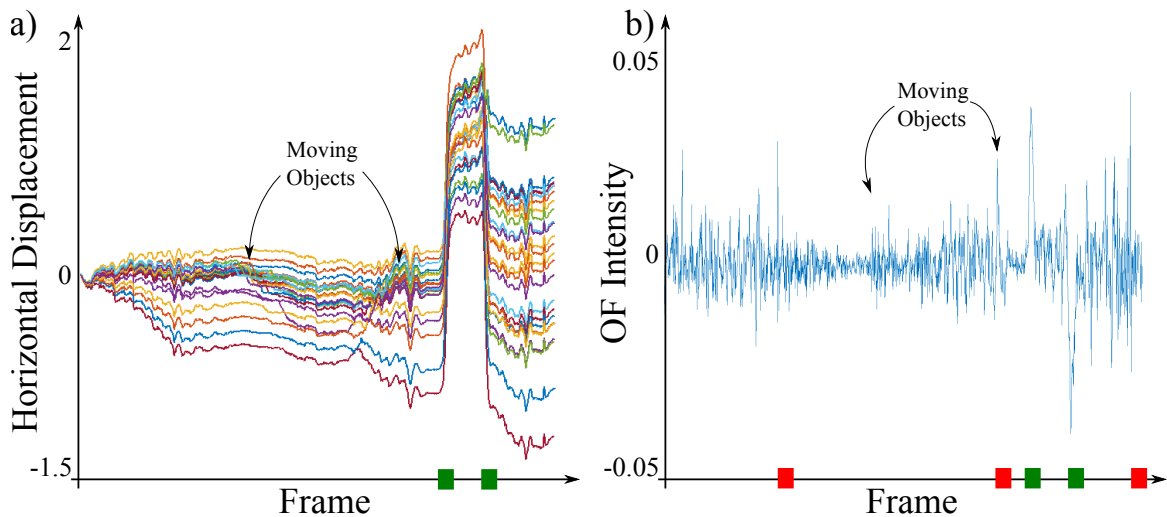
Figure 4.11 illustrates the effect of solving the sparse sampling by weighting the activation vector in a real experiment. Weighting strategy create smoother frame movement by using a denser sampling in curves (Figure 4.11-b) than when applying the non-weighted sparse sampling version (Figure 4.11-a). In this particular segment, our approach sampled twice the number of frames, leading to less shaky in lateral motions.

## 4.6.2 Detection of abrupt camera movement

As important as using a weighted formulation is the estimation of the weight values. The calculation of weight matrix  $W$  used in the weighted sparse sampling formulation (Equation 3.10) is based on the detection of video segments with abrupt camera movements. To detect these segments, we estimate the optical flow in a  $5 \times 5$  grid window between all consecutive frames by applying the sparse optical flow proposed in the work of Poley et al. [2014]. The information of the horizontal displacements is used to create the Cumulative Displacement Curves (CDC) [Poley et al., 2014].

The advantage of using CDC over Optical Flow (OF) to detect abrupt camera movements is regarding the CDC robustness against dynamic objects in the scene. By using OF, camera motion can be miss-estimated in case of a displacement of large objects, *e.g.*, a car or a person close to the recorder.

Figure 4.12 illustrates the OF and the CDC of a video recorded in a controlled environment. For this experiment, a person crossed the path of the camera to show the effect of the movement of scene components. Figure 4.12-b shows the average optical flow magnitude for each frame of the video, and the abrupt camera movement are inferred through thresholding. The moving object results in high average magnitude, leading the detector to miss-interpretation in some cases (red boxes in X-axis). However, when analyzing the CDC (Figure 4.12), the moving object caused interference in some of the



**Figure 4.12.** Comparison of abrupt camera detection using Optical Flow versus Cumulative Displacement Curves CDC (a) and Optical Flow OF (b) of a video recorded in a controlled environment. Green and Red boxes in the  $x$ -axis indicate true and false abrupt camera motion detection, respectively. The false detection was actually small variations in the scene and a person walking through the scene.

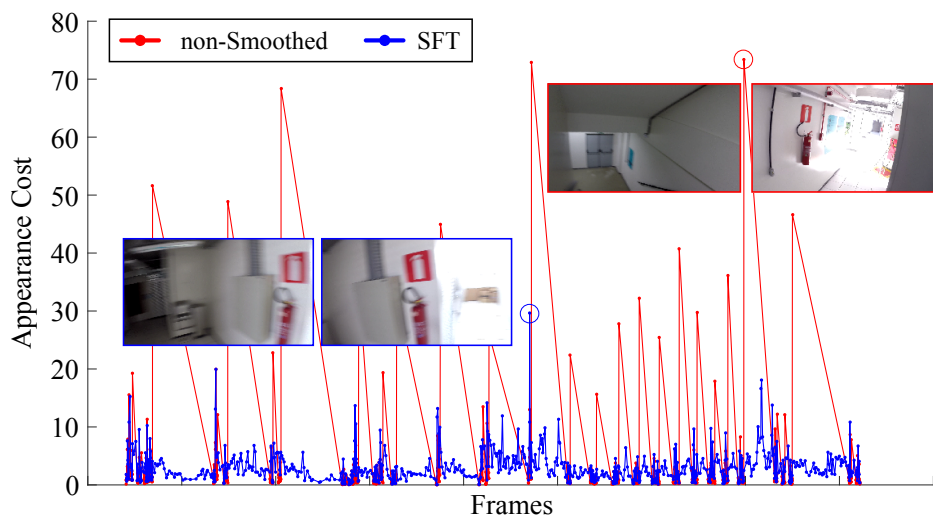
displacement curves only, not affecting the overall result (green boxes in X-axis).

### 4.6.3 Smooth Frame Transition

To analyze the effects of the Smooth Frame Transition step, we first execute the sparse-based frame sampling only, and then we execute the sparse-based frame sampling followed by the Smooth Frame Transition. The step Fill Gap between segments is not performed to make clear the comparison.

By computing the coefficient of variation (CV), we measured the relative variability of the points representing the appearance cost of the frames (blue and red points in Figure 4.13). The appearance cost is computed as the Earth Mover’s Distance [Pele and Werman, 2009] between the color histogram of frames in a transition.

After applying the proposed smoothing approach, we achieved  $CV = 0.97$ , while the simple sampling provided  $CV = 2.39$ . The smaller value for our method indicates a smaller dispersion and consequently fewer visual discontinuities inside the segments. Figure 4.13 shows the result when using the Smooth Frame Transition (SFT) approach and the non-smooth approach during the frame sampling step. The horizontal axis contains the index of the selected frames and the vertical axis represents the appearance cost between the  $i$ -th frame and the following frame in the final video. Points in the red line represent the oversampling pattern of non-smoothed sparse sampling, in which



**Figure 4.13.** Frame sampling and appearance cost of the transitions in the final video before and after applying the Smoothing Frame Transitions (SFT) to the video “Walking 25p”. Images with blue border show the frames composing the transition with the highest appearance cost using SFT. Images with red borders are related to the non-smoothed sparse sampling.



many frames are sampled in segments hard to reconstruct followed by a big jump.

The abrupt scene changing is depicted by high values of appearance cost. Red-bordered frames in Figure 4.13 show an example of two images that compose the transition with the highest appearance cost for a fast-forwarded version of the video “Walking 25p” using non-smooth approach. After applying the SFT step, we see a more spread sampling covering all segments, and with less video discontinuities. Blue-bordered images present the frames composing the transition with the highest appearance cost using the sparse sampling with the SFT step. By comparing the red and blue curves, one can clearly see that after using SFT, we achieve smoother transitions, *i.e.*, lower values for the appearance cost.

#### 4.6.4 Filling Gap Between Segments

The effect of the Fill Gap Between Segments step was evaluated by producing videos either using the proposed methodology (sparse-based frame sampling, Smooth Frame Transition, and Fill Gap Between Segments) and the methodology without the Fill Gap Between Segments step.

Table 4.6 shows the results of the evaluation performed using the sequences in the Semantic Dataset concerning the amount of semantic retained, visual instability, temporal discontinuity, and deviation between the required and achieved speed-up rates. The Column “Discontinuity” in Table 4.6 presents the temporal gap problem

**Table 4.6.** Evaluation of the frame sampling modeling applying only the Smoothing Frame Transition (STF) step and applying the complete framework with Fill Gap step.

Videos	Semantic(%) <sup>1</sup>		Instability <sup>2</sup>		Discontinuity <sup>2</sup>		Speed-up Deviation <sup>2</sup>	
	SFT	Ours	SFT	Ours	SFT	Ours	SFT	Ours
B.0p	<b>22.4</b>	<b>22.4</b>	23.4	<b>20.7</b>	<b>9.3</b>	<b>9.3</b>	<b>0.3</b>	0.3
B.25p	20.9	<b>23.6</b>	48.9	<b>43.0</b>	20.7	<b>10.1</b>	<b>0.6</b>	0.9
B.50p	26.4	<b>27.9</b>	29.0	<b>27.7</b>	34.3	<b>9.2</b>	<b>0.2</b>	0.5
B.50p 2	19.2	<b>21.2</b>	25.8	<b>24.2</b>	38.9	<b>10.7</b>	<b>0.2</b>	1.1
D.0p	28.1	<b>29.3</b>	43.9	<b>40.5</b>	36.5	<b>11.4</b>	<b>0.3</b>	0.8
D.25p	25.2	<b>25.7</b>	34.2	<b>33.1</b>	15.8	<b>9.6</b>	2.7	<b>2.0</b>
D.50p	19.5	<b>22.2</b>	35.7	<b>35.0</b>	29.3	<b>10.0</b>	<b>1.3</b>	2.6
W.0p	<b>7.4</b>	<b>7.4</b>	36.9	<b>32.3</b>	<b>7.1</b>	<b>7.1</b>	<b>0.0</b>	<b>0.0</b>
W.25p	37.1	<b>38.5</b>	33.3	<b>30.8</b>	12.3	<b>9.7</b>	1.0	<b>0.6</b>
W.50p	23.4	<b>26.7</b>	34.7	<b>32.5</b>	21.3	<b>13.7</b>	<b>0.0</b>	0.5
W.75p	48.9	<b>52.7</b>	33.0	<b>30.1</b>	27.3	<b>10.3</b>	<b>0.1</b>	0.6
<i>Mean</i>	<i>25.3</i>	<b>27.1</b>	<i>35.9</i>	<b>31.8</b>	<i>23.0</i>	<b>10.1</b>	<b>0.6</b>	<i>0.9</i>

<sup>1</sup>Higher is better

<sup>2</sup>Lower is better

related to the frame selection of the sparse-based frame sampling, and by comparing the results, we can observe the effect of applying the Fill Gap correction between segments presented in Section 3.2.1.2. After applying the proposed step, the RMSE between inter-frames jumps and the required speed-up  $F_d$  dropped from 23.0 (STF) to 10.1 (Ours).

Fill Gap Between Segments steps also leads to the creation of more visual stable videos, as showed in column “Instability”. However, the speed-up deviation increases from 0.3 without using Fill Gap step to 0.9 when using this step. This behavior is expected, using Fill Gap step implies in run frame sampling more often, increasing the cumulative error.

The complete methodology proposed in this dissertation outperformed the frame sampling without applying the Fill Gap Between Segments step regarding semantic, instability and mainly in discontinuity matters. Finally, the semantic retained in the video produced by the complete methodology is greater than the amount related to the video produced without using the Fill Gap Between Segments.

#### 4.6.5 Comparison between Sparse Coding formulations

As presented in Section 3.2.1, the first step of our sparse-based approach to frame selection is to model the sampling process as a Minimum Sparse Reconstruction problem. The final video will be composed of the frames related to the dictionary basis selected by the solution of the Minimum Sparse Reconstruction problem, which lead to better video story reconstruction. The design decision in this step is the choice of which formulation to solve the Minimum Sparse Reconstruction problem. We present the experimental evaluation comparing the performance of tree formulations: Locality-constrained Linear Coding (LLC), Orthogonal Matching Pursuit (OMP) and Lasso (SC).

In the proposed methodology, the Minimum Sparse Reconstruction problem is solved using the formulation presented in Section 3.2.1. The same problem can be solved by the Lasso formulation based on the weighted sparse coding using  $L_1$  distance as follows:

$$\boldsymbol{\alpha}^* = \arg \min_{\boldsymbol{\alpha} \in \mathbb{R}^n} \frac{1}{2} \|\mathbf{v} - \mathbf{D} \boldsymbol{\alpha}\|_2^2 + \lambda_\alpha \|\mathbf{W} \boldsymbol{\alpha}\|_1, \quad (4.8)$$

where  $\lambda_\alpha$  is a regularization term of the sparsity of  $\boldsymbol{\alpha}$ . The definitions of  $D$ ,  $\mathbf{v}$ ,  $W$ , and  $\boldsymbol{\alpha}$  are the same as presented in Section 3.2.1. We solved Equation 4.8 using the Lasso package implementation [Efron et al., 2004], and we adjusted the  $\lambda_\alpha$  value according to Algorithm 1.

The Minimum Sparse Reconstruction problem can be also solved by the Orthogonal Matching Pursuit (OMP) based on the sparse coding using  $L_0$  distance as follows:

$$\boldsymbol{\alpha}^* = \arg \min_{\boldsymbol{\alpha} \in \mathbb{R}^n} \frac{1}{2} \|\mathbf{v} - D \boldsymbol{\alpha}\|_2^2 + \lambda_\alpha \|\boldsymbol{\alpha}\|_0. \quad (4.9)$$

This equation is solved using the Orthogonal Matching Pursuit implementation, and  $\lambda_\alpha$  value is calculated according to Algorithm 1. Due to the usage of  $L_0$  distance to calculate the sparsity term, weighting can not be applied to dictionary basis.

As stated by Wang et al. [2010], the locality present in the LLC formulation provides better results than sparse solutions, since locality leads to sparsity without reciprocity. To verify this statement in the frame sampling problem, we run LLC, OMP, and SC approaches over all videos of the Semantic Dataset. To make the analysis of effects among the sparse coding formulations feasible, we did not apply the steps Fill Gap Between Segments and Video Stabilization. Table 4.7 presents the results concerning the amount of retained semantic, visual instability, temporal discontinuity, absolute speed-up deviation, and running times of the sampling step.

The overall result of Table 4.7 shows the LLC approach achieves the best performance in creating smoother videos and with a significant amount of semantic information (second better). OMP outperforms LLC in all videos in the discontinuity of the frame selection. However, the running times for OMP are approximately  $25\times$  slower than LLC in average for the Semantic dataset. This is due to the analytic solution

**Table 4.7.** Evaluation of the frame sampling modeling by Locality-constrained Linear Coding (LLC) and regular sparse coding methods Orthogonal Matching Pursuit (OMP) and Lasso (SC).

Videos	Semantic(%) <sup>1</sup>			Time(s) <sup>2</sup>			Instability <sup>2</sup>			Discontinuity <sup>2</sup>			Speed-up Deviation <sup>2</sup>		
	LLC	SC	OMP	LLC	SC	OMP	LLC	SC	OMP	LLC	SC	OMP	LLC	SC	OMP
B.0p	<b>24.6</b>	21.5	22.6	<b>3.1</b>	63.8	67.9	<b>23.4</b>	24.2	23.9	9.8	7.7	<b>5.8</b>	0.3	0.2	0.2
B.25p	20.4	19.4	<b>22.9</b>	<b>1.2</b>	16.9	25.3	48.9	49.3	<b>46.4</b>	21.8	20.3	<b>5.3</b>	<b>0.6</b>	<b>0.6</b>	<b>0.6</b>
B.50p	26.3	28.9	<b>29.9</b>	<b>1.6</b>	19.9	33.5	<b>29.0</b>	31.8	31.7	34.3	14.2	<b>5.6</b>	<b>0.2</b>	<b>0.2</b>	<b>0.2</b>
B.50p 2	18.1	18.2	<b>23.4</b>	<b>1.0</b>	8.9	24.6	<b>25.8</b>	26.4	27.4	38.8	35.1	<b>5.1</b>	0.8	<b>0.2</b>	<b>0.2</b>
D.0p	30.0	28.1	<b>31.6</b>	<b>0.7</b>	13.9	30.7	43.9	45.4	<b>41.5</b>	10.8	15.7	<b>6.7</b>	<b>0.3</b>	<b>0.3</b>	<b>0.3</b>
D.25p	24.7	25.8	<b>26.2</b>	<b>0.5</b>	8.2	14.1	<b>34.2</b>	35.3	35.2	22.0	15.0	<b>6.3</b>	<b>2.1</b>	<b>2.1</b>	<b>2.1</b>
D.50p	19.0	19.3	<b>21.6</b>	<b>0.6</b>	5.8	10.1	<b>35.7</b>	37.0	38.5	27.6	39.1	<b>6.4</b>	<b>1.3</b>	1.5	<b>1.3</b>
W.0p	7.5	7.9	<b>11.3</b>	<b>0.9</b>	26.1	26.1	36.8	38.0	<b>32.6</b>	8.4	15.6	<b>4.7</b>	<b>0.0</b>	<b>0.0</b>	<b>0.0</b>
W.25p	<b>36.7</b>	35.9	31.4	<b>0.9</b>	30.8	13.1	<b>33.3</b>	35.5	33.8	12.3	11.5	<b>5.7</b>	1.0	0.6	<b>0.0</b>
W.50p	<b>25.2</b>	24.8	<b>25.2</b>	<b>0.8</b>	24.1	27.6	<b>34.7</b>	36.2	36.2	19.2	15.7	<b>6.0</b>	<b>0.0</b>	0.1	0.1
W.75p	<b>49.4</b>	44.6	49.1	<b>1.0</b>	16.0	29.6	<b>33.0</b>	37.0	36.3	35.7	23.4	<b>7.0</b>	0.3	<b>0.1</b>	<b>0.1</b>
Mean	<i>25.6</i>	<i>24.9</i>	<b>26.8</b>	<b>1.1</b>	<i>21.3</i>	<i>27.5</i>	<b>34.4</b>	<i>36.0</i>	<i>34.8</i>	<i>21.9</i>	<i>19.4</i>	<b>5.9</b>	<i>0.6</i>	<b>0.5</b>	<b>0.5</b>

<sup>1</sup>Higher is better

<sup>2</sup>Lower is better

provided by LLC. Regarding the speed-up evaluation, all competitors achieved closer values with respect to speed-up deviation.

Bearing an analytic solution is also a major advantage of LLC over both SC and OMP as it leads to better performance. The column “Time” in Table 4.7 shows the running times for each frame sampling method. When using LLC, the frame sampling becomes approximately  $20\times$  faster than the best competitor (SC).

#### 4.6.6 Feature Scalability

One of the problems related to current frame sampling methodologies in literature is the non-scalability regarding the number of features used to characterize frames and their transitions. This limitation is related to the direct mapping from the number of features to the dimensionality of the search space where the solution of the optimization problem relies on. One of the advantages of solving the Minimum Sparse Reconstruction problem using the LLC formulation is the analytic solution instead of the optimization solution.

To demonstrate the capability of our methodology in handling high dimensional features, we perform the frame sampling step using CNN deep-features instead of hand-crafted features ( $d_i \in \mathbb{R}^{446}$ ) proposed in Section 3.2.1. Frames descriptors were extracted from the Architecture AlexNet cropped after the layer `fc-7`, resulting in a 4,069d-feature vector for each frame ( $d_i \in \mathbb{R}^{4,069}$ ).

**Table 4.8.** Evaluation of the frame sampling describing the video frames through handcrafted features proposed in Section 3.2.1 against using Deep features (AlexNet layer `fc7`).

Videos	Semantic(%) <sup>1</sup>		Instability <sup>2</sup>		Discontinuity <sup>2</sup>		Speed-up Deviation <sup>2</sup>	
	Hand-crafted	Deep	Hand-crafted	Deep	Hand-crafted	Deep	Hand-crafted	Deep
B.0p	<b>22.4</b>	18.9	<b>20.7</b>	26.2	9.3	<b>6.4</b>	0.3	<b>0.2</b>
B.25p	23.6	<b>25.1</b>	<b>43.0</b>	44.6	<b>10.1</b>	16.3	<b>0.9</b>	1.6
B.50p	27.9	<b>32.6</b>	<b>27.7</b>	30.2	<b>9.2</b>	13.9	<b>0.5</b>	1.3
B.50p 2	21.2	<b>24.6</b>	<b>24.2</b>	26.3	<b>10.7</b>	13.1	<b>1.1</b>	2.6
D.0p	29.3	<b>30.6</b>	40.5	<b>35.3</b>	<b>11.4</b>	17.8	<b>0.8</b>	2.0
D.25p	25.7	<b>34.6</b>	33.1	<b>29.6</b>	<b>9.6</b>	12.9	<b>2.0</b>	3.3
D.50p	22.2	<b>28.6</b>	35.0	<b>33.7</b>	<b>10.0</b>	12.1	<b>2.6</b>	3.2
W.0p	7.4	<b>12.9</b>	<b>32.3</b>	36.3	7.1	<b>6.1</b>	<b>0.0</b>	<b>0.0</b>
W.25p	38.5	<b>39.4</b>	30.8	<b>26.3</b>	<b>9.7</b>	13.9	<b>0.6</b>	1.1
W.50p	<b>26.7</b>	<b>26.7</b>	32.5	<b>30.7</b>	<b>13.7</b>	16.5	<b>0.5</b>	1.4
W.75p	52.7	<b>57.7</b>	<b>30.1</b>	<b>30.1</b>	<b>10.3</b>	14.9	<b>0.6</b>	1.7
Mean	<i>27.1</i>	<b>30.2</b>	<i>31.8</i>	<b>31.7</b>	<i>10.1</i>	<i>13.1</i>	<b>0.6</b>	<i>1.7</i>

<sup>1</sup>Higher is better

<sup>2</sup>Lower is better

Table 4.8 presents the results for the Semantic dataset. Sparse sampling performed using deep-features outperformed the sampling from hand-crafted features in Instability and Semantic metrics. However, the discontinuity values for experiments using hand-crafted features outperformed the values for the frame sampling using deep-features.

### 4.6.7 Video Stabilization

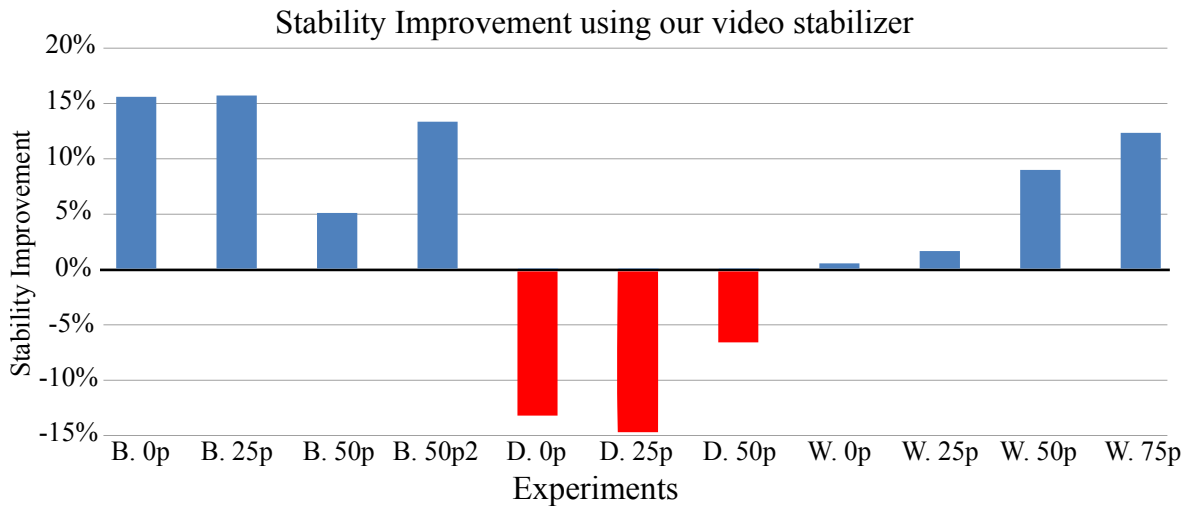
To evaluate the effect of the Video Stabilization process, we compared the visual instability index of accelerated videos from Semantic Dataset before and after applying this step. Figure 4.14 depicts the achieved results heading our discussion, in which 100% improvement indicates that the output video became as stable as the original one.

Regarding Walking and Biking experiments, the application of the stabilization step led to an improvement meaning the creation of a more stable final video. Driving experiments are failure cases of the application of the stabilization process. Our assumption is that the reason for the badly performance is two fold. First, videos recorded in a moving car have mostly high speed of motion causing low scenes overlaps between the accelerated video frames. Thus, due to the features mismatches, the target homography planes are erroneously computed, leading the video to present unstable transitions. The second reason is related to the diversity of motions in patches of the frame. Most of the frames are forward-facing views composed of the car dash and windshield. The frame region depicting the car dash presents slow motion while the region related to the windshield has high speed motion. If the homography transformation was calculated mostly by keypoints sampled in the windshield region, this transformation will deform the car dash generating shakiness. Otherwise, if the keypoints sampled to calculate the homography transformation are from the car dash, the transformation will be insufficient to stabilize the image region related to the windshield.

The average stability improvement over all experiments in the Semantic Dataset by applying the Stabilization process was 3.5%. Analyzing the effect of this stabilization excluding the failures cases, Driving experiments, the value improvement is 9.1%.

A more detailed performance assessment of stabilizing accelerated egocentric videos was performed by comparing our stabilization method with the work of Joshi et al. [2015] (MSH), which is a smoothed homography frame-to-frame transformation. We create a video using the frames selected by the MSH frame sampling step. Then, we execute our stabilization step on this video. To evaluate the smoothness, we compare the values of the instability index of this stabilized video with the MSH video.

The average instability values over all experiments in the Semantic Dataset was



**Figure 4.14.** Visual stability improvement by applying the Video Stabilization process in the accelerated videos. An improvement of 100% indicates that the output video is as stable as the original one. The W., D., and B. stand for Walking, Driving, and Biking experiments, respectively.

equal to 35.0, facing 34.0 achieved by MSH stabilizer. However, our methodology has not been designed to perform well with larger movements, like driving. Then, considering the Driving sequences as outliers samples and not including them in the analysis, our stabilizer achieved an average instability index of 32.3 against 32.5 of the MSH stabilizer. Further, in the work of Joshi et al. [2015], the authors stated that their frame selection is optimal. Therefore, our video stabilization outperforms their stabilizer step in the best set of frames.

## 4.7 Concluding Remarks

To the sight of clarification, we summarize the results achieved and presented in this research, and associate them to the scientific hypothesis presented in Section 1.3.

The first dissertation statements is that a Multi-Importance approach is achieved by addressing the speed-up rates inversely proportional to the semantic level of the segment. Results presented in Sections 4.5.1 demonstrated that the graph-based methodology retained almost  $3\times$  more semantic information when using the proposed Multi-Importance approach when compared to the graph-based technique which treats the semantic definition as a binary problem (FFSE method). The amount of semantic information retained in the final video when using the sparse-based frame sampling was as high as the graph-based in both analyzed datasets (Sections 4.5.1 and 4.5.4).

The second dissertation statements is: A machine learning based method trained

from users' data tie the definition of semantics to the users' preferences. We proposed the CoolNet to confirm this hypothesis, and the results in Section 4.5.2 showed that a machine learning method is capable of infer the users' preference directly from video frames and the video statistics. The main advantage is to define semantics without user interaction or object predefinition.

Regarding the smoothing of abrupt changes of speed-up rates, the third dissertation statements suggest to create an intermediate segment and used the mean values between the preceding and following segments speed-up value. Experiments performed in Section 4.5.3 demonstrated that after applying the proposed speed-up smoothing approach, the RMSE between the speed-up applied to consecutive video segments dropped from 5.7 to 3.6 in the Semantic dataset, and from 8.4 to 4.5 in DoMSEV.

The last dissertation statements is regarding to model the frame sampling as a Minimum Sparse Reconstruction problem and solving through a sparse-based technique turns the frame sampling process scalable in the number of features and creates visual stable videos. Results performed in Sections 4.5.4 comparing to semantic hyperlapse methods demonstrate the videos produced by the proposed sparse-based technique are more visually stable. Also, the amount of retained semantic is greater than the competitors (Section 4.5.1). Finally, the scalability is confirmed in Section 4.6.6 by executing the frame sampling describing the frame and transitions using the deep-feature ( $d_i \in \mathbb{R}^{4096}$ ) extracted using the AlexNet CNN.

Concluding this Section, we detail how methodology step composed the published works during the development of this dissertation. The graph-based frame sampling combined with the Multi-Importance approach and an in-depth analysis of the Video Stabilization method were published at the Journal of Visual Communication and Image Representation (JVCI) 2018 [Silva et al., 2018a]<sup>3</sup>. Parts of the sparse-based approach, such as sparse-based frame sampling and Smoothing Frame Transition step, and the Dataset of Multimodal Semantic Egocentric Videos (DoMSEV) were published at the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2018 [Silva et al., 2018b]<sup>4</sup>.

---

<sup>3</sup>Project webpage: <https://www.verlab.dcc.ufmg.br/semantic-hyperlapse/jvci2018/>

<sup>4</sup>Project webpage: <https://www.verlab.dcc.ufmg.br/semantic-hyperlapse/cvpr2018/>





# Chapter 5

## Conclusions

In this dissertation, we tackled the challenging task of creating Semantic Hyperlapse for first-person videos through a parameter-free sparse coding-based framework composed of the adaptive frame sampling, Smooth Frame Transition, and Fill Gap steps. The frame sampler was modeled as a Minimum Sparse Reconstruction problem using a weighted formulation allowing a denser sampling along the segments with high camera movement. Smoothing Frame Transitions step identifies peaks of visual instability and inserts new frames to address abrupt camera movements by using a denser sampling along the segments with high movement. Finally, the Fill Gap processing deals with both visual discontinuities introduced by the sampler and abrupt changes in speed-up rates between video segments. Contrasting with previous fast-forward techniques that are not scalable in the number of features used to describe the frame/transition, our method is not limited by the size of feature vectors.

The experiments showed that our method was superior to state-of-the-art semantic fast-forward methods in terms of semantic, speed-up, stability, and processing time. We also performed an ablation analysis that showed the improvements provided by the weighted modeling, smoothing transition step, and Fill Gap between segments. An additional contribution of this dissertation are the three proposed datasets. The first is a 80-hour multimodal dataset with several annotations related to the recorder preferences, activity, interaction, attention, and the scene where the video was taken. The second is a dataset with controlled amount of semantic information per video to perform semantic evaluation. The third is a dataset of frames from YouTube egocentric videos split in two classes: videos with high number of likes, and videos recorded in boring and monotonous places.

Regarding the dilemma of semantic definition, we proposed a methodology – be it general or customized – learns semantics from the user preferences. To validate the

proposed methodology, we ran quantitative experiments, based on fixed and specific semantics, and qualitative experiments, using the Recorder-Aware semantic approach proposed here.

## 5.1 Limitations

The major drawback of the proposed methodology is to model the frame sampling problem regardless of temporal information about frames transitions, only appearance and movement characteristic of frames were encoded. Due to this limitation, it was necessary to employ two post-processing step during the frame sampling, *i.e.*, Smoothing Frame Transitions and Fill Gap Between Segments.

The video stabilization technique does not perform satisfactorily when the camera motion is high or the scene present regions with different movement behaviors, like recording from the inside of a moving car. This is due to the premise that a global and planar homography transformation can be used to stabilize the scene. However, homography transformation are only acceptable when the surface is planar.

We can also point out as a limitation of this dissertation the bounded definition of semantic focused only on the user who will watch the video. This definition does not consider the recorder interest. Here we present one example of this limitation: considering a video recorded by a person who passes through a beach along his path to work, besides the fact of the beach is a natural view, it could not be interesting for him once he sees this scenario every day.

Regarding the emphasis effect to attract the user attention to the relevant information of the video, we only explored the difference of speed-ups. By using this approach, the user is capable of identifying the relevant parts of the video, but it could not understand why these parts are important. One example of this situation, the user sees the playback speed-up decreasing in a video segment containing people and trees, (s)he will not understand why the acceleration rate decreases in this part if (s)he is not aware of the semantics used in the Semantic Hyperlapse process.

## 5.2 Future works

The research area that this dissertation belongs is recent, there are many opportunities to expand the borders. Following we present some directions to continue improving the creation of Semantic Hyperlapse for First-Person Videos focusing, but not limited to, addressing the limitations presented in Section 5.1.

Regarding the temporal representation limitation, the dictionary used in the sparse coding based frame sampling could be formulated encoding information about the inter frames relations, such as, frame alignment, movement pattern and velocity, and visual continuity. By formulating the dictionary in this manner, also the video story representation and the insight about the activation vector should be revised.

To improve the stabilization process, we could represent an unsteady sequence of selected frames by virtually created frames composing visually stable transitions. These virtual frames could be created by, for example: reconstructing the geometry of the scene where the video was recorded and repositioning virtual cameras; using non-rigid transformations to increase the alignment between consecutive frames; or even training a Convolutional Neural Network to learn how to create a visually pleasant sequence of frames given a unsteady input video segment.

Given the amount of available information, the semantic definition problem could be approached fusing multimodal data. Internet activity can be used to infer the user's profile, defining what is relevant for specific users. Furthermore, analyze the recorder behavior while recording the video using extra sensors such as depth information, inertial measurements, heart beating, and temperature sensors<sup>1</sup>, could provide information to infer what attracts was important to the record. In this last example, the system could remember the record what as relevant in that specific video. One application example, a foreign person recording a video visiting a specific monument in the new city where he/she just moved in. After a few years, when this person passes through the same monument, he/she could act differently, because it is now part of his/her routine. However, the person will be able to remember the feeling of seeing the monument in the first time when watching the Semantic Hyperlapse of his/her first visit.

To address the emphasis effect limitation mentioned early, methods such zooming-in where is the relevant information on the image and the changing of speed-up could be combined to emphasize which are the relevant parts of the video and why they are important. An additional emphasis approach is to play the relevant segments in slow-motion. For this, frame rendering approaches could be applied.

Additional contributions can be reached by using the labeled dataset proposed in this dissertation. For each frame of the videos, we have annotation about the scene where the frame was taken. A study can be conducted using the scene information to related the place where the frame was taken with the semantics. For example, using noise patterns as relevant information, if a person is in a concert, the noise pattern is different from the case in which the person is inside a library.

---

<sup>1</sup>Most of these sensor are attached on moderns action cameras and smart watches.

Also, the semantic definition and the policy applied in the frame sampling could be modified by the activity performed by the recorder using annotation of the proposed dataset. One example of this situation is the difference between a person jogging and standing in a bus stop. When someone is jogging, the visual stability of the camera is a major problem and should have a proper attention during the frame sampling, and at the same time, it is more difficult to attract the attention of the jogger. Controversially, to sample frames in the video segment in which the person is standing in the bus stop is not a challenging task, and a lot of situations and objects could attract the recorder attention since he/she is idle while waiting for the bus.

Finally, we suggest the use of the annotation of recorder interest as a potential study to understand how the recorder behavior during the video capture is related to his/her personal interest. This discussion can be extended to the topic of how the information about the personal interest of the recorder can be used to infer semantics in an unknown situation.

# Bibliography

- Abu-El-Haija, S., Kothari, N., Lee, J., Natsev, P., Toderici, G., Varadarajan, B., and Vijayanarasimhan, S. (2016). Youtube-8m: A large-scale video classification benchmark. *CoRR*, abs/1609.08675.
- Bai, C. and Reibman, A. R. (2016). Characterizing distortions in first-person videos. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, pages 2440–2444.
- Bettadapura, V., Castro, D., and Essa, I. (2016). Discovering picturesque highlights from egocentric vacation videos. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–9, Lake Placid, USA.
- Cong, Y., Yuan, J., and Luo, J. (2012). Towards scalable summarization of consumer videos via sparse dictionary selection. *IEEE Transactions on Multimedia*, 14(1):66–75. ISSN 1520-9210.
- del Molino, A. G., Tan, C., Lim, J. H., and Tan, A. H. (2016). Summarization of Egocentric Videos: A Comprehensive Survey. *IEEE Transactions on Human-Machine Systems*, 47(1):65–76. ISSN 21682291.
- Dollár, P. (2016). Piotr’s Computer Vision Matlab Toolbox (PMT). <https://github.com/pdollar/toolbox>.
- Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (2004). Least angle regression. *Annals of Statistics*, 32:407–499.
- Fathi, A., Hodgins, J. K., and Rehg, J. M. (2012a). Social interactions: A first-person perspective. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1226–1233. ISSN 1063-6919.
- Fathi, A., Li, Y., and Rehg, J. M. (2012b). Learning to Recognize Daily Actions Using Gaze. In Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., and Schmid,

- C., editors, *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 314–327, Berlin, Heidelberg. Springer Berlin Heidelberg.
- Fu, T.-J., Tai, S.-H., and Chen, H.-T. (2019). Attentive and adversarial learning for video summarization. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*, Hawaii, USA. to appear.
- Gemmell, J., Bell, G., Lueder, R., Drucker, S., and Wong, C. (2002). Mylifebits: Fulfilling the memex vision. In *10th ACM International Conference on Multimedia*, pages 235–238, New York, NY, USA. ACM.
- Gong, Y. and Liu, X. (2000). Video summarization using singular value decomposition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 174–180. ISSN 1063-6919.
- Gygli, M., Grabner, H., and Gool, L. V. (2015). Video summarization by learning submodular mixtures of objectives. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3090–3098, Boston, USA. ISSN 1063-6919.
- Gygli, M., Grabner, H., Riemenschneider, H., and Van Gool, L. (2014). Creating Summaries from User Videos. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 505–520.
- Halperin, T., Poley, Y., Arora, C., and Peleg, S. (2017). Egosampling: Wide view hyperlapse from egocentric videos. *IEEE Transactions on Circuits and Systems for Video Technology*, PP(99):1–1. ISSN 1051-8215.
- Higuchi, K., Yonetani, R., and Sato, Y. (2017). Egoscanning: Quickly scanning first-person videos with egocentric elastic timelines. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, CHI '17*, pages 6536–6546, New York, NY, USA. ACM.
- Hsu, Y.-F., Chou, C.-C., and Shih, M.-Y. (2012). Moving camera video stabilization using homography consistency. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, Orlando, FL, USA. (IEEE).
- Joshi, N., Kienzle, W., Toelle, M., Uyttendaele, M., and Cohen, M. F. (2015). Real-time hyperlapse creation via optimal frame selection. *ACM Trans. Graph.*, 34(4):63:1–63:9. ISSN 0730-0301.

- Karpenko, A. (2014). The technology behind hyperlapse from instagram. <http://instagram-engineering.tumblr.com/post/95922900787/hyperlapse>. Accessed: 2016-05-12.
- Kennedy, J. and Eberhart, R. (1995). Particle swarm optimization. In *IEEE International Conference on Neural Networks*, volume 4, pages 1942–1948 vol.4.
- Kopf, J., Cohen, M. F., and Szeliski, R. (2014). First-person hyper-lapse videos. *ACM Trans. Graph.*, 33(4):78:1--78:10. ISSN 0730-0301.
- Lai, W.-S., Huang, Y., Joshi, N., Buehler, C., Yang, M.-H., and Kang, S. B. (2017). Semantic-driven Generation of Hyperlapse from 360° Video. *ArXiv e-prints*.
- Lal, S., Duggal, S., and Sreedevi, I. (2019). Online video summarization: Predicting future to better summarize present. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*, Hawaii, USA. to appear.
- Lan, S., Panda, R., Zhu, Q., and Roy-Chowdhury, A. K. (2018). Ffnet: Video fast-forwarding via reinforcement learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6771–6780. ISSN 2575-7075.
- Lee, Y. J., Ghosh, J., and Grauman, K. (2012). Discovering important people and objects for egocentric video summarization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1346–1353. ISSN 1063-6919.
- Liao, S., Jain, A. K., and Li, S. Z. (2016). A fast and accurate unconstrained face detector. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 38(2):211–223. ISSN 0162-8828.
- Lin, Y. L., Morariu, V. I., and Hsu, W. (2015). Summarizing while recording: Context-based highlight detection for egocentric videos. In *Proceedings of the IEEE International Conference on Computer Vision Workshop (ICCVW)*, pages 443–451.
- Liu, G., Liu, Y., Guo, M., Liu, P., and Wang, C. (2015). Non-negative locality-constrained linear coding for image classification. *Intelligence Science and Big Data Engineering. Image and Video Data Engineering. Lecture Notes in Computer Science.*, 9242:462--471. ISSN 16113349.

- Lu, Z. and Grauman, K. (2013). Story-driven summarization for egocentric video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2714–2721. ISSN 1063-6919.
- Mahasseni, B., Lam, M., and Todorovic, S. (2017). Unsupervised video summarization with adversarial LSTM networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 202--211, Honolulu, USA.
- Mann, S. (1998). 'wearcam' (the wearable camera): personal imaging systems for long-term use in wearable tetherless computer-mediated reality and personal photo/videographic memory prosthesis. In *Digest of Papers. Second International Symposium on Wearable Computers (Cat. No.98EX215)*, pages 124–131.
- Marvaniya, S., Damoder, M., Gopalakrishnan, V., Iyer, K. N., and Soni, K. (2016). Real-time video summarization on mobile. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, pages 176–180.
- Mei, S., Guan, G., Wang, Z., He, M., Hua, X. S., and Feng, D. D. (2014). L2,0 constrained sparse dictionary selection for video summarization. In *2014 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. ISSN 1945-7871.
- Mei, S., Guan, G., Wang, Z., Wan, S., He, M., and Feng, D. D. (2015a). Video summarization via minimum sparse reconstruction. *Pattern Recognition*, 48(2):522 – 533. ISSN 0031-3203.
- Mei, S., Wang, Z., He, M., and Feng, D. (2015b). Resource restricted on-line video summarization with minimum sparse reconstruction. In *Picture Coding Symposium (PCS)*, pages 139–143.
- Ngo, C.-W., Ma, Y.-F., and Zhang, H. (2003). Automatic video summarization by graph modeling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 104–109 vol.1.
- Ogawa, M., Yamasaki, T., and Aizawa, K. (2017). Hyperlapse generation of omnidirectional videos by adaptive sampling based on 3d camera positions. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, pages 2124–2128. ISSN 2381-8549.
- Okamoto, M. and Yanai, K. (2014). *Summarization of Egocentric Moving Videos for Generating Walking Route Guidance*, pages 431–442. Springer Berlin Heidelberg, Berlin, Heidelberg.



- Oliveira, G., Nascimento, E., Vieira, A., and Campos, M. (2014). Sparse spatial coding: A novel approach to visual recognition. *IEEE Transactions on Image Processing (TIP)*, 23(6):2719–2731. ISSN 1057-7149.
- Otani, M., Nakashima, Y., Rahtu, E., Heikkilä, J., and Yokoya, N. (2017). Video summarization using deep semantic features. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, pages 361–377, Cham. Springer International Publishing.
- Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66. ISSN 0018-9472.
- Panda, R. and Roy-Chowdhury, A. K. (2017). Collaborative summarization of topic-related videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4274–4283, Honolulu, USA.
- Pele, O. and Werman, M. (2009). Fast and robust earth mover’s distances. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 460–467. ISSN 1550-5499.
- Plummer, B. A., Brown, M., and Lazebnik, S. (2017). Enhancing video summarization via vision-language embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1052–1060, Honolulu, USA.
- Poleg, Y., Arora, C., and Peleg, S. (2014). Temporal segmentation of egocentric videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2537–2544.
- Poleg, Y., Halperin, T., Arora, C., and Peleg, S. (2015). Egosampling: Fast-forward and stereo for egocentric videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4768–4776.
- Potapov, D., Douze, M., Harchaoui, Z., and Schmid, C. (2014). Category-specific video summarization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 540–555.
- Ramos, W. L. S. (2017). Semantic Hyperlapse for Egocentric Videos. Master’s thesis, Universidade Federal de Minas Gerais (UFMG), Belo Horizonte, Brazil.
- Ramos, W. L. S., Silva, M. M., Campos, M. F. M., and Nascimento, E. R. (2016). Fast-forward video based on semantic extraction. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, pages 3334–3338, Phoenix, AZ, USA.

- Rani, P., Jangid, A., Namboodiri, V. P., and Venkatesh, K. S. (2018). Visual odometry based omni-directional hyperlapse. In Rameshan, R., Arora, C., and Dutta Roy, S., editors, *National Conference on Computer Vision, Pattern Recognition, Image Processing, and Graphics*, pages 3--13, Singapore. Springer Singapore.
- Redmon, J. and Farhadi, A. (2016). YOLO9000: Better, Faster, Stronger. *ArXiv e-prints*.
- Romberg, J. (2008). Imaging via compressive sampling. *IEEE Signal Processing Magazine*, 25(2):14–20. ISSN 1053-5888.
- Sharghi, A., Laurel, J. S., and Gong, B. (2017). 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)*, pages 2127–2136, Honolulu, USA.
- Silva, M. M., Ramos, W. L. S., Chamone, F. C., Ferreira, J. P. K., Campos, M. F. M., and Nascimento, E. R. (2018a). Making a long story short: A multi-importance fast-forwarding egocentric videos with the emphasis on relevant objects. *Journal of Visual Communication and Image Representation (JVCI)*, 53:55 – 64. ISSN 1047-3203.
- Silva, M. M., Ramos, W. L. S., Ferreira, J. P. K., Campos, M. F. M., and Nascimento, E. R. (2016). Towards semantic fast-forward and stabilized egocentric videos. In *Proceedings of the European Conference on Computer Vision Workshop (ECCVW)*, pages 557--571, Amsterdam, NL. Springer International Publishing.
- Silva, M. M., Ramos, W. L. S., Ferreira, J. P. K., Chamone, F. C., Campos, M. F. M., and Nascimento, E. R. (2018b). A weighted sparse sampling and smoothing frame transition approach for semantic fast-forward first-person videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2383–2392, Salt Lake City, USA.
- Song, X., Chen, K., Lei, J., Sun, L., Wang, Z., Xie, L., and Song, M. (2016). Category driven deep recurrent neural network for video summarization. In *IEEE International Conference on Multimedia Expo Workshops*, pages 1–6.
- Su, Y.-C., Jayaraman, D., and Grauman, K. (2016). Pano2vid: Automatic cinematography for watching 360° videos. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*.

- Traffic-Inquiries (2018). Cisco visual networking index: Forecast and methodology, 2017-2022. Technical report 1543280537836565, CISCO.
- Wang, J., Yang, J., Yu, K., Lv, F., Huang, T., and Gong, Y. (2010). Locality-constrained linear coding for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3360–3367, San Francisco, USA. ISSN 1063-6919.
- Wang, M., Liang, J., Zhang, S., Lu, S., Shamir, A., and Hu, S. (2018). Hyper-lapse from multiple spatially-overlapping videos. *IEEE Transactions on Image Processing (TIP)*, 27(4):1735–1747. ISSN 1057-7149.
- Wright, J., Yang, A. Y., Ganesh, A., Sastry, S. S., and Ma, Y. (2009). Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 31(2):210–227. ISSN 0162-8828.
- Xiong, B., Kim, G., and Sigal, L. (2015). Storyline representation of egocentric videos with an applications to story-based search. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 4525–4533.
- Xu, J., Mukherjee, L., Li, Y., Warner, J., Rehg, J. M., and Singh, V. (2015). Gaze-enabled Egocentric Video Summarization via Constrained Submodular Maximization Supplementary Material. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 58, page 6964.
- Yang, J. A., Lee, C. H., Yang, S. W., Somayazulu, V. S., Chen, Y. K., and Chien, S. Y. (2016). Wearable social camera: Egocentric video summarization for social interaction. In *IEEE International Conference on Multimedia Expo Workshops*, pages 1–6.
- Yao, T., Mei, T., and Rui, Y. (2016). Highlight detection with pairwise deep ranking for first-person video summarization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Zhang, K., Chao, W.-L., Sha, F., and Grauman, K. (2016). *Video Summarization with Long Short-Term Memory*, chapter Proceedings of the European Conference on Computer Vision (ECCV), pages 766–782. Springer International Publishing, Amsterdam, NL.
- Zhao, B., Fei-Fei, L., and Xing, E. P. (2011). Online detection of unusual events in videos via dynamic sparse coding. In *Proceedings of the IEEE Conference on Com-*

*puter Vision and Pattern Recognition (CVPR)*, pages 3313--3320, Colorado Springs, USA.

Zhao, B. and Xing, E. P. (2014). Quasi real-time summarization for consumer videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2513--2520, Columbus, USA. ISSN 1063-6919.

Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., and Oliva, A. (2014). Learning deep features for scene recognition using places database. In *Advances in Neural Information Processing Systems 27*, pages 487--495. Curran Associates, Inc.