Análise Visual do Comportamento de Objetos em Vídeos de Vigilância

Cibele Mara Fonseca

Universidade Federal de Uberlândia
Faculdade de Computação
Programa de Pós-Graduação em Ciência da Computação

Uberlândia
2021

Cibele Mara Fonseca

Análise Visual do Comportamento de Objetos em Vídeos de Vigilância

Dissertação de mestrado apresentada ao Programa de Pós-graduação da Faculdade de Computação da Universidade Federal de Uberlândia como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação.

Área de concentração: Ciência da Computação

Orientador: José Gustavo de Souza Paiva

Uberlândia
2021

Ficha Catalográfica Online do Sistema de Bibliotecas da UFU
com dados informados pelo(a) próprio(a) autor(a).

F676
2021

Fonseca, Cibele Mara, 1996-
Análise Visual do Comportamento de Objetos em Vídeos de Vigilância [recurso eletrônico] / Cibele Mara Fonseca. - 2021.

Orientador: José Gustavo de Souza Paiva.
Dissertação (Mestrado) - Universidade Federal de Uberlândia, Pós-graduação em Ciência da Computação.
Modo de acesso: Internet.
Disponível em: http://doi.org/10.14393/ufu.di.2021.532
Inclui bibliografia.
Inclui ilustrações.

1. Computação. I. Paiva, José Gustavo de Souza,1979-, (Orient.). II. Universidade Federal de Uberlândia. Pós-graduação em Ciência da Computação. III. Título.

CDU: 681.3

Bibliotecários responsáveis pela estrutura de acordo com o AACR2:

Gizele Cristine Nunes do Couto - CRB6/2091

UNIVERSIDADE FEDERAL DE UBERLÂNDIA
Coordenação do Programa de Pós-Graduação em Ciência da Computação
Av. João Naves de Ávila, nº 2121, Bloco 1A, Sala 243 - Bairro Santa Mônica, Uberlândia-MG, CEP 38400-902
Telefone: (34) 3239-4470 - www.ppgco.facom.ufu.br - cpgfacom@ufu.br

ATA DE DEFESA - PÓS-GRADUAÇÃO

Programa de Pós-Graduação em:	Ciência da Computação				
Defesa de:	Mestrado Acadêmico, 21/2021, PPGCO				
Data:	26 de agosto de 2021	Hora de início:	14:03	Hora de encerramento:	16:10
Matrícula do Discente:	11822CCP003				
Nome do Discente	Cibele Mara Fonseca				
Título do Trabalho:	Visual Analysis of Objects Behavior in Surveillance Videos				
Área de concentração:	Ciência da Computação				
Linha de pesquisa:	Ciência de Dados				
Projeto de Pesquisa de vinculação:	CNPq MCTIC/CNPq 28/2018 - 431860/2018-1				

Reuniu-se, por videoconferência, a Banca Examinadora, designada pelo Colegiado do Programa de Pós-graduação em Ciência da Computação, assim composta: Professores Doutores: Humberto Luiz Razente - FACOM/UFU, Nivan Roberto Ferreira Júnior - CIN/UFPE e José Gustavo de Souza Paiva - FACOM/UFU, orientador da candidata.

Os examinadores participaram desde as seguintes localidades: Nivan Roberto Ferreira Júnior - Recife/PE ; Humberto Luiz Razente e José Gustavo de Souza Paiva - Uberlândia/MG. A discente participou da cidade de Uberlândia/MG.

Iniciando os trabalhos o presidente da mesa, Prof. Dr. José Gustavo de Souza Paiva, apresentou a Comissão Examinadora e a candidata, agradeceu a presença do público, e concedeu à Discente a palavra para a exposição do seu trabalho. A duração da apresentação da Discente e o tempo de arguição e resposta foram conforme as normas do Programa.

A seguir o senhor presidente concedeu a palavra, pela ordem sucessivamente, aos examinadores, que passaram a arguir a candidata. Ultimada a arguição, que se desenvolveu dentro dos termos regimentais, a Banca, em sessão secreta, atribuiu o resultado final, considerando a candidata:

Aprovada.

Esta defesa faz parte dos requisitos necessários à obtenção do título de Mestre.

O competente diploma será expedido após cumprimento dos demais requisitos, conforme as normas do Programa, a legislação pertinente e a regulamentação interna da UFU.

Nada mais havendo a tratar foram encerrados os trabalhos. Foi lavrada a presente ata que após lida e achada conforme foi assinada pela Banca Examinadora.

Documento assinado eletronicamente por **José Gustavo de Souza Paiva**, **Professor(a) do Magistério Superior**, em 01/09/2021, às 14:42, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

Documento assinado eletronicamente por **Nivan Roberto Ferreira Junior**, **Usuário Externo**, em 01/09/2021, às 15:19, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

Documento assinado eletronicamente por **Humberto Luiz Razente**, **Professor(a) do Magistério Superior**, em 01/09/2021, às 15:36, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

A autenticidade deste documento pode ser conferida no site https://www.sei.ufu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **2942878** e o código CRC **3CBD0D2B**.

Referência: Processo nº 23117.050417/2021-44 SEI nº 2942878

Acknowledgements

I am grateful to my parents and my husband for all love and support. I appreciate the support of my advisor during this work.

This work has received financial support from the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001, and from the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) - 431860/2018-1.

Visual Analysis of Objects Behavior in Surveillance Videos

Cibele Mara Fonseca

Universidade Federal de Uberlândia
Faculdade de Computação
Programa de Pós-Graduação em Ciência da Computação

Uberlândia
2021

Abstract

The use of surveillance camera system based on CCTV (closed-circuit television) is present in different sectors of society, whether to prevent thefts, depredations, vandalism, invasions, violence, terrorism, among other threats, generating a large volume of video footage. The manual analysis of these videos is an unfeasible task due to the large volume of long duration videos, as well as due to intrinsic human limitations, which compromise the perception of multiple strategic events. Several surveillance video analysis tasks employ information from moving objects detection/tracking in order to analyse their behavior in the scene, and thus to understand their role on events that occurs in the video. However, most of the existing surveillance applications focus only in detecting/tracking these objects, or provide basic behavior analysis, with few or no user insertion in the process. In this sense, Information Visualization techniques is a potential tool to represent/explore objects behavior/relationship in these videos. These representations allow the user to effectively identify objects behavior patterns and comprehend how they contribute to the occurrence of strategic events in the videos.

This work proposes a visual analysis strategy of surveillance video with focus on the identification and exploration of objects behavior and their relationship with events occurrence. The proposed strategy combines a set of coordinated interactive layouts to represent multiple aspects of the objects behavior, such as their presence in scene, relationships/interaction among them, movement and scene occupation. Users may change the visualization perspective, focusing on specific objects, time periods and scene regions, providing spatial and temporal perspectives of these behavior. The conducted experiments in several surveillance scenarios demonstrate the ability of the proposed methodology in identifying different aspects of objects behavior and how these behavior relate to events occurrence in the surveillance videos, enabling the user to make effective decisions in these videos.

Keywords: Objects behavior, Visualization, Visual analytics, Surveillance video.

Resumo

O uso de sistema de câmeras de vigilância baseado em CFTV (circuito fechado de televisão) está presente em diversos setores da sociedade, seja para prevenir furtos, depredações, vandalismo, invasões, violência, terrorismo, entre outras ameaças, gerando um grande volume de filmagens. A análise manual desses vídeos é uma tarefa inviável devido ao grande volume de vídeos de longa duração, além das limitações humanas intrínsecas, que comprometem a percepção de múltiplos eventos estratégicos. Diversas tarefas de análise de vídeos de vigilância empregam informações de detecção/rastreamento de objetos em movimento a fim de analisar seu comportamento na cena e, assim, compreender seu papel em eventos ocorridos no vídeo. No entanto, a maioria dos sistemas de vigilância existentes se concentra apenas na detecção/rastreamento desses objetos, ou fornece uma análise básica de comportamento, com pouca ou nenhuma inserção do usuário no processo. Nesse sentido, as técnicas de Visualização de Informação são uma ferramenta em potencial para representar/explorar o comportamento/relacionamento dos objetos nesses vídeos. Essas representações permitem que o usuário efetivamente identifique os padrões de comportamento dos objetos e compreenda como eles contribuem para a ocorrência de eventos estratégicos nos vídeos.

Este trabalho propõe uma estratégia de análise visual de vídeos de vigilância com foco na identificação e exploração do comportamento de objetos e sua relação com a ocorrência de eventos. A estratégia proposta combina um conjunto de *layouts* interativos coordenados para representar múltiplos aspectos do comportamento dos objetos, como presença em cena, relacionamentos/interações entre eles, movimento e ocupação da cena. Os usuários podem mudar a perspectiva de visualização, focando em objetos específicos, períodos de tempo e regiões da cena, fornecendo perspectivas espaciais e temporais desses comportamentos. Experimentos conduzidos em diversos cenários de vigilância demonstram a capacidade da metodologia proposta em identificar diferentes aspectos do comportamento dos objetos e como esses comportamentos se relacionam com a ocorrência de eventos nos vídeos de vigilância, permitindo ao usuário tomar decisões eficazes nesses vídeos.

Palavras-chave: Comportamento de objetos, Visualização, Análise visual, Vídeo de vigilância.

List of Figures

List of Tables

Contents

I hereby certify that I have obtained all legal permissions from the owner(s) of each third-party copyrighted matter included in my thesis, and that their permissions allow availability such as being deposited in public digital libraries.

Cibele Mara Fonseca Adviser: Prof. Dr. José Gustavo de Souza Paiva

Chapter 1

Introduction

The worldwide increasing in the security concerning by governments, companies, institutions and also by the citizens in general led to the popularization of Closed-circuit television (CCTV) systems, specially CCTV cameras (KRUEGLE, 2011). CCTV cameras produce large volumes of recording footage that hold a rich informational content, whose analysis is useful for identify events of interest, and thus for the comprehension of the phenomena captured in scene. The comprehension of these phenomena may help solving crimes and thefts, understanding people movement, identifying suspicious behavior, among other tasks. However, the manual analysis of these videos is an unfeasible task because of the large volume of videos, as well as their long duration. In addition, due to human limitations, the potential occurrence of multiple strategic events can not be perceived by the agents, hampering the analysis process.

Several surveillance video analysis tasks employ information produced by the detection and tracking of moving objects in scene (JOSHI; THAKORE, 2012). This task includes identifying the objects location at different instants in the video, in order to understand their behavior on the scene and analyze aspects such as speed and direction of movement. These objects are the main elements directly involved in the events, and thus the analysis of their behavior provides the comprehension of what happens in the video. Several strategic aspects related to the objects behavior can be considered, including their time presence (when a object is in the scene, when it is not), trajectories (positions occupied by the object in specific time instants), movement (the object is static or in movement in the scene) and relationships among them.

Smart Surveillance employs automatic video analysis technologies in video surveillance applications (HAMPAPUR et al., 2003). In this context, several computational techniques have been used to analyze the content of surveillance videos in a variety of scenarios. In most of the existing surveillance applications, the objects are only detected and tracked (ELHOSENY, 2020; HU; NI, 2017), which may limit the comprehension of their behavior. Some applications focus on automatic behavior analysis, but implement "black boxes" procedures, with few or no user insertion into the process (LEE; SHIN, 2019; SREENU; DURAI, 2019; MABROUK; ZAGROUBA, 2018)

User participation in the data analysis process is essential to combine flexibility, creativity and background knowledge with the storage capacity and computational power of the machines. Information Visualization techniques apply interactive visual representations in order to amplify the acquisition and use of knowledge from abstract data. They are used to create graphical representations for a dataset, in order to increase the user cognition about the information present in data, helping the user to better understand the involved phenomena.

Information Visualization techniques represent an important tool for Smart Surveillance. There are several systems that use visual strategies to analyze different aspects of the videos (LIANG; NIU, 2002). Some systems visually represent trajectories (MEGHDADI; IRANI, 2013), while others focus on summarizing and identifying events (MENDES; PAIVA; SCHWARTZ, 2019), among other tasks. To the extent of our knowledge, no system focus directly on the objects on the scene and the relationship between them. The identification and exploration of these aspects can assist understanding events present in video. Information Visualization techniques are a tool with great potential for representation/exploration of objects present in videos and their relationships, offering the possibility to better and more quickly identify objects behavior patterns existing in the surveillance materials. Users then actively participate in the analysis process and are able to make effective decisions on the events in these videos.

1.1 Objectives

This work proposes a visual analysis strategy of surveillance video with focus on the identification and exploration of objects behavior and their relationship with events occurrence. The proposed layouts seek to effectively represent several aspects of the objects behavior, such as their presence, relationships/interaction among them, movement and scene occupation. The proposed approach combine a set of interactive layouts: *Appearance Bars View*, *Brush View* and *Frame View*. The main one, *Appearance Bars View*, depicts the dynamics and distribution of each object presence in scene, revealing detailed information such as the moments in which each object is in the scene, when the objects interact with other objects, when each of them occupy a certain region in the scene, as well as the objects average speed variation. All the layouts are coordinated with each other, and interactions in one captured aspect are reflected on all other aspects. Thus, the objects behavior can be quickly identified and explored by observing the aspects highlighted by the layouts.

We believe that such a strategy supports the identification of objects behavior enabling the discovery of how these behavior influence on the relevant events occurred in surveillance videos. The contributions of this work are listed as follows:

- A surveillance video visual analysis methodology with focus on summarizing/exploring object behavior in scene, alone and/or interacting with other objects, and how these behavior relates to the occurrence of events in the video;

- A computational system that implements our proposed methodology;

- The methodology validation through a series of case studies considering various surveillance scenarios, including different movement patterns and types of events.

1.2 Thesis Organization

The next sections follow the structure bellow:

- Fundamentals: presents the definition of basic concepts related to this work and the state of art approaches. Smart surveillance process is introduced, including approaches for object detection and some smart surveillance tasks. Concepts related to Information Visualization are also introduced and approaches related to visualization of surveillance videos are presented and discussed;

- System Description: describes the proposed system, enumerating the requirements, explaining the developed visual and related design decisions and presenting the system interface and functionalities;

- Experimental Results: presents the experimental procedure to evaluate our proposal, including the description of the videos and experimental process. The results are discussed, as well as a the limitations of our proposal;

- Conclusion: presents the findings achieved after the execution of this master project, detailing our contributions to the visual analysis of objects behavior in surveillance videos, in addition to limitations and future works.

Chapter 2

Fundamentals

Analyzing surveillance videos is a laborious task due to several factors, including their long duration, as well as intrinsic human limitations, and focus the analysis only in interesting information, excluding irrelevant events, represents a big challenge for video surveillance area. Computational strategies, in particular the use of interactive layouts may help to communicate relevant information about objects behavior during their presence in scene, facilitating the decision making by the security agent. This chapter presents basic concepts employed in works related to this project, in addition to a literature review of surveillance tasks and visual analysis techniques of surveillance videos which motivated the proposal of this work.

2.1 Basic Concepts

This section presents basic definitions that are employed in the related work and in our proposal:

- **Frame:** Image captured by a surveillance camera which describes an instant of time from a video;

- **Scene:** Spatial location captured by the surveillance camera in which events of interest occur;

- **Object:** Element captured in the video that may be interesting for a specific analysis. It may be a person, an animal, or inanimate objects such as cars, backpacks, among others;

- **Foreground:** Invisible layer in all the scene which contains the objects under analysis;

- **Background:** Invisible layer in the scene containing all scene region, except the foreground;

- **Video:** Electronic reproduction technique covering a sequence of frames, which corresponds to a certain period of time in which eventually one or more events of interest occurred;
- **Surveillance Video:** Video aiming to monitor and track people activities or places with the aim of maintaining inspection, social control and security, recognizing and monitoring threats, preventing and investigating criminal activities, among others;
- **Object Identification:** Process of listing objects that compose a scene in a video, assigning an individual label;
- **Object Tracking:** Object detections in sequential frames for objects behavior analysis. Object tracking aims to monitor the spatial position of the object over time, identifying its position in all frames of the video;
- **Event:** Activity or phenomenon which occurs during a specific period of time from the video;
- **Layout:** Interactive visual representation of one or multiple aspects of the surveillance video, for the analysis of specific phenomena;
- **Interaction:** Actions performed by an object toward another objects, considering these objects participating in a meeting;
- **Meeting:** Instant of time in which two objects interact with each other;
- **Speed:** Measurement of object movement in scene, considering both distance traveled and a certain period of time;
- **Object Behavior:** The set of actions executed by an object in the video, during its presence in scene. It can be related to where and when the object is in the scene, its interactions/relationships and direction/speed of its movement.

2.2 Smart Surveillance

Video Surveillance, commonly referred as closed-circuit TV, involves monitoring people and objects of interest using video cameras (JYOTHI; BABU; BACHU, 2019). Video surveillance systems are composed of a cameras network controlled or monitored by a human operator, aiming at investigate behavior, activities and other information in a video sequence (TAHA et al., 2015; VERMA; KUMAR; TOMAR, 2015). Video surveillance systems are used for development of smart cities, enabling tasks as traffic management and public places monitoring (LI et al., 2021; DAO et al., 2018). Some real-time monitoring systems use edge computing to improve large-scale information acquisition and performance of data processing (WANG et al., 2019; WANG; PAN; ESPOSITO, 2017).

Video surveillance can be performed manually, semi-automatically or fully automatically using video surveillance systems (KIM et al., 2010). The manual analysis of surveillance videos is unfeasible because of their long duration and large volume, in addition to the potential occurrence of multiple events that may not be perceived by an human operator. Several video analysis techniques are used to extract knowledge from these videos, including computer vision techniques (VEZZANI; CUCCHIARA, 2010).

These automatic video analysis techniques are employed in **Smart Surveillance** systems, aiming to identify important aspects in surveillance videos (HAMPAPUR et al., 2003), supporting several surveillance tasks. The processing framework of a smart surveillance system can include the stages of motion/object detection, object tracking and behavior/activity analysis (KO, 2011). A general framework of an automatic smart surveillance system is shown in Figure 1. The video acquired by cameras is processed in several sequential tasks providing information to support decision making. First, the objects and motion are detected, and thus the detected objects are classified and tracked, allowing the execution of final high-level tasks: analyzing behavior and activity and identifying people.

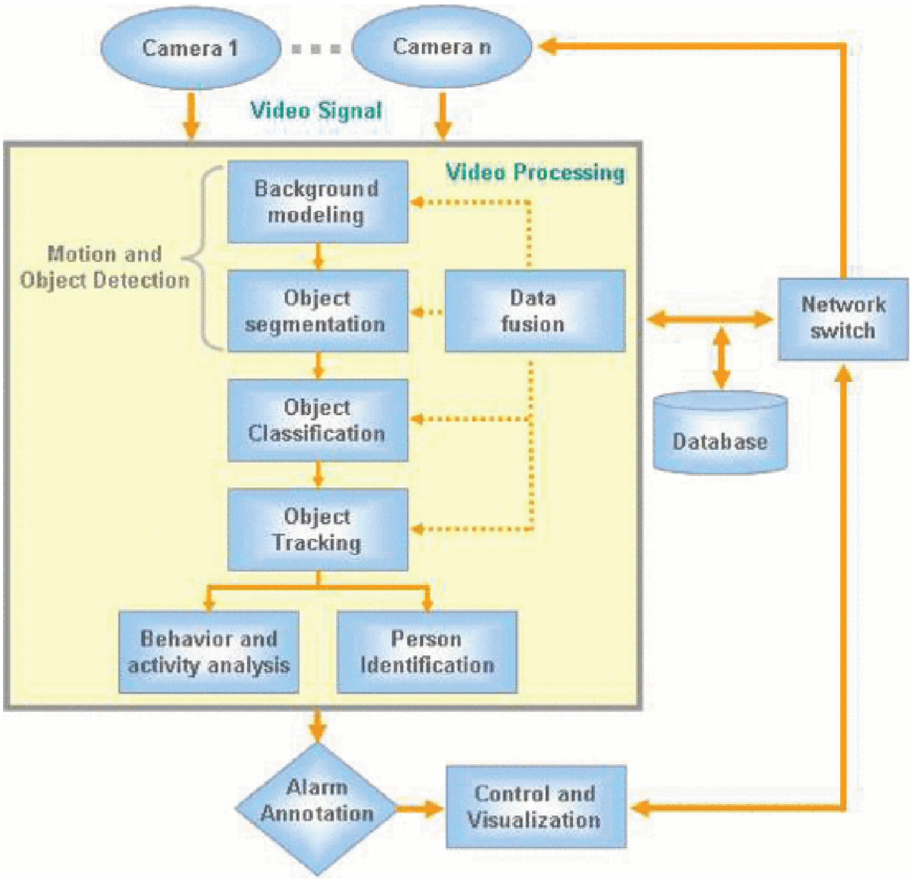


Figure 1 – General framework of an automatic smart surveillance system. (Figure extracted from (KO, 2011))

Surveillance video analysis may be performed with different objectives, and for each objective, video content analysis is performed under a certain aspect or multiple aspects, such as events occurrence, trajectories, among others. One of these aspects is objects present in scene. The analysis based on objects involves the detection of objects in scene in order to further analyze their dynamics. The object dynamics include occupied places, movement and interaction. Thus, the analysis of objects dynamics allows the comprehension of these objects behavior.

The detection and tracking of objects is performed using a variety of computer vision approaches (ELHOSENY, 2020; HU et al., 2015). YOLO[1] (REDMON; FARHADI, 2018) is a real-time object detection system which applies a neural network to the full image dividing it into grids. The network then predict bounding boxes and class probabilities for each grid, and bounding boxes having class probability above a threshold are considered as detected objects. Several works use YOLO to enhance the detection of specific classes, such as pedestrians (QU et al., 2018; MOLCHANOV et al., 2017), faces (LI et al., 2020) and vehicles (ZHU et al., 2021; KIM et al., 2018). As YOLO is capable of detecting and tracking multiple objects, it is used for multi-target tracking in surveillance (TIAN et al., 2020; ZHANG et al., 2019a), supporting the monitoring of multiple objects at the same time.

Faster R-CNN (REN et al., 2016) is an object detection algorithm which uses an image as input to a convolutional network providing a convolutional feature map and predicting region proposals, obtained by a separate Region Proposal Network (RPN), and then a ROI Pooling layer is used to extract a fixed-length feature vector from each region proposal, thus the objects within the region are classified and bounding boxes values are defined. Li et al. (2017) use Faster R-CNN for facial expression recognition, in order to avoid explicit feature extraction and the problem of low-level data operation. Faster R-CNN is employed in video surveillance for detecting commonly analyzed objects, as pedestrians (LUO et al., 2018) and vehicles (HUANG, 2018), and it is also used for detecting specific objects which represent security threat, such as guns and knives (FERNANDEZ-CARROBLES; DENIZ; MAROTO, 2019).

Single Shot Detector (SSD) (LIU et al., 2016) detects multiple objects in an image using a single shot, similar to YOLO and different from Faster R-CNN, which needs two shots, one for generating region proposals and another one for detecting the object of each proposal. The SSD algorithm extracts features maps and applies convolutional filters to detect objects. Applications based on SSD algorithm are used for several purposes, including railway surveillance (YUNDONG et al., 2020; LI et al., 2020), counting vehicles (CHEN et al., 2018) and recognizing people on moving (GONG; SHU, 2020).

The results of detection and tracking techniques support several smart surveillance tasks, including:

❏ Objects Speed Estimation: Several algorithms can be used to calculate the objects

[1] <https://pjreddie.com/darknet/yolo/>

speed in the video using the result of detection and tracking techniques, supporting recognition of motion patterns and analysis of object behavior focusing on objects movement. Rahim et al. (2010), for example, presents a vehicle speed detection algorithm using a frame differencing technique and a kinematic equation, which can be implemented in a surveillance system in order to monitor vehicles guaranteeing safety of parking lots, streets and roads;

- Objects Abandonment Identification: Abandonment of objects is an event of great interest in surveillance videos (LUNA et al., 2018). The moving objects in the scene are detected with the objective of identifying the stationary ones, which are classified as abandoned object if they remain still after having been previously moving; Lin et al. (2015) approach focus on detecting abandoned luggage in surveillance videos by extracting foreground objects to identify static foreground regions as left-luggage candidates and a framework is used to identify if this regions contain abandoned objects by analyzing luggage owners trajectories;

- People Monitoring: People are the main agents in surveillance videos and many smart surveillance systems use strategies focused on human detection and tracking to monitor people. Nikouei, Chen e Faughnan (2018) propose a smart surveillance as an edge service applying a combination of two algorithm for human detection and tracking. Xu, Lv e Meng (2010) use human detection and tracking for counting people entering or leaving a region of interest and analyzing trajectories in order to counting the number of moving people in the scene;

- Crowd Analysis: Crowded scenes require monitoring an excessive number of individuals and their activities, thus computer vision techniques are used to automate the analysis of crowd behavior and activity, motion pattern learning and anomaly detection (LI et al., 2014). Many approaches for crowd analysis are based on deep learning techniques for detecting and count people and identifying activities occurrence, as violence or panic situations (SREENU; DURAI, 2019). Wang e Loy (2017) propose scene-independent crowd analysis with deep learning in which people in crowd are detected/tracked to estimate crowd density/counting and perform crowd attribute recognition through a CNN model;

- Activity Recognition: The automatic analysis of human activities in order to understand humans behavior is generally referred to as activity recognition (TAHA et al., 2015). There are several types of human activities. Vishwakarma e Agrawal (2013) divide human activity into four levels, according to its complexity: gesture, action, interaction and group activity. Babiker et al. (2017) developed an intelligent human activity system recognition which first performs human body detecting and tracking using background subtraction and then analyze the images to extract the values of the bounding box, centroid and area in order to build a sheet of features database, the obtained features are used as input for a neural network and the classification set model defines the human activity as walking, laying, sitting, boxing or hand waving;

- ❏ Abnormal Behavior Recognition: Involves the automatic detection and recognition of unusual activities/events in surveillance scenes. A local abnormal behavior detection method is proposed by Zhu, Hu e Shi (2016), performing spatio-temporal blobs extraction and using a statistical method to detect the candidates for blobs containing abnormal behavior, which often have characteristics of higher motion velocity and disordered motion direction, and thus using a maximum optical flow energy and local nearest descriptor to identify blobs with really abnormal behavior;

- ❏ Trajectory Analysis: Trajectory is composed of moving object localization over time and it can be obtained by object tracking. The trajectories in video can be analysed for different purposes, such as human behavior detection, event detection, suspicious activity detection and video summarization (AHMED et al., 2018). Ahmed et al. (2017) employ trajectory analysis for automatic detecting and extracting regions of interest from surveillance scene by estimating total time spent, speed and direction of objects while crossing a region, in order to localize regions of interest which influence the motion characteristics of moving objects. Cheng e Hwang (2011) propose an application for event detection based on trajectories analysis, in which tracking algorithms results serve as input for an artificial neural network, classifying the trajectories in classes used for detection of events: irregular speed, accelerating and sudden slowing down.

These algorithms are effective, but often fail in communicating the results to the user. Generally, the results are shown in tables with large volume of dense and non-intuitive information. In addition, the way these results are shown does not allow the user to contextualize these solutions with the scenario under analysis, which difficult the comprehension of these results and, consequently, decision making. A resolution that displays information in a intuitive way and allows more user interaction during analysis process may contribute to better data comprehension and decision making.

2.3 Information Visualization

Card, Mackinlay e Shneiderman (1999) define Information Visualization as the . Figure 2 shows the complete visualization process proposed by them. First, **Raw Data**, containing the original collected information, is structured and transformed into **Data Tables**. This step is necessary in situations in which the Raw Data does not contain a standard representation of the information or its content is not organized in a trivial way.

Data Tables are then mapped into **Visual Structures** using metaphors related to the tasks to be performed on these data, to the data context, or even to conventions/customs which aid in the user comprehension. Then, these Visual Structures are organized according to one or several analysis aspects for composing the **Views**, which allow users to perform analysis tasks by interaction/exploration of its content, in order to understand data meaning. Information Visualization improve the computer-human interaction through visual and interactive computational strategies, facilitating data understanding and simplifying decision-making process.

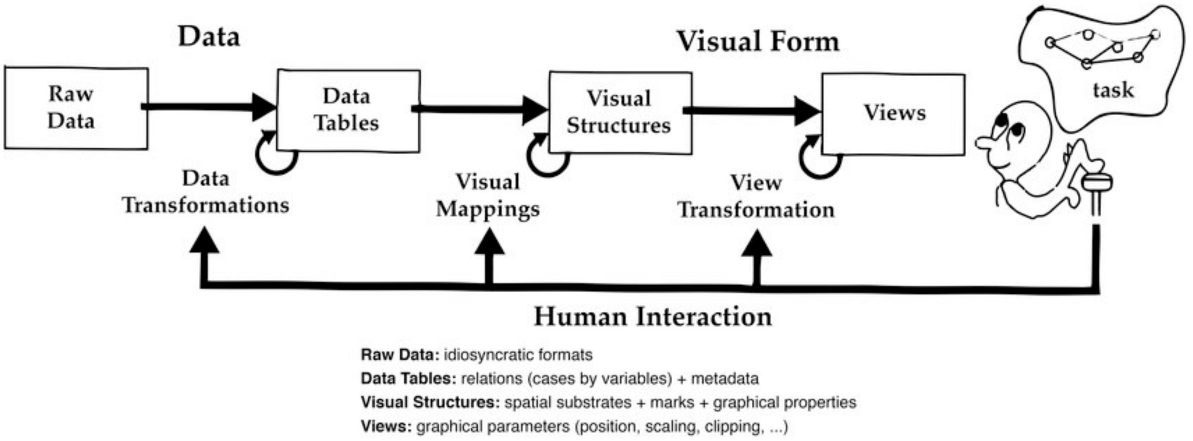

Data
Visual Form
task
Raw Data
Data Tables
Visual Structures
Views
Data Transformations
Visual Mappings
View Transformation
Human Interaction
Raw Data: idiosyncratic formats
Data Tables: relations (cases by variables) + metadata
Visual Structures: spatial substrates + marks + graphical properties
Views: graphical parameters (position, scaling, clipping, ...)

Figure 2 – Visualization process (CARD; MACKINLAY; SHNEIDERMAN, 1999).

There are several types of Information Visualization techniques in literature, and they can be categorized into two types, according to the layout construction procedure and to the displayed information. In **attribute-based** techniques, multiple data attributes are mapped in a two-dimensional space using visual cues such as colors, areas, shapes/icons or any other visual appealing symbol to highlight patterns involving these attributes, such as the correlation among them, recurring patterns, abnormal events, etc. Parallel Coordinates (INSELBERG; DIMSDALE, 1990), Scatterplot Matrices (CLEVELAND, 1993), Pixel Bars Charts (KEIM et al., 2002), Space Filling Curves (VELHO; GOMES, 1991) , RadViz (HOFFMAN; GRINSTEIN; PINKNEY, 1999) and Treemaps (JOHNSON, 1992) are some examples of attribute-based techniques provided in literature. We employ in our proposal strategies that can be categorized in this type of technique. **Point-placement** techniques organize each data instance of a n-dimensional space as a point in two-dimensional or three-dimensional space, employing a dimensionality reduction procedure, in which the n dimensions are transformed into 2 or 3 new attributes. The objective of these techniques is to preserve, in visualization space, the same relationships existing in original space. Ideally, the more similar the instances/points, the closer they are plotted and vice versa. Some examples of this type of techniques in the literature are: Multidimensional Scaling (MDS) (COX; COX, 2008), Principal Component Analysis

(PCA) (HOTELLING, 1933), Least Square Projection (LSP) (PAULOVICH et al., 2008) and Local Affine Multidimensional Projection (LAMP) (JOIA et al., 2011).

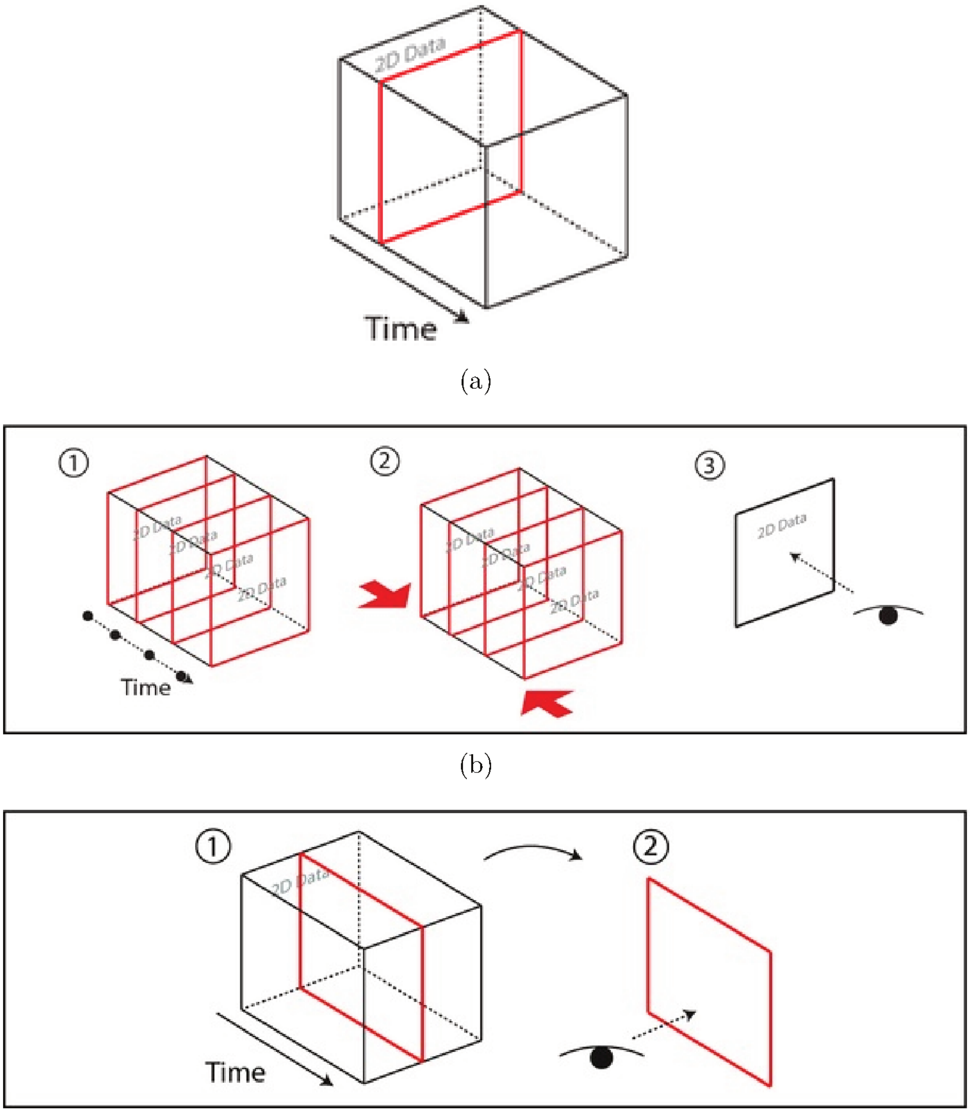
Information Visualization is employed in several data analysis applications of different domains, such as finance (WANNER et al., 2016), business (KO et al., 2012), sports (LOSADA; THERÓN; BENITO, 2016), politics (SCHREDER et al., 2016), health (CLAES et al., 2015) and scientific data (MA et al., 2012), with satisfactory results in terms of communicating relevant information to the user. Information Visualization represents a potential tool for surveillance area, revealing patterns that illustrate several important aspects for video analysis. Section 2.4 illustrates several examples of existing applications.

2.3.1 Temporal Data Visualization

Temporal data include information distributed over time domain. Visualizing temporal data is a hard task, especially when data present more than one dimension in addition to time (BACH et al., 2014). Bach et al. (2017) present a descriptive model for visualization of temporal data: the space-time cube (see Figure 3(a)). The conceptual space-time cube is a three-dimensional Euclidean space, consisting of a two-dimensional space and time. The red square in Figure 3(a) represents the time slice, a temporal snapshot extracted from the cube. Temporal data visualization techniques can be described as operations on the cube, as described as follows:

- *Time Flattening*: merges time slices in a single image, by collapsing the cube along time axis. This operation however often generates images with information overload;
- *Discrete Time Flattening*: seeks to solve information overload problem, by selecting only meaningful time slices, and then merge them. Figure 3(b) demonstrate how *Discrete Time Flattening* operation works. The process basically consists of identify and merge meaningful time slices (red squares) from the videos, resulting in a image that is analyzed by the user;
- *Space Cutting*: consists in extracting a planar cut in a direction perpendicular to 2D data plane. Figure 3(c) illustrates the planar cut (red square) extracted from space-time cube and further visualized by the user.

Vrotsou, Forsell e Cooper (2010) use the space-time cube for analyzing individuals activity diaries. The data used in this work consists of handwriting activity time diaries, which can be used for comprehension of a population daily habits. The space-time cube representation (see Figure 4(a)) illustrates all activities performed by individuals along the day, highlighting when they start/finish. The x-axis represents the individuals, the y-axis represents the time and the z-axis is used to display the activities, one at a time.



(a)

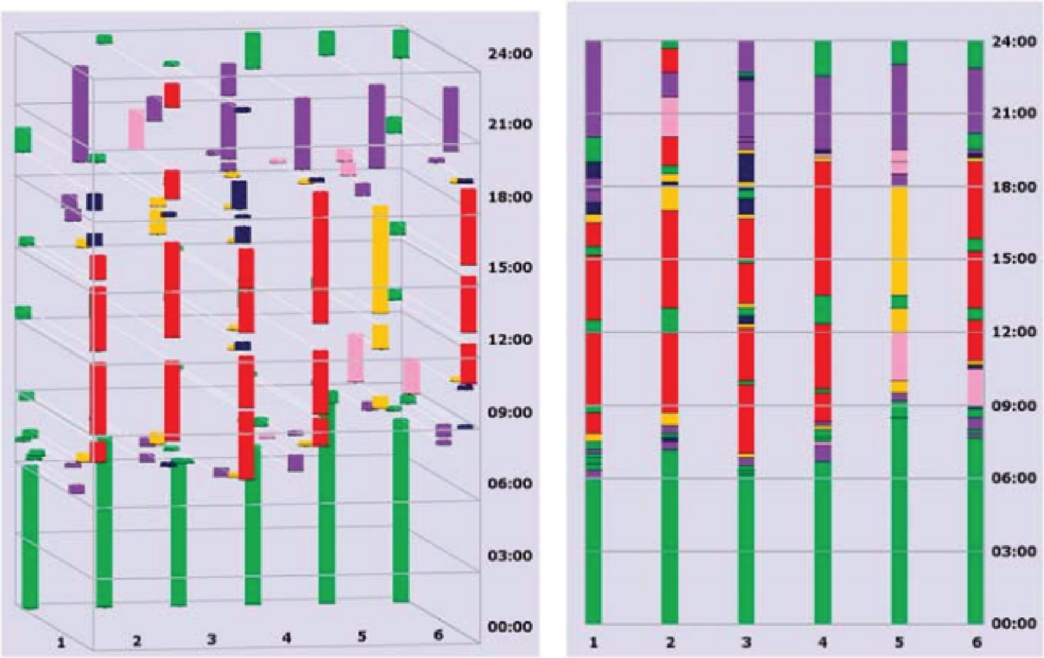
(b)

(c)

Figure 3 – Descriptive model for temporal data visualization and its operations. (a) Space-time cube conceptual model (b) *Discrete Time Flattening* operation (c) *Space Cutting* operation. (Figures adapted from (BACH et al., 2017)).

Colors are also used to distinguish the activities, allowing the transformation of cube information in a two-dimensional space without losing information about event types, as shown in Figure 4(b).

The visual exploration and analysis of temporal data in healthcare allows the execution of important tasks. Afroz e Morshed (2018) propose a visualization for temporal health data provided by a smartphone app. The physiological data are collected from users smartphones and send to a server that generate visualizations for these health data, in order to measure the severity of people health conditions over the time. The temporal plot visualization is composed of two views: graph and flow. In Time Plot Graph (see Figure 5(a)) y-axis represents Events of Interest (EoI) severity and x-axis represents date

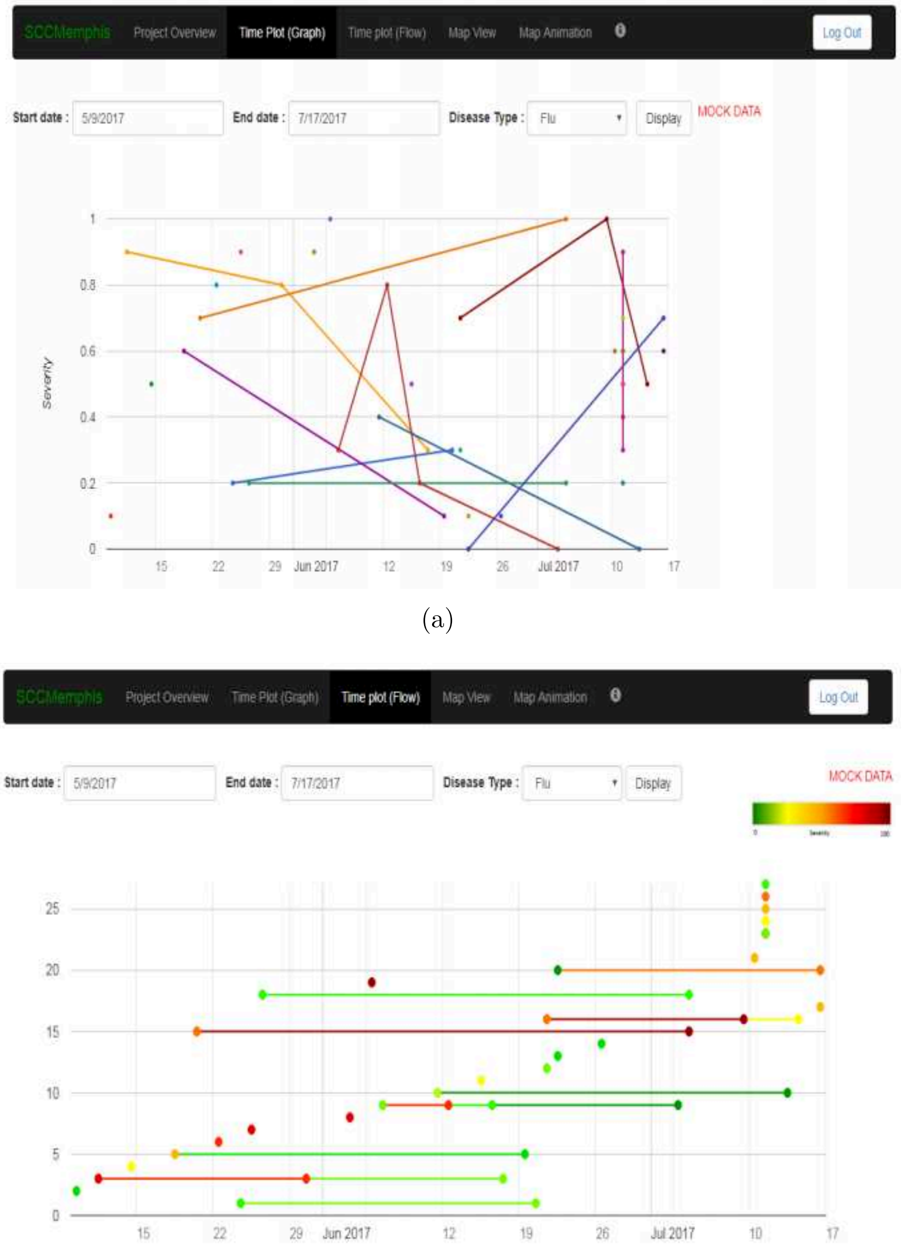


(a) (b)

Figure 4 – Visual representation of individuals activities. (a) shows a space-time cube and (b) a 2D space, both illustrating all activities performed by individuals along the day. (Figures adapted from (VROTSOU; FORSELL; COOPER, 2010)).

and time. Each point represents a data collection and their colors distinguish the patients. The points are connected to keep continuity of patient data, providing a gradual report of patient health status. Time Plot Flow (see Figure 5(b)) shows patient EoI severity flow over time. The x-axis represents time and y-axis represents patients id. The points are colored according EoI severity values and the colors are ranged from dark gray (low severity) to dark red (high severity).

Traffic data visualization can facilitate moving objects behavior comprehension and traffic patterns discovery. Traffic data are generated from vehicles and objects movement, depicting its temporal and spatial properties. Wang et al. (2014) present a visual analysis system to explore traffic data recorded by transportation cells located at roads of a city. The system estimates some statistical properties, such as speed, flow volume and abnormality, for each cell location. In order to observe these information over the time, the system provides the Pixel Table View (see Figure 6), which summarize daily temporal patterns according to properties. Table rows represent days of a month and each column represents 10 minutes. The table cells are colored according properties values, and if a property value at a period of time is unknown its cell is colored in gray.



Start date : 5/9/2017 End date : 7/17/2017 Disease Type : Flu

(a)

Start date : 5/9/2017 End date : 7/17/2017 Disease Type : Flu

(b)

Figure 5 – Temporal visualization of health data. (a) shows a Time Plot Graph example and (b) a Time Flow Graph example. (Figures adapted from (AFROZ; MORSHED, 2018)).

2.4 Related Work

Automatic video analysis techniques reduce the amount of video information analyzed by human operators, but they are still responsible for solving video ambiguities, summarizing video context information and perform decision making (RÄTY, 2010). The design of interactive visualizations can support information synthesis and decision making for surveillance tasks. Existing approaches propose surveillance systems employing visualiza-

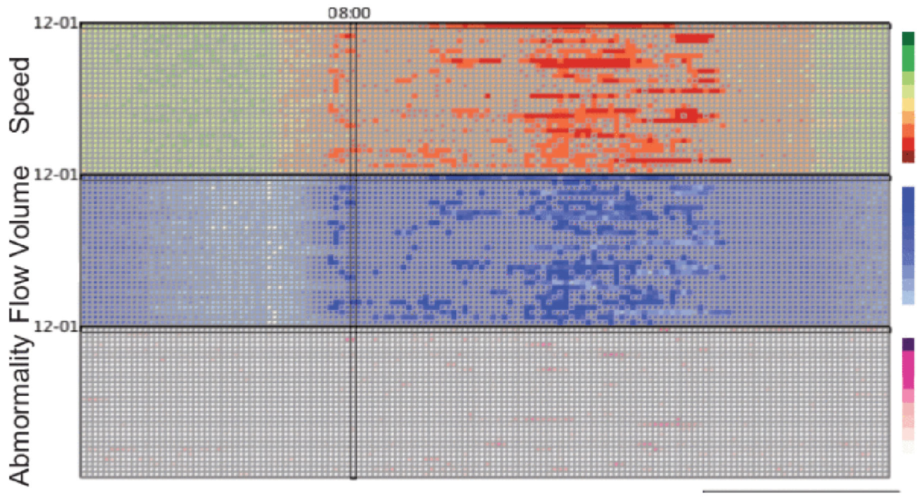


Figure 6 – Pixel Table View. (Figure extracted from (WANG et al., 2014))

tion techniques in order to analyze different aspects, such as objects, trajectories, scene and events. The works in surveillance visualization are proposed focusing on most of the tasks presented in Section 2.2.

Several works employ visual strategies to explore trajectories in videos. Höferlin, Höferlin e Weiskopf (2009) propose a framework for video analysis, in which the video is visualized as a *VideoPerpetuoGram* (VPG) and the user can interact with the VPG defining filters. The VPG shows key frames between sparse equidistant spaces which represent the time slices of the video. The objects trajectories are traced in the key frames and sparse spaces, thus the user can visualize the moments in which each object is in the video and its exact location in the key frames. Figure 7 shows a video visualization with VPG. In this figure, three key frames are shown with their respective video times. The blue bar represents the video duration time and the pink and beige lines represent two different trajectories. Users can filter trajectories according to properties such as: camera location, start and end position in image coordinates, speed and direction. Filters can be combined using logical AND and OR operations. The filters allow displaying only relevant trajectories. To verify filters efficiency, Höferlin, Höferlin e Weiskopf (2009) defined a task of identifying a meeting between two people in a 10-hour surveillance video. At the beginning, 809 trajectories were displayed. A filter capable of detecting the division and union of trajectories was applied and reduced the number of trajectories to 22.

Exploring the space-time cube concept, Meghdadi e Irani (2013) present a visual analytics system for interactive exploration of surveillance video. The approach provides multiple views of information related to moving objects in a video. The views consist of action shot image and space-time cube. The action shot image shows all the positions the object occupied in the video. Moving objects are extracted from each frame and added to the background image. A new segment of moving object is added to image when it does

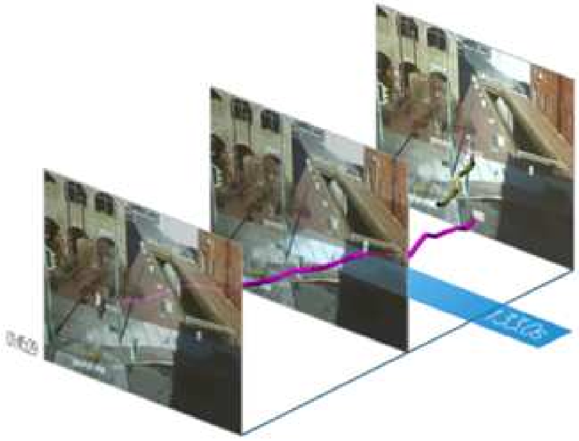
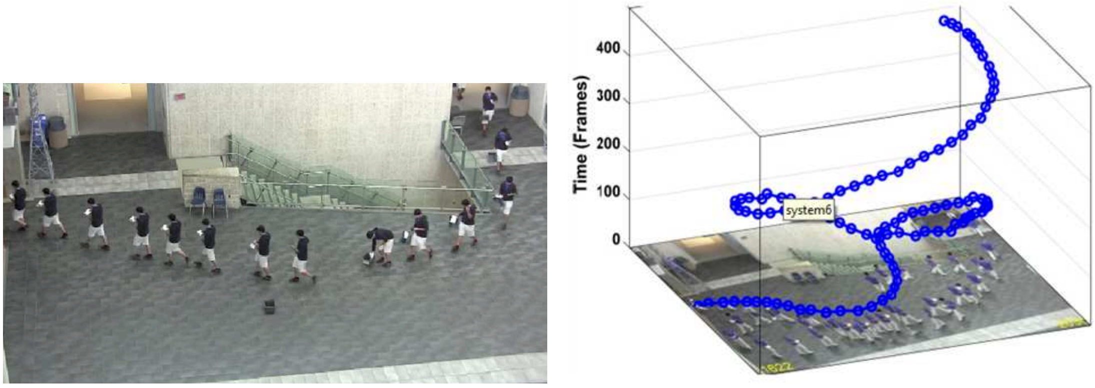


Figure 7 – Video visualized with *VideoPerpetuoGram* (VPG). (Figure extracted from (HÖFERLIN; HÖFERLIN; WEISKOPF, 2009))

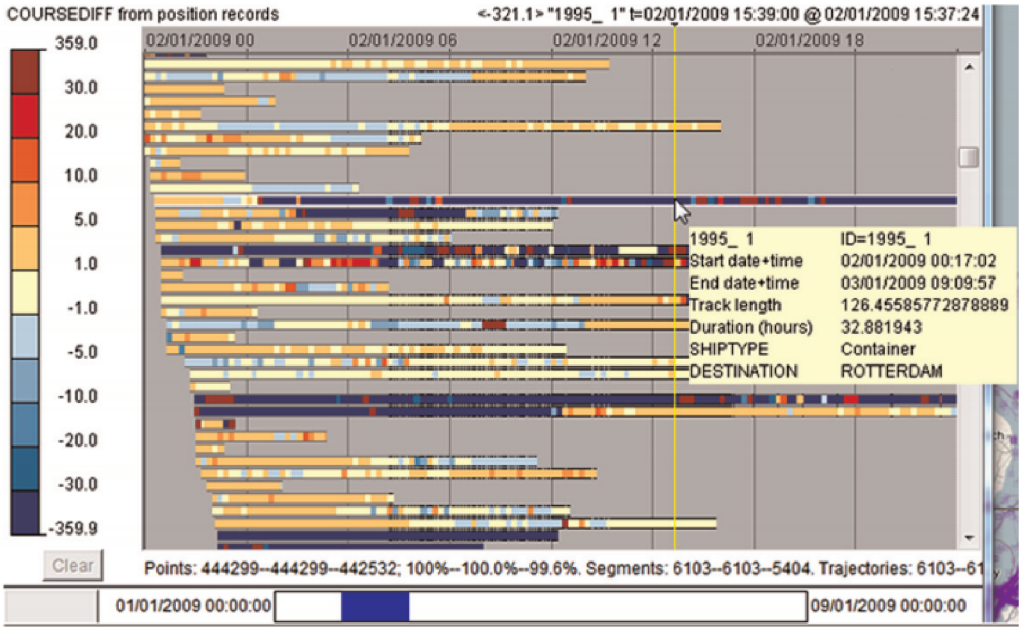
not overlap the previous segment. Figure 8(a) shows an example of action shot image, and its possible to see all the positions the objects occupied over the time. The space-time cube represents spatial and temporal aspects of the video, in which the axes x and y represent video space (background) and the z-axis represents the video time in frames. Thus, the object position is plotted considering space and time. Figure 8(b) shows a space-time cube example. The space-time cube display only one trajectory at time, thus it is difficult to analyze some situations in video, such as meeting between objects.



(a) (b)

Figure 8 – Views proposed by (MEGHDADI; IRANI, 2013). (a) shows action shot image and (b) shows space-time cube.

Andrienko e Andrienko (2013) illustrate several techniques for visual analysis of motion data, aiming to describe general approaches to analyze these data. The authors analyze ship routes data, but their approaches can be employed in several domains containing motion objects. To represent temporal aspects of motion data, the *time bars display* technique represents ship trajectories using bars, the horizontal dimension represents time and the vertical dimension is used to arrange trajectories, which can be ordered according to one or more attributes. The bars are divided into segments colored according to some attribute selected by the user. Attribute values are divided into ranges and each range is assigned to a color. *Time bars display* allows to observe ships trajectories duration and trajectories attributes behavior over time. Figure 9 shows an example of *time bars display* view in which the trajectories are ordered by the start time and the bars segments colors represent values of an attribute called “course difference”, which expresses the difference between the ship heading and its actual direction of movement at each trajectory point, in which darkest color intensities corresponding to larger differences.



<-321.1> "1995_ 1" t=02/01/2009 15:39:00 @ 02/01/2009 15:37:24

1995_ 1	ID=1995_ 1
Start date+time	02/01/2009 00:17:02
End date+time	03/01/2009 09:09:57
Track length	126.45585772878889
Duration (hours)	32.881943
SHIPTYPE	Container
DESTINATION	ROTTERDAM

Points: 444299--444299--442532; 100%--100.0%--99.6%. Segments: 6103--6103--5404. Trajectories: 6103--61

01/01/2009 00:00:00 09/01/2009 00:00:00

Figure 9 – *Time bars display* view. (Figure adapted from (ANDRIENKO; ANDRIENKO, 2013))

According to Ahmed et al. (2018) trajectory clustering is often used for long duration video analysis. Clustering trajectories is not a trivial task due to factors such as inaccurate object tracking, movement patterns with great variation, and large number of trajectories. Clustering trajectories allows the visualization of object movement patterns that may help to understand their behavior. Several approaches employ trajectory clustering for video visual analysis. Kumar et al. (2017) introduce a method for trajectory clustering using hierarchical cluster algorithms based on Visual Assessment of Tendency (VAT), in which the trajectories are initially clustered ignoring the direction, and then the obtained clusters are divided considering the direction of trajectories. The Figure 10(a) shows all the trajectories detected in a video and the Figure 10(b) shows the result of trajectories clustering obtained by the method, in which each cluster is represented by a color.

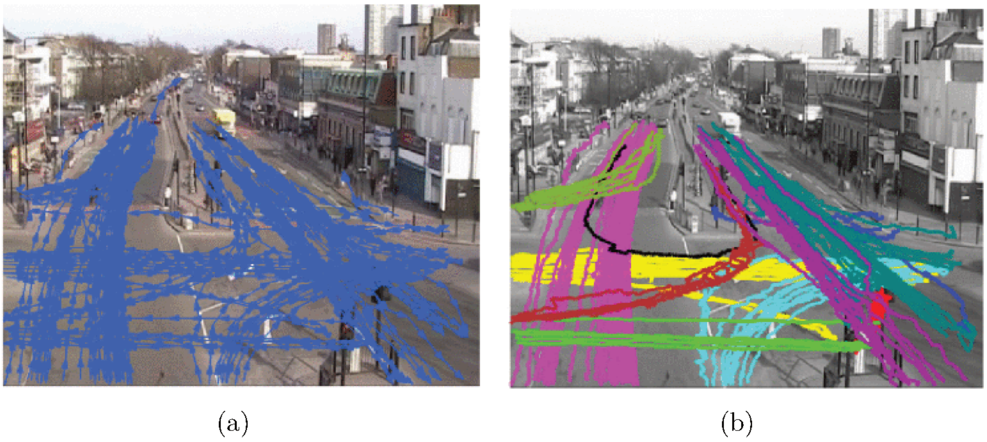
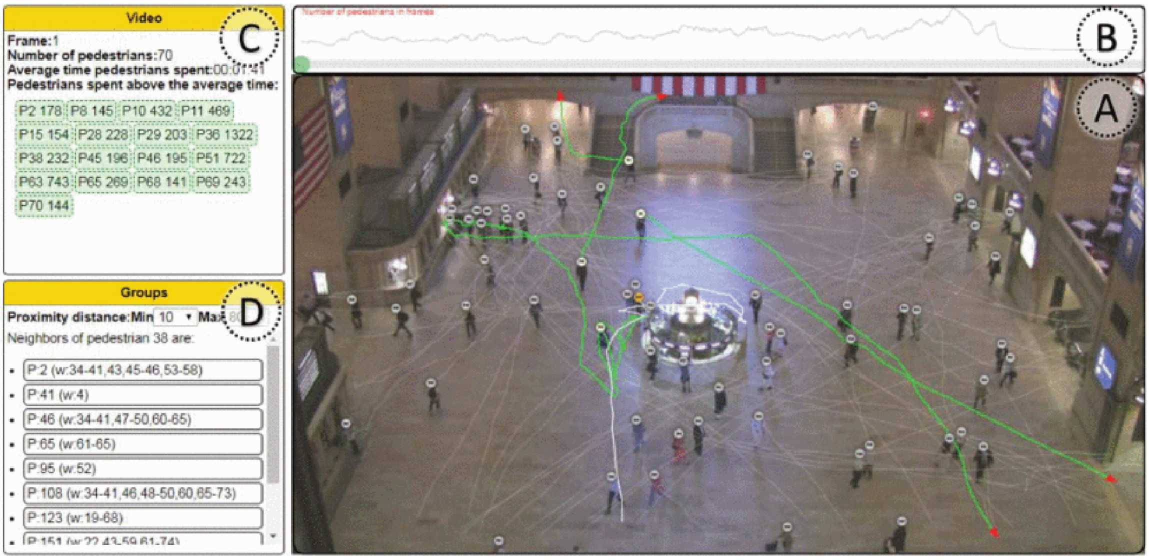


Figure 10 – Trajectories clustering using VAT-based hierarchical clustering algorithms. (a) shows the all trajectories present in the video and (b) the result of clustering trajectories by the proposed method, in which each cluster is represented by a color. (Figures extracted from (KUMAR et al., 2017))

Trajectolizer (SAWAS et al., 2018) is a system which draws trajectory group patterns over the scene background, providing an interactive visualization of group trajectory dynamics over time. Figure 13 shows Trajectolizer interface. The system is composed of four views: a video frame panel showing current frame with objects ID and trajectories, a timeline slider which allows to navigate to any video frame and to observe the number of objects in the scene, a descriptive analytics panel which displays current frame information and a group information panel which contains analysis of object group behavior.



Frame:1
Number of pedestrians:70
Average time pedestrians spent:00:01:41
Pedestrians spent above the average time:

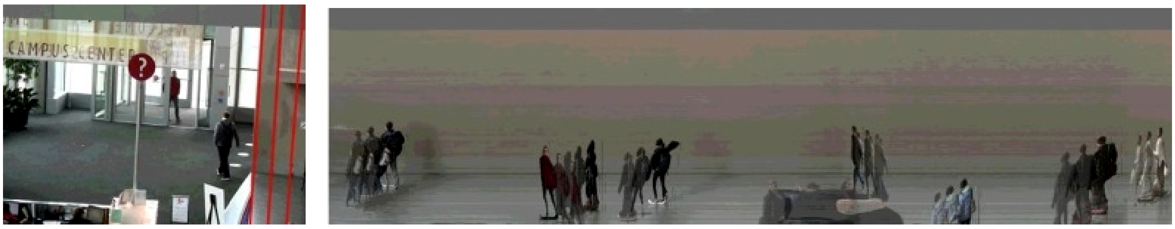
P2 178 P8 145 P10 432 P11 469
P15 154 P28 228 P29 203 P36 1322
P38 232 P45 196 P46 195 P51 722
P63 743 P65 269 P68 141 P69 243
P70 144

Proximity distance:Min 10 ▾ Max 80
Neighbors of pedestrian 38 are:

- P:2 (w:34-41,43,45-46,53-58)
- P:41 (w:4)
- P:46 (w:34-41,47-50,60-65)
- P:65 (w:61-65)
- P:95 (w:52)
- P:108 (w:34-41,46,48-50,60,65-73)
- P:123 (w:19-68)
- P:151 (w:22,43-59,61-74)

Figure 11 – Trajectolizer interface composed of: (A) video frame panel, (B) timeline slider, (C) descriptive analytics panel and (D) group information panel. (Figure extracted from (SAWAS et al., 2018))

Some approaches focus on visualization of identified objects in the video. Visualize objects allows analyzing specific objects and understanding their dynamics, such as motion patterns, places occupied and occurrence. Bagheri, Zheng e Sinha (2016) converts surveillance videos into a temporal profile in order to visualize the dynamics of object targets which pass into critical region of surveillance videos. Multiple sampling lines are defined at critical locations, consequently extracting multiple temporal slices. Multiple temporal slices allow the insertion of more spatial information in temporal profile and a better visualization of targets movement direction. After extracting temporal slices, they are combined according spatial locations to create a combined temporal profile representing the foreground dynamic flow. Figure 12 shows an example of a frame from a surveillance video (Figure 12(a)) and the resulting layout with the produced temporal profile generated by multiple temporal slices (Figure 12(b)).



(a) (b)

Figure 12 – Example of a temporal profile showing pedestrians passing through a doorway. (a) shows a frame with sampling lines (red lines) and (b) its generated temporal profile. (Figures extracted from (BAGHERI; ZHENG; SINHA, 2016))

Zhang et al. (ZHANG et al., 2019b) propose a multi-scale visualization for interactive exploration of surveillance data. The visualization is composed of coordinated views which provide, for a set of target objects selected by the user, a timeline depicting the frequency of occurrence of these objects (Figure 13(a)), an object recognition view (Figure 13(b)) and a frame representation to contextualize these objects in the video (Figure 13(c)).

Generally, video summarization approaches seek to automatically find key frames, in which important actions occur, and present them to users, in order to show the main events occurred in video as an "events summary". Visualization techniques can aid summarize video in order to allow the observation of general video structure, representing all video frames, not just key frames. Mendes, Paiva e Schwartz (2019) present a methodology for video summarization with focus on event identification, employing a point-placement visualization technique to highlight events spatial aspects and a Temporal Self-similarity Maps (TSSM) to explore the temporal aspects. The points in point-placement view (see Figure 14(a)) represent video frames, and the Euclidean distance among points reflects their similarities according to event occurrence, from a spatial perspective. TSSM view (Figure 14(b)) shows difference between frames content mapping a color coding according

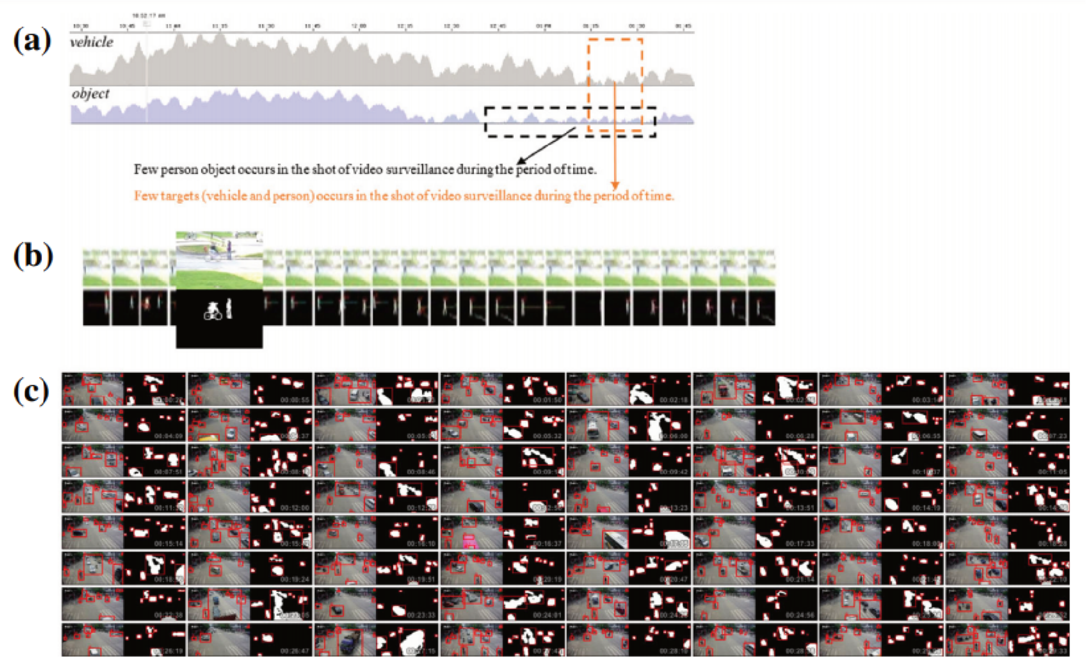
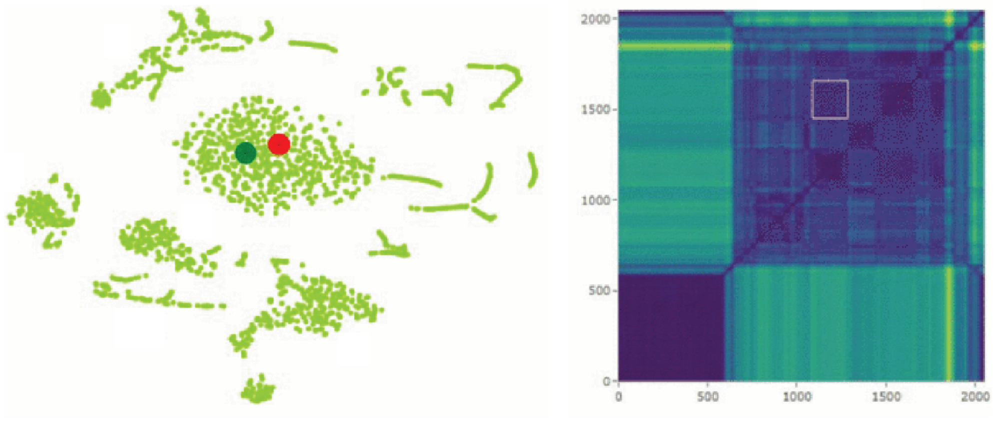


Figure 13 – Multi-scale visualization. (a) shows the scrollbar of the data flow clustering, (b) object recognition view and (c) frames representation (Figure extracted from (ZHANG et al., 2019b))

to similarity values, in which the higher the similarity, the lower the intensity and vice versa.



(a) (b)

Figure 14 – Video summarization employing a point-placement visualization (a) and a TSSM view (b). (Figures extracted from (MENDES; PAIVA; SCHWARTZ, 2019))

Bach et al. (2015) introduce time curves for visualizing evolution patterns in temporal data of different domains. The authors employ time curve technique into a surveillance video in order to detect sudden changes in video and observe its structure. Each point in time curves represent a video frame or groups of adjacent video frames. The similar points are presented close to each other and interesting moments may be identified as outliers on time curve. The Figure 15 shows the result of time curve technique in a surveillance video. By looking at the central cluster, one notice that few changes occur during the video extent. Sparse points represent frames in which pedestrians cross the scene.

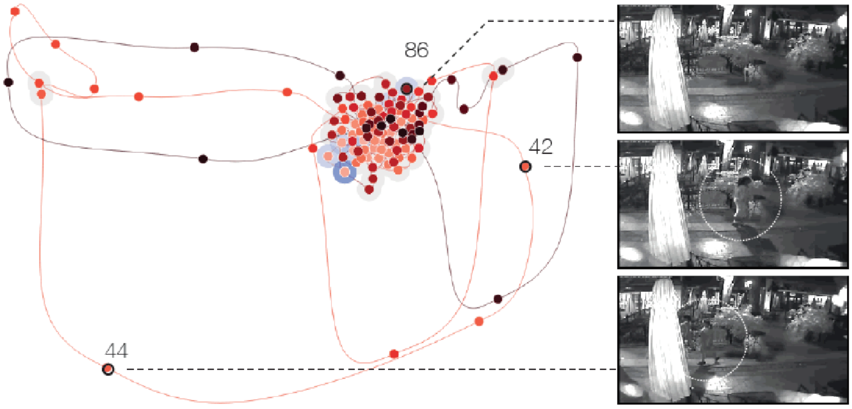


Figure 15 – Time curves. (Figure adapted from (BACH et al., 2015))

2.5 Final Remarks

The presented approaches focus on analyzing several aspects of the surveillance videos, such as the events and trajectories. However, few works are focused in identifying how objects behave when they are in scene, and how these behaviors are related to the occurrence of the events.

In this sense, this work propose a set of interactive layouts to focus exactly on the relationship among the identified objects, as well as all the important information regarding the presence of them in scene. We believe that, as main actors of the events that occur in these videos, the highlight of such aspect may reveal relevant information for security agents, helping in their decision making.

Chapter 3

System Description

We developed a computational system to visually analyze surveillance videos, focusing on the behavior and relationship of the objects in these videos. The system is originally designed to perform a post-event surveillance analysis, using previously generated videos. Based on the nature of this analysis and on existent works from the literature, we outlined a set of requirements that our proposed system must fulfil, which we categorize in in three types: general requirements, temporal analysis requirements and spatial analysis requirements. These requirements are detailed as follows:

General Requirements

- ❏ **GR1:** Show the video content in a way that allows the domain expert to explore the video story quickly and effectively;
- ❏ **GR2:** Efficiently generate a video representation within a reasonable period of time that is affordable to the users;

Temporal Analysis Requirements

- ❏ **TR1:** Provide the analysis of multiple temporal aspects (such as time in scene, interactions and speed) of the identified objects, all of them related to their entire presence in the video;
- ❏ **TR2:** Provide an exploration of meetings between objects, including these meetings instants, duration and objects involved;
- ❏ **TR3:** Provide the analysis of the objects speed distribution during their presence in the video;

Spatial Analysis Requirements

- ❏ **SR1:** Provide the analysis of positions occupied by the objects in the scene;
- ❏ **SR2:** Provide the analysis of the objects meeting positions;

❏ **SR3:** Provide the analysis of user defined regions of interest in the scene.

Figure 16 illustrates our proposed analysis workflow. The first stage consists of employing a strategy to identify and track objects from a previously generated video. Addressing the quality of these methods is beyond the scope of this work, and we consider the use of proper methods for such tasks. We then use the data produced by this identification/tracking process to generate the layout. We also extract all video frames, which will be used for interaction purposes. Finally, the user can interact with the layout, executing a variety of basic exploration tasks with all detected objects, such as timeline zoom/pan, objects selection, as well as more sophisticated interactions which will be detailed in Section 3.1.

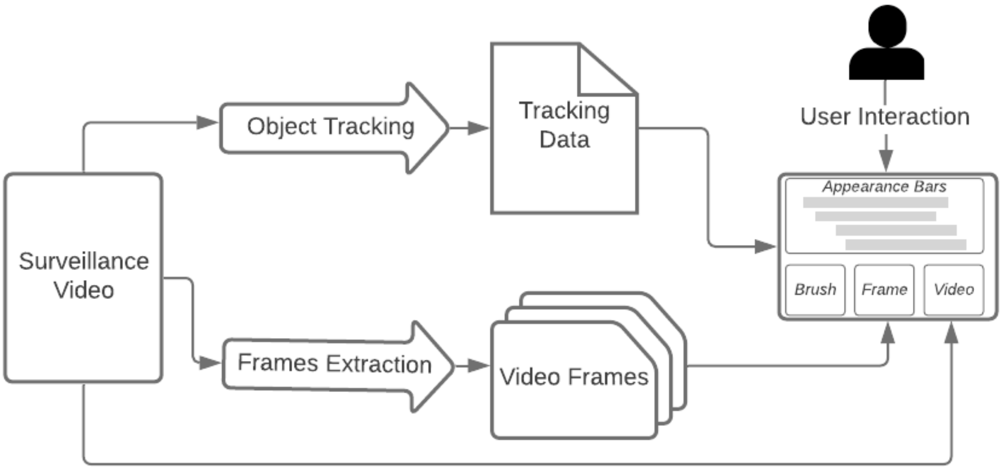


Figure 16 – Video analysis workflow.

The system interface is shown in Figure 17, and provides the following views: *Appearance Bars View* (A), *Brush View* (B), *Frame View* (C) and *Video Player View* (D).

The *Appearance Bars View* (A) presents an overview of all detected objects, showing their scene presence moments over the entire video duration, and highlights their behavior during this presence, including their movement patterns and the relationship among them. The x axis depicts the video duration and the V axis lists all the identified objects. For each object a bar or a set of bars is associated, named appearance bar(s), illustrating the object presence distribution over the video duration. The extent of each bar indicates the instants in which the associated object enters/leaves the scene. Several interactions are available in *Appearance Bars View*, which also displays the results of most of the interactions. These interactions will be detailed in Section 3.1. All instants selected by the user in the appearance bars are reflected in the *Frame View* (C) and *Video Player View* (D), allowing him/her to investigate specific observed pattern directly on the video contents,

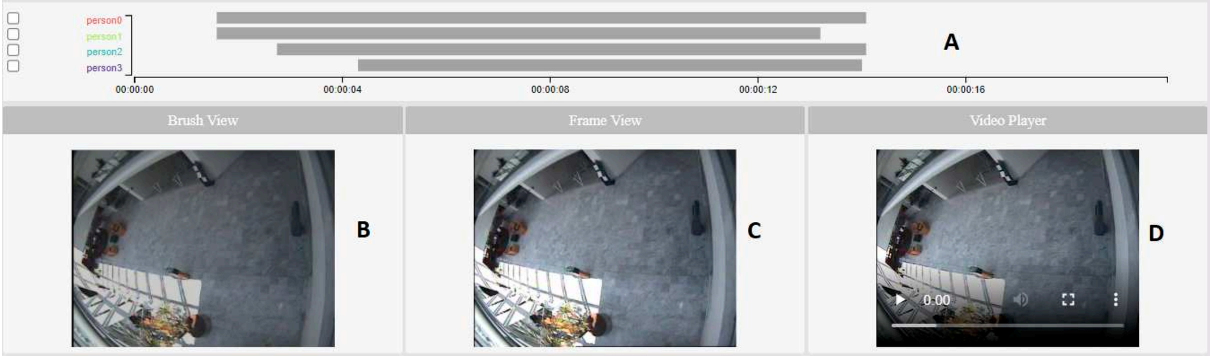


Figure 17 – System interface. *Appearance Bars View* (A) shows the objects appearance bars, *Brush View* (B) allows to select a region of the scene, *Frame View* (C) shows frames from the video, *Video Player View* (D) contains a video player that allows to watch the video.

and vice-versa. Hovering the appearance bars displays a tooltip with details related to the task context, which may include the object label, objects that are participating in a meeting at that instant, among other information. The *Brush View* (B) allows the user to select a region of interest from the video background in order to filter objects which crossed the region. The *Frame View* (C) allows the user to observe details of a particular instant of the video selected on the appearance bars or exhibited on the *Video Player View*, highlighting the bounding boxes of the objects identified in the respective frames. The *Video Player View* (D) implements a traditional interaction tool used for watch and navigate in videos, allowing the user to play/pause the video execution, advance/rewind frames, turn on/off the sound and set the video in full screen or in picture-to-picture. When the user interacts with the appearance bars by clicking on a specific instant, the video player is set to the selected instant.

3.1 Analysis Interaction

This section presents a set of interaction tools to enhance the exploration of the proposed layout, as well as to highlight strategic patterns present on the data, allowing for an effective events analysis in the video. These tools are described in the following sections.

3.1.1 Meeting Analysis

Meeting among objects represent an important analysis aspect to comprehend the relation among these objects during their presence. In our proposal, a meeting between two objects is defined as the occurrence of an intersection between their correspondent bounding boxes for a minimum consecutive time interval. An intersection is detected when at least one pixel of both bounding boxes coincide at the same frame. Figure 18 shows an example of two bounding boxes intersection. Users are able to define the minimum intersection time to be defined as a meeting, which allows different perspective analyses.

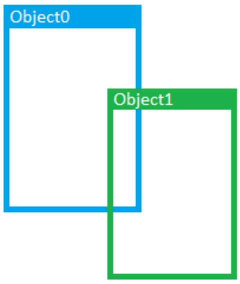


Figure 18 – Bounding boxes intersection between two objects.

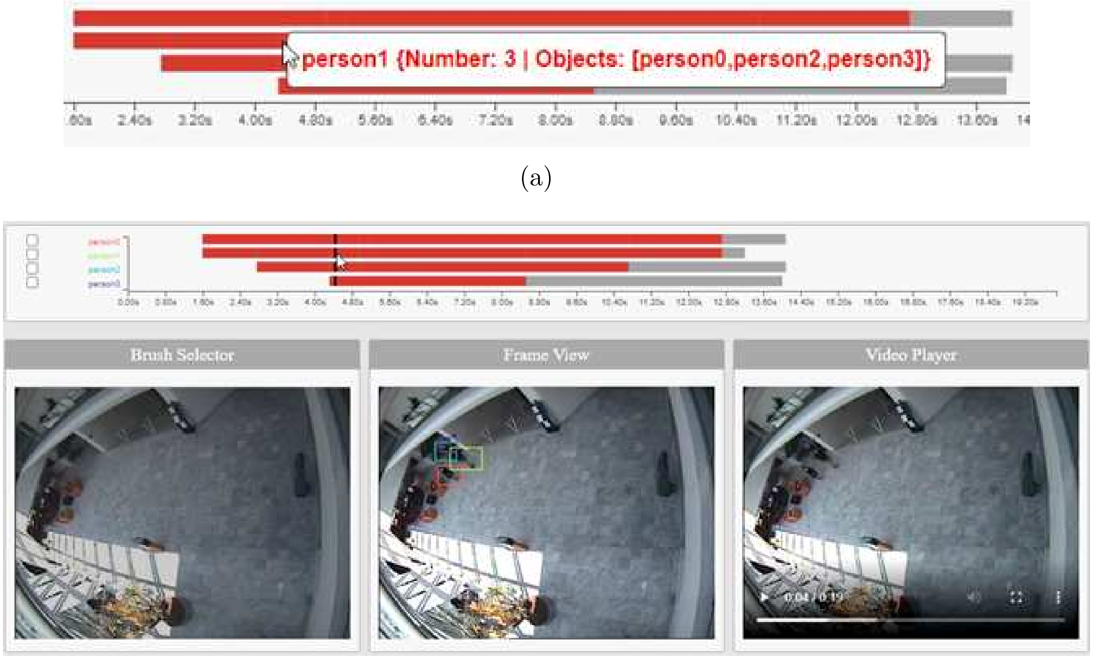
The system provides two meeting analyses: *Show All Meetings* and *Show Meetings*, described as follows.

Show All Meetings

The *Show All Meetings* functionality provides a general view of objects meetings, highlighting the moments in which each object participates in meeting. All these moments are highlighted in a red layer over the appearance bars, in the portions which represent these moments. By hovering the mouse over an object bar, one can notice the number of objects with which it met, as well as a list containing all these objects, considering the instant hovered by the mouse (Figure 19(a)). Users may also see which objects met with a specific object in a specific instant, by clicking in the object appearance bar. These meetings are then highlighted by black markers on the appearance bars of the objects which participated in this meeting (Figure 19(b)). The *Frame View* shows the correspondent instant frame, and the identified objects bounding boxes are highlighted to allow their identification.

Show Meetings

The *Show Meetings* functionality allows viewing all meetings among specific objects selected by the user. Users may select objects by the correspondent checkboxes in the *Appearance Bars View*. When a single object is selected, all the moments in which this object participate in a meeting are mapped to a layer over its appearance bar, in the portions which represent these moments. The color intensity of these layers are proportional to the number of object with which it meets. On the other hand, the moments in which the other objects meet the selected object are mapped to a dark gray layer over these objects appearance bars, in the portions which represent these moments. Yellow marks are used to highlight the instants in which the meetings change somehow, either by removing or adding new objects. The resulting layout is shown in Figure 20(a). When the user selects



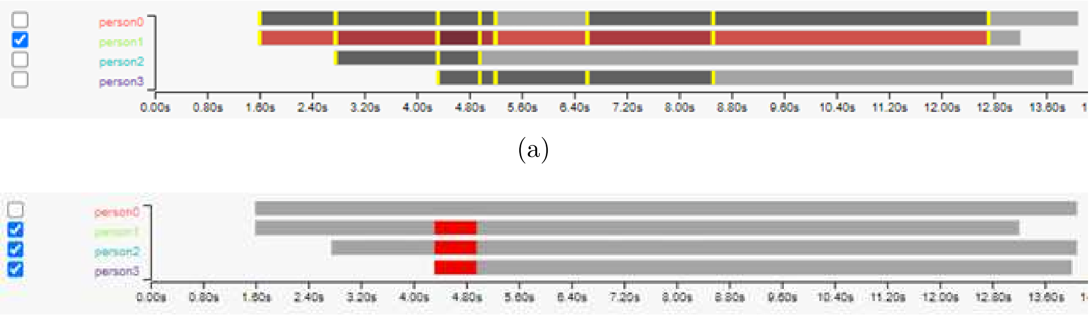
person1 {Number: 3 | Objects: [person0,person2,person3]}

(a)

(b)

Figure 19 – (a) shows how many and what objects participate in the meeting by hovering the mouse on the appearance bar and (b) shows the selection of an instant by clicking on the appearance bar and how this modifies the layout.

multiple objects, the moments in which all these objects meet are mapped to a layer over their appearance bars, in the portions which represent these moments. In this case, all the corresponding bounding boxes must intersect with each other simultaneously, for at least the previously defined minimum consecutive time interval. An example of selecting multiple objects and viewing their meetings is shown in Figure 20(b).



(a)

(b)

Figure 20 – Examples of *Show Meetings* functionality results. (a) shows the result of selecting one object and (b) shows the result of selecting three objects.

3.1.2 Sorting

The *Show Sorting* functionality allows users to sort, in descending order and in a top-down organization, the appearance bars, according to a variety of aspects. This functionality ranks objects according a set of temporal aspects, such as permanence in scene and time spend in meetings (**TR1**), and some objects meetings aspects, such as distinct objects met and number of meetings of an object (**TR2**). The default ordering is the first presence time. Each ordering aspect is described as follows:

- ❏ *Scene Permanence:* The frames in which an object is detected are counted, resulting in its total scene permanence value. The objects are then sorted according to their scene permanence values;

- ❏ *Number of Meetings:* All the objects meetings are calculated and counted using the same strategy described in *Show All Meetings* functionality (Section 3.1.1). It is important to highlight that meeting changes, in terms of addition/removal of new participant objects are considered as new meetings. Thus, the meeting counting for an object is performed by counting the number of meeting changes considering all its scene appearance time;

- ❏ *Distinct Objects Meetings:* All the objects meetings are calculated using the same strategy described in *Show All Meetings* functionality (Section 3.1.1). Whenever an object participate in a meeting with another object that it has never met before, a new distinct object is considered. Thus, the number of distinct objects meetings is achieved by counting the number of distinct objects met by an object during all its scene presence time;

- ❏ *Time in Meetings:* The frames in which an object participate in meetings are counted, resulting on its total meeting time value. The objects are then sorted according to their total meeting time values.

3.1.3 Speed Analysis

This tool allows users to analyze the objects movement speed variation during their presence in the scene. Users may define time intervals (in seconds) for which an average speed is calculated, providing an analysis in distinct time resolutions. The resulting average speed for each time interval is then mapped to the correspondent time portion of the appearance bar, whose color intensity is proportional to the average speed value, as shown in Figure 21.

The average speed in each time interval is calculated considering the Euclidean distance (in pixels) between the bounding boxes centers positions in consecutive frames of

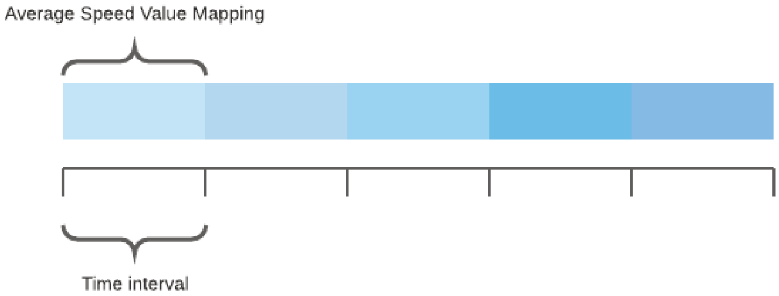


Figure 21 – Result of mapping average speed.

the interval. Considering an object k, whose bounding box center is C_k, a user defined time interval as t, the average speed S in the interval t is calculated as follows:

$$S = \frac{\sum_{i=f_1}^{f_{|F|-1}} d_{(i,i+1)}}{t}, \tag{1}$$

where $F = \{f_1, f_2, ..., f_{|F|}\}$ represents the frames set in t and

$$d_{(i,j)} = \sqrt{(C_{k_{x_i}} - C_{k_{x_j}})^2 + (C_{k_{y_i}} - C_{k_{y_j}})^2}$$

represents the Euclidean distance between C_k in frames i and j.

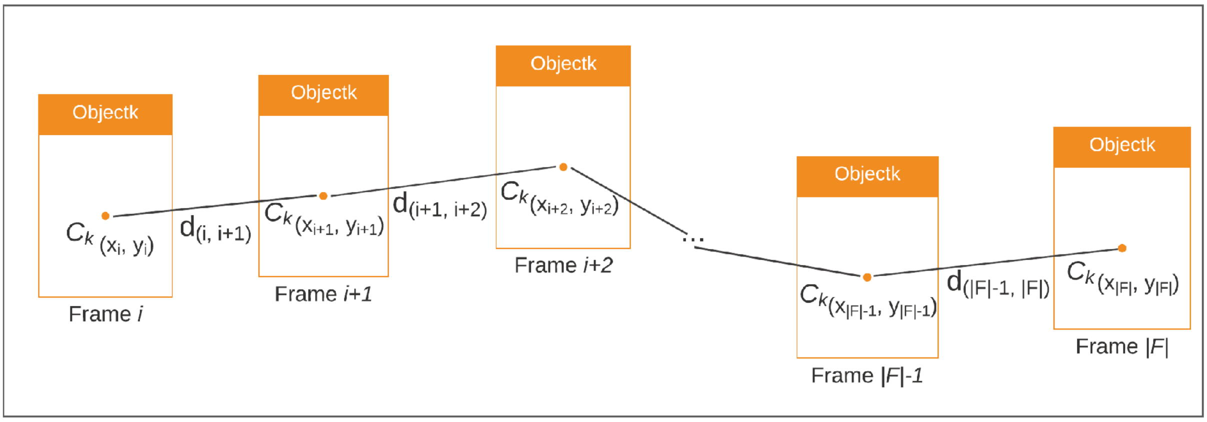


Figure 22 – Calculation of the distance (in pixels) travelled by an object during an user defined interval (in seconds).

It is important to highlight that the result is an average speed in a pixel/second speed measure. Although the resulting value presents no correspondence with employed real world speed units, the idea here is only to highlight the speed variation of an object, as well as to relate speeds from different objects. If the last time interval of the object presence is shorter than the time interval defined by the user, the average speed is calculated considering the remaining time interval of the object presence. An example of resulting layout is shown in Figure 23.

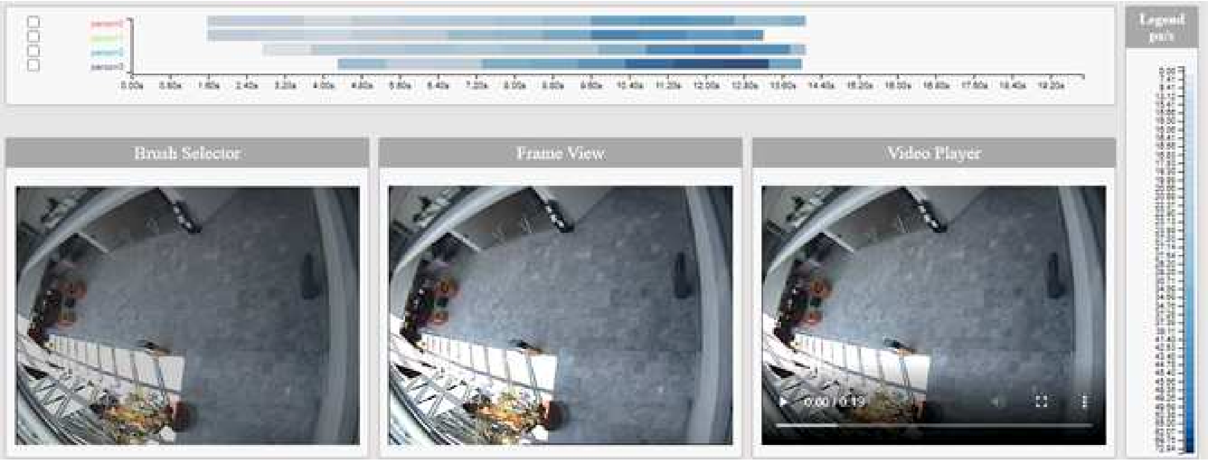
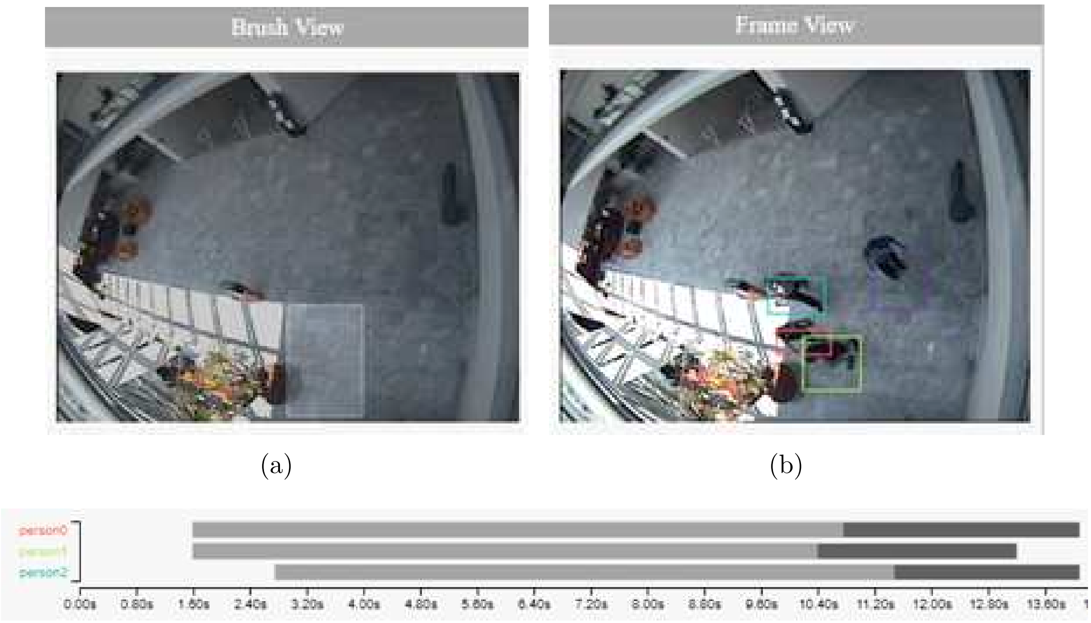


Figure 23 – Example of the layout result obtained by *Show Speed* functionality with a time interval of 1 second.

3.1.4 Scene Region Filtering

The *Show Filter by Scene Region* functionality allows the user to select a rectangular region of interest in the video background and filter the objects which crossed this region in any specific moment. The selection is performed in the *Brush View*, and the results of the filtering is shown in the *Appearance Bars View*. The moments in which the filtered objects crossed the selected regions is highlighted in dark gray. The user is then able to identify which objects in which moments crossed a specific region captured by the surveillance camera. Figure 24(a) shows an example of a region of interest selection, whose filtering result in the *Appearance Bars View* is shown in Figure 24(c). A frame corresponding to the instant in which the three objects cross the region at the same time is shown in Figure 24(b). One can notice that there is another object which does not appear on the resulting *Appearance Bars View*, which means that it does not cross the selected region considering all its scene presence time.



(a)

(b)

(c)

Figure 24 – *Filter by Scene Region* functionality. (a) shows a region of interest selection, (b) shows a video frame and (c) shows the result of filtering the objects by the selected region.

Chapter 4

Experimental Results

In this chapter we present the results of applying our proposed visual strategy to several surveillance scenarios, in order to analyze the objects behavior during the video duration. We first explore the general view of the layouts in order to investigate how the identified objects behave during their presence time. We then refine the analysis exploring the relationships between the objects and temporal/spatial aspects of objects trajectories. We also analyze how previously known events are shown in layout, as well as how the layout represents different event categories. Finally, we identify which of the requirements presented in Chapter 3 our proposal fulfills.

The objects labels shown in the layout are defined by the detector, and do not necessarily represent what the object really is, thus we always refer to them as objects. However, it is important to highlight that an accurate object type detector can enhance the layout capabilities, which becomes even more intuitive/informative, as it allows the expert to make more accurate inferences about the relationship between the objects on the scene, as well as their behavior during the video.

4.1 MeetCrowd

The Context Aware Vision using Image-based Active Recognition (CAVIAR[1]) repository consist of video clips recorded showing different surveillance scenarios, including people walking alone, meeting with others, entering and leaving shopping stores, fighting, leaving packages in public places, among others. We use *Meet_Crowd.mpg* video from this repository, which we name here as **MeetCrowd**. In the **MeetCrowd** video, two people enter the scene together and then two other people enter, also together, join the first ones, walking through the scene as a single group and then leaving it.

[1] http://homepages.inf.ed.ac.uk/rbf/CAVIAR/

4.1.1 Overview

The **MeetCrowd** was filmed in the entrance lobby of the INRIA Labs at Grenoble, France with a wide angled camera lens. The video is composed of 497 frames and a frame rate of 25 frames per second, resulting in almost 20 seconds of video. By watching the video, we manually identified the main events occurring in the **MeetCrowd** video and these events are described in Table 1. Figure 25 shows three key frames from **MeetCrowd** video, representing some of the identified events.

Table 1 – Description of the main events occurring in **MeetCrowd** video.

Event	Frame Interval	Description
1	0-39	The lobby is empty.
2	40-68	Two people enter the scene.
3	69-110	Two other people enter the scene.
4	111-329	Four people cross the lobby together.
5	330-352	Four people leave the scene.
6	353-497	The lobby is empty again.

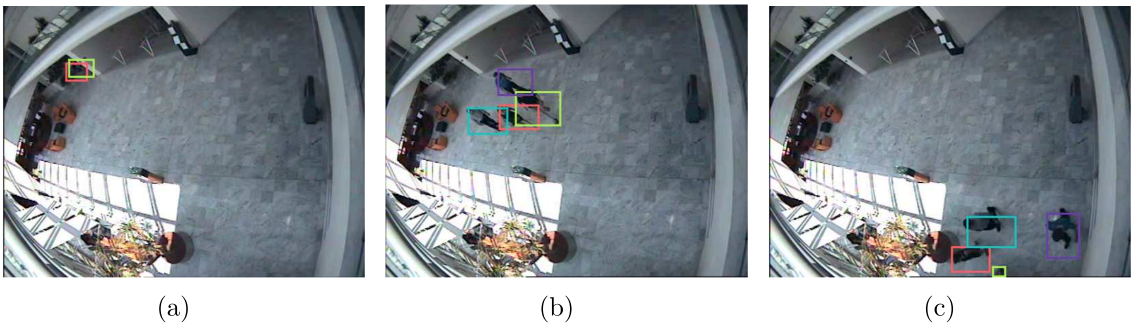

(a) (b) (c)

Figure 25 – Examples of key frames corresponding to events in **MeetCrowd** video. (a) Event 2; (b) Event 4; (c) Event 5.

4.1.2 Analysis

The layout produced from **MeetCrowd** video is shown in Figure 26, and was generated in a short period of time, 2 milliseconds (**GR2**). The *Appearance Bars View* displays four identified objects and their respective appearance bars, which quickly allows the identification of when each object appears (beginning of the bar) and leaves the scene (end of the bar) (**GR1**, **TR1**). It is possible to notice that the appearance bars do not fill the entire timeline, because at some moments there are no identified objects in the scene. It is also possible to notice the order in which objects enter and leave the scene and the portion of time in which all the objects are in the scene at the same time (between 4s and 13s). ***Person0*** and ***person1*** are the first objects to enter the scene, and they enter at the same time. The last objects to leave the scene are ***person0*** and ***person2***, practically at the same time. All the objects are on the scene at the same time in a certain moment and their presence time are relatively similar. It is possible to notice that during 6 seconds (approximately 32% of the video), there is no identified objects on the scene. This occurs at the beginning and the end of the video (Events 1 and 6). In this sense, the layout is able to highlight moments in which all activities occur, allowing the surveillance agent quickly identify what time portions are interesting for analysis.

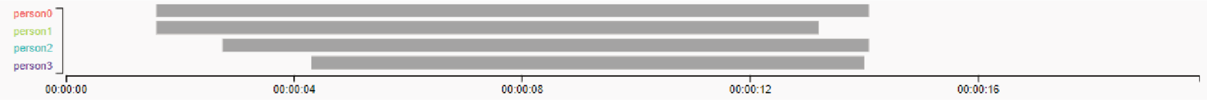


Figure 26 – **MeetCrowd** *Appearance Bars View* layout.

Figure 27 shows the results of highlighting all meetings in the video (**TR2**). It is possible to notice that all objects met another object in at least one specific moment in the video. It is also possible to notice that all objects enter the scene meeting at least one object, which suggests that multiple objects enter the scene together or an object enters the scene and immediately meets other objects. ***Person0*** and ***person1*** enter the scene at the same time, in a moment in which there are no other objects in the scene, which allows us to conclude they enter together (Event 2). It is also possible to notice that ***person0*** and ***person1*** are the last objects participating in a meeting in the video. When ***person2*** enters the scene, the two objects already in the scene participate in meetings, and it is not possible to identify with which of them it meets with. However, as we know these objects are together, we can conclude that ***person0***, ***person1*** and ***person2*** participate in this meeting. When ***person3*** enters the scene however, although one notices that it meets one/some objects, nothing can be inferred about which ones it exactly meets with, because at this time multiple separate meetings involving all the objects in scene may be occurring. The layout also shows that ***person0*** and ***person1*** spend most of their scene time participating in meetings, while ***person3*** is the one with less meeting moments. The layout offers this way a quick guidance to the surveillance agent about which objects may be more interesting to track during the analysis.

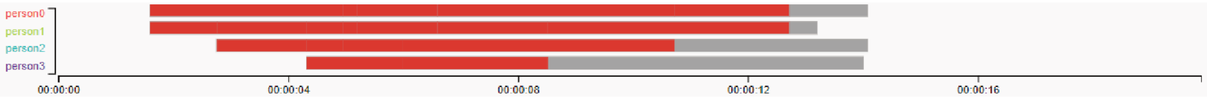
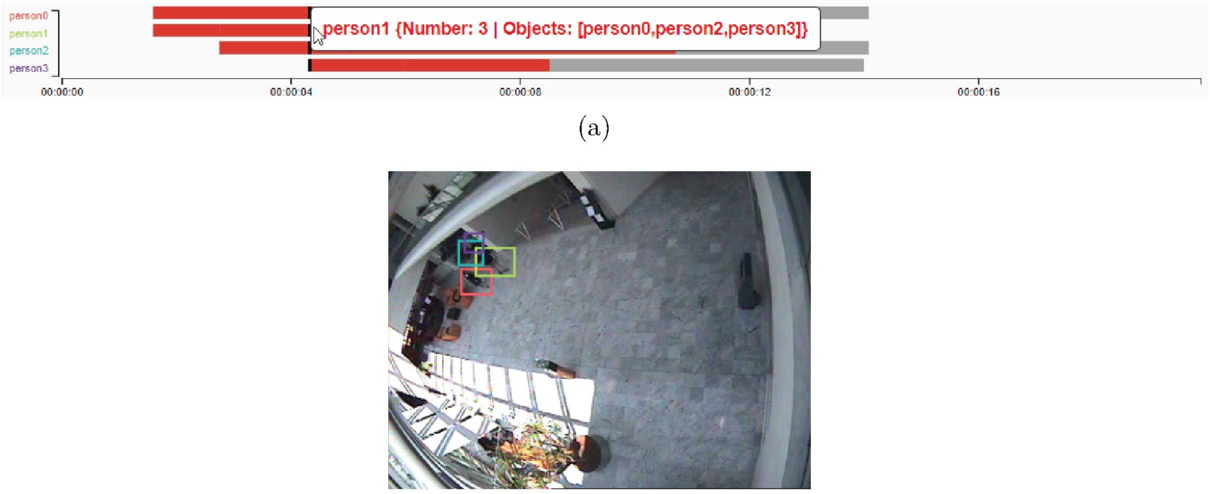


Figure 27 – *Show All Meetings* functionality applied to **MeetCrowd** video, highlighting all the moments in which objects participated in meetings in the *Appearance Bars View*.

By hovering the mouse over ***person1*** bar, it is possible to identify which moment it is meeting with all objects on scene (**TR2**). Figure 28(a) highlights the instant when this meeting starts. The black mark at the beginning of ***person3*** bar indicates ***person1*** meets it as soon as it enters the scene. The frame corresponding to this instant is shown in Figure 28(b) (**SR1**, **SR2**). The frame shows the ***person3*** (purple) bounding box intersecting with ***person1*** (green) and ***person2*** (blue) bounding boxes, and it is possible to notice that ***person1*** bounding box intersects with all other ones in the scene, which suggests that all objects are meeting at this moment.



(a)

(b)

Figure 28 – Highlight of the instant in which ***person1*** starts meeting all the objects in the scene. (a) The black marks indicate that at the selected instant all the objects met ***person1***; (b) a frame corresponding to the selected instant is also the instant when ***person3*** enters the scene.

When filtering ***person1*** meetings in *Appearance Bars View*, it is possible to notice all the moments in which ***person1*** participates in a meeting, as well as all the moments when the other objects met ***person1***, as shown in Figure 29 (**TR2**). It is possible to identify the moment in which ***person1*** meets the other three objects, depicted by the highest color intensity in the ***person1*** bar and all the other bars are dark gray in this moment, that occurs between 4.32 and 4.96 seconds. The *Appearance Bars View* shows that ***person1*** enters the scene participating in a meeting with ***person0***, then it meets ***person2*** and later ***person3***. When ***person1*** meets ***person3*** for the first time, it is already participating in a meeting with all the other identified objects. ***Person1*** stops the meeting with ***person2***. Although it is possible to notice, by watching the video, that ***person0*** and ***person1*** walked together during all their presence in scene, the *Appearance Bars View* shows some portions in the bar in which there is no meeting between these objects. These "gaps" can be produced by the method used for object tracking or even when the objects distance slightly increases for a moment, which impacts the bounding boxes generation and consequently in the meetings definition.

Although it is possible to notice by watching the video, that all the four people walked

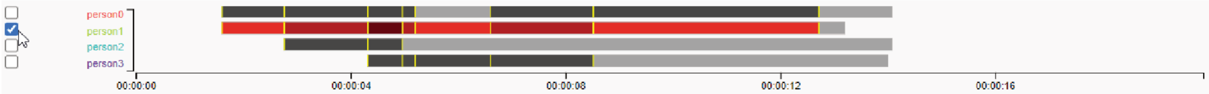


Figure 29 – ***Person1*** meetings in *Appearance Bars View*, highlighting the time portions which correspond to each meeting, as well as the distribution of meetings over time (red intensities). The layout shows that ***person1*** meets all the objects and in a specific moment it meets all of them at the same time, in this case depicted by the highest color intensity in ***person1*** bar and all the other bars are dark gray in this moment.

together during most of the moments in the video, Figure 30 shows that no simultaneous meeting between the four objects is observed. The reason is that there are no moments in which all bounding boxes intersect with each other simultaneously. However, it is possible to notice that each bounding box intersect at most two other bounding boxes simultaneously, and the meeting involving all the objects is thus indirectly created.

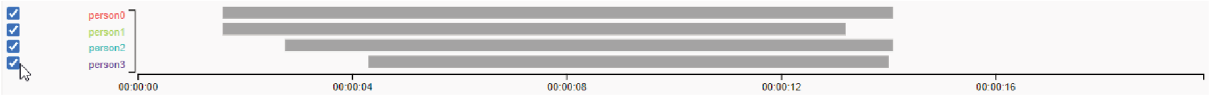


Figure 30 – Selection of all identified objects, highlighting that no simultaneous meetings occurs.

Figure 31 shows the objects average speed when they are moving in the scene, considering intervals of 1s (**TR3**). The blue shade variation in the bars segments suggests an acceleration in the objects speed in most of their presence in the video, and a small deceleration before they leave the scene. In general, no sudden speed variation is observed, and the increasing in the objects speed is roughly homogeneous, indicating a group walking pattern (Event 4). The exception is ***person3***, which presents a movement pattern that diverges from the others, specially after the 10s instant, as highlighted by a black vertical line in Figure 31. When watching the video from this moment, one notices that ***person3*** went to the same direction of the other people to leave the scene, but he/she was a little far from the group and made a bigger curve, then increasing its speed in order to come closer to the group again. The layout allows to identify this way some object behaviors, as people and vehicles running, sudden stops, parked vehicles or people stopped for a long time, for example. Observing these behavior types can help the surveillance agent to identify interesting objects to track according to the analysis context.

Figure 32 shows the selection of a region in the *Brush View* and the result in the *Appearance Bars View*. One can notice that only ***person3*** passes through the selected region. The bar portion in dark gray, which coincides with the last portion of ***person3*** bar, suggests that it is inside this region alone, when leaving the scene (**SR3**). In Fig-

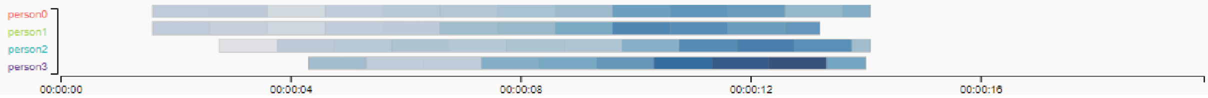
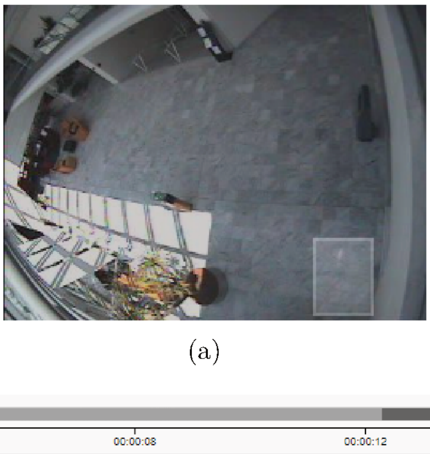


Figure 31 – Objects average speed layout for **MeetCrowd** video, highlighting objects speed variations. It is possible to notice how ***person3*** increases its speed during its presence.

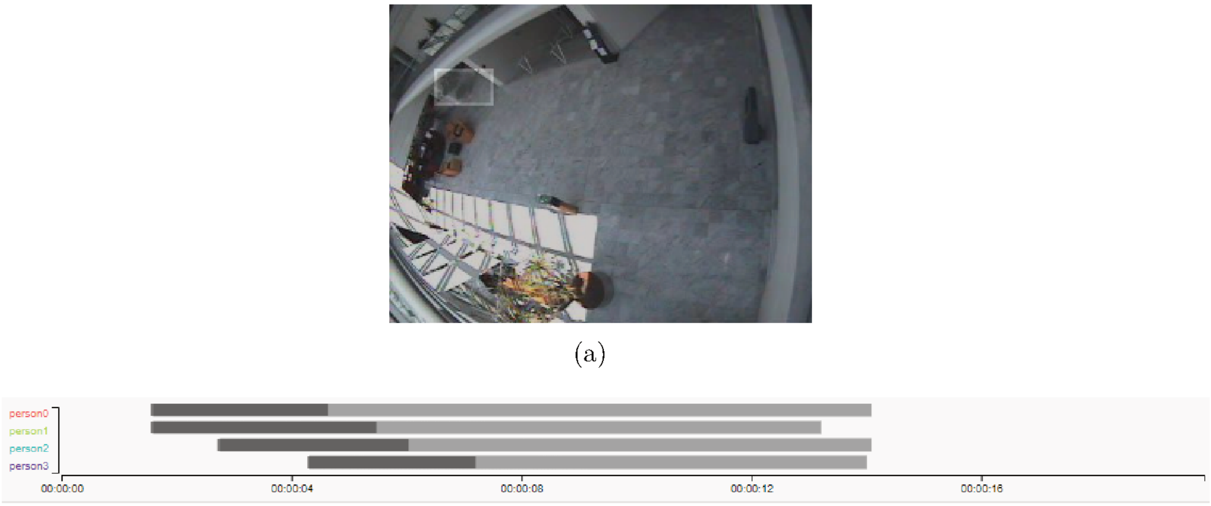
ure 33, a potential region to be used as scene entrance/exit is selected. When filtering objects by this region, one notices that all objects in the video occupy this space at their trajectories beginnings practically at the same time (see Figure 33(b)), which suggests that all objects enter the scene by crossing this region (Events 2 and 3) (**SR3**). Another potential region to be used as entrance/exit is selected in Figure 34(a). The result of filtering objects by this region (see Figure 34(b)) shows that all the objects occupy this region at the end of their presence practically at the same time, indicating they use the region to exit the scene together (Event 5). This functionality is useful for monitoring strategic regions defined by the surveillance agent, since it highlights the objects and moments in which they occupied a region of interest. The brush can be used to monitor static objects such as store cashs, ATMs, safe boxes and cabinets, as well as places where permanence is forbidden, risk areas and entrances/exits, for example. The layout is capable to quickly highlight identified objects crossing a scene region, no matter how fast they are, which allows the surveillance agent to notice important quick events that could be missed just by watching the video.



(a)

(b)

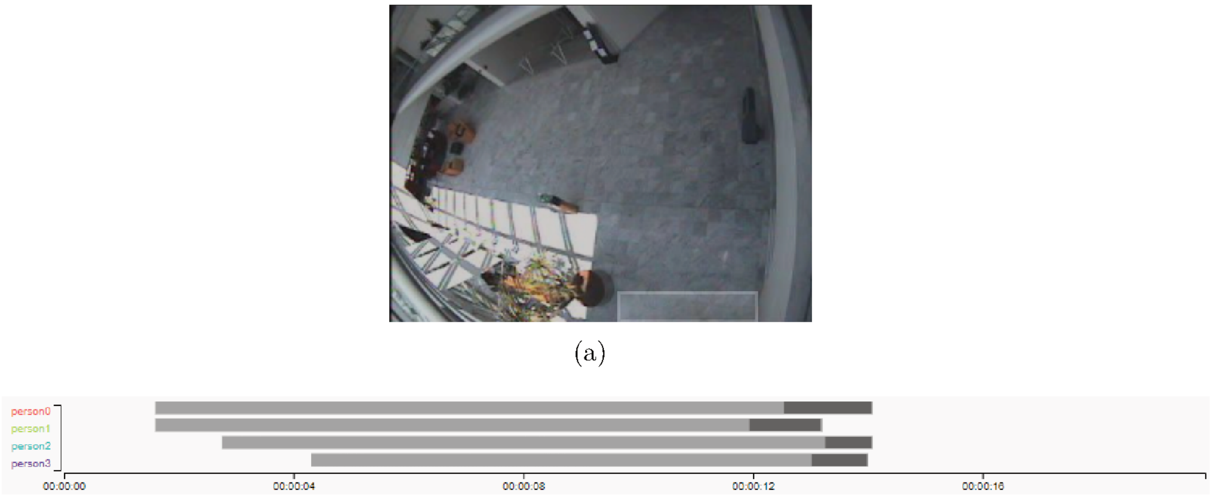
Figure 32 – Selection of a region of interest in **MeetCrowd** video. (a) selection of a region of interest in the *Brush View*; (b) brushing in *Appearance Bars View*.



(a)

(b)

Figure 33 – Selection of a potential region of interest to be used as entrance/exit in **MeetCrowd** video. (a) selection of a potential region to be used as scene entrance/exit; (b) brushing in *Appearance Bars View*.



(a)

(b)

Figure 34 – Selection of a potential region of interest to be used as entrance/exit in **MeetCrowd** video. (a) selection of a potential region to be used as scene entrance/exit; (b) brushing in *Appearance Bars View*.

4.2 ParkingLot1

VIRAT[2] (OH et al., 2011) is a repository consisting of videos representing several useful events for surveillance tasks, such as people walking, running, standing, carrying, getting into or getting out of vehicle, among others. The dataset videos present several scenarios and we use two videos from this repository representing two different parking lots. The

[2] https://viratdata.org/

repository provides, for each video, an annotation file, depicting the bounding boxes of a set of identified objects, for each frame. In order to enhance these annotations, and reflect a scenario in which a highly accurate object detector was employed, we decided to provide an extra manual annotation, resulting in the identification of additional objects, and the adjustment of some inaccurate identifications. The first video is *VIRAT_S_000002.mp4*, which we name here as **ParkingLot1**. In the **ParkingLot1** video some people walk, talk, separate/approximate to others, a car parks, a person gets out of the car and a person removes a box from the trunk.

4.2.1 Overview

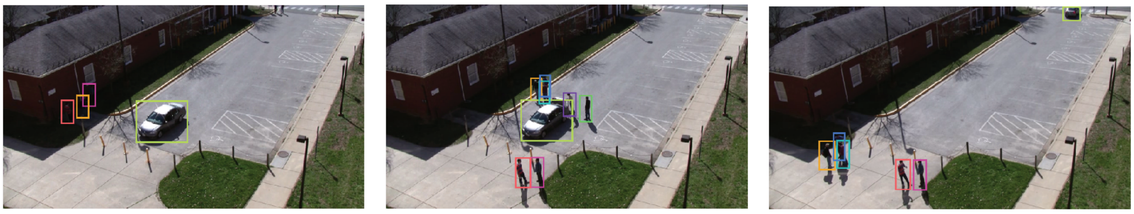
The video was recorded in a parking lot in USA, and is composed of 9075 frames and a frame rate of 29.97 frames per second, resulting in a 5 minutes and 2 seconds video. We manually defined the main events occurring in the **ParkingLot1** video and we describe them in Table 2. Figure 35 shows three key frames from **ParkingLot1** video, representing some of the events described on Table 2. Although the video scene is a parking lot, only one car appears on the scene. The rest of the identified objects are people and objects.

Table 2 – Description of the main events occurring in **ParkingLot1** video.

Event	**Frame Interval**	**Description**
1	0-2457	A group of three people (**group1**) walks through the parking lot and stops near a facility.
2	2067-4735	A car enters the scene and parks close to group1.
3	2397-9074	Another group of two people (**group2**) enters the parking lot by the upper part of the scene.
4	2457-3265	A person from group1 gesture to the group2.
5	2517-3655	The driver gets out the car and walks around it.
6	3266-9075	A person leaves group1 and join group2. Both groups walk to different positions and stop at the bottom of the scene.
7	3386-4885	A person with a hand truck dolly enters the parking lot by the upper right part of the scene, walks to the car, get a box from the car trunk, and leaves the parking lot by the upper right part of the scene.
8	4345-5394	The driver enters the car and leaves the parking lot by the upper part of the scene.

4.2.2 Analysis

Figure 36 shows the layout produced from **ParkingLot1** video, and it was generated in 2 milliseconds (**GR2**). The *Appearance Bars View* displays 10 identified objects and their respective appearance bars, which quickly allows the identification of when each object appears and leaves the scene (**GR1**, **TR1**). It is possible to notice that there are no bars labeled as cars that occupy the entire layout timeline, which may indicates that the parking spaces are empty for at least one moment in the video duration. Objects



(a) (b) (c)

Figure 35 – Examples of key frames corresponding to events in **PakingLot1** video. (a) Event 2; (b) Events 6 and 7; (c) Event 8.

labeled as people occupy the scene in most of the time, including three objects that occupy the entire layout timeline. One can thus conclude that there are always at least three objects in the parking lot. It is also possible to notice the moment when most objects occupy the timeline at the same time, indicating the moment of greatest movement in the parking lot. ***Person0***, ***person1*** and ***person2*** are in scene in the entire video duration. ***Person5*** and ***person6***, on the other hand, enter the scene after 1 minute and 28 seconds and remain until the end of the video. Finally, ***car3***, ***person4***, ***object7***, ***person8*** and ***object9*** remain in the scene for only a period of the video duration.

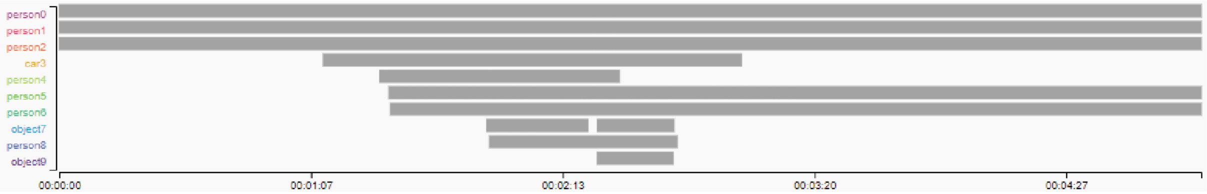


Figure 36 – **ParkingLot1** *Appearance Bars View* layout.

Figure 37 shows the result of highlighting all meetings in *Appearance Bars View.* It is possible to notice that all objects met another object in at least one moment of the video (**TR2**). ***Person0***, ***person1*** and ***person3*** start the video participating in a meeting and as they are the only objects in scene, one can conclude they are meeting each other. ***Person6*** and ***object9*** participate in meetings during all their presence in scene, while ***person2***, ***person5***, ***object7*** and ***person8*** participate in meetings during most of their presence time. The *Time in Meetings* sorting reveals that the objects that spend more time in meetings are ***person2***,***person6*** and ***person5***, respectively (see Figure 38).

Object7 is the only object with a gap on its appearance bar and it participates in meeting during practically all its presence. Thus, it can be interesting to investigate the relationships involving this object in order to further analyze its behavior. Figure 39 highlights ***object7*** meetings and it is possible to notice that it starts its presence participating in a meeting with ***person8*** and this meeting occurs during practically all their presence, despite the gap on ***object7*** appearance bar, indicating they were together. By watching

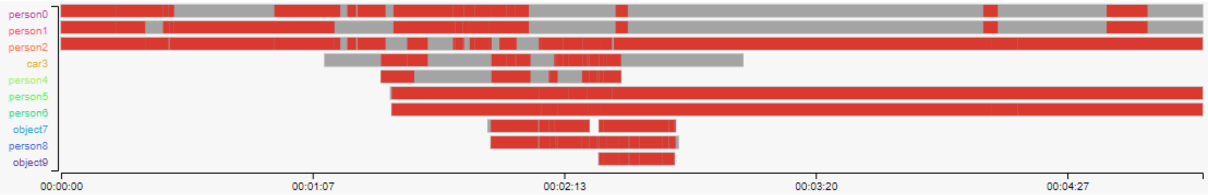


Figure 37 – *Show All Meetings* functionality applied to **ParkingLot1** video, highlighting all the moments in which objects participated in meetings in the *Appearance Bars View*.

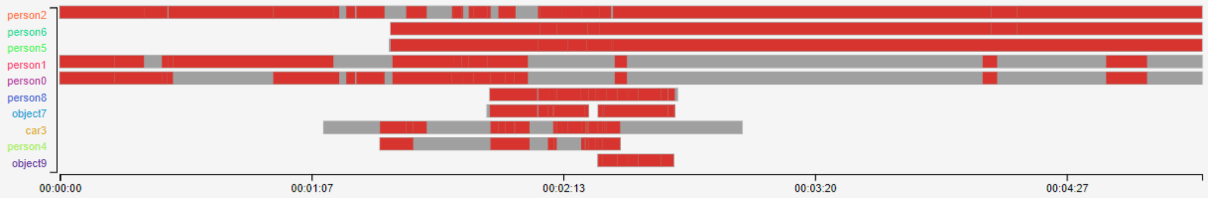


Figure 38 – *Time in Meetings* sorting highlighting all meetings in the **ParkingLot1** video.

the video, its possible to notice that ***person8*** carried a hand truck dolly (***object7***) and he/she together with ***person4*** occluded the hand truck dolly for a few seconds, causing the gap. ***Object9*** meets ***object7*** during all its presence and this meeting starts while ***object7*** finishes meeting ***car3*** and ***person4***, which suggests that ***object9*** come from these objects, and then ***person8*** walks away from ***car3*** and ***person4*** with two objects (***object7*** and ***object9***), as described in Event 7.

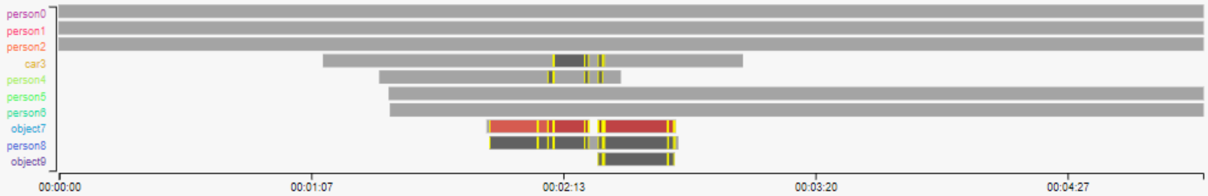


Figure 39 – ***Object7*** meetings in **ParkingLot1** *Appearance Bars View*, highlighting the time portions which correspond to each meeting, as well as the distribution of meetings over time (red intensities). The layout shows that ***person7*** meets ***person8*** during almost all its presence indicating they were together and ***object9*** meets ***object7*** during all its presence, which suggests that ***person8*** was with these two objects.

Figure 40 highlights ***person2*** meetings in the *Appearance Bars View*. One can notice that ***person2*** starts the video participating in a meeting with ***person0*** and ***person1*** only, and when it meets ***person5***, they remain together until the end of the video. ***Person2*** meets ***person6*** during a short period of time, while in meeting with ***person5*** (**TR2**). Finally, one notices that ***person2*** does not participate in meetings with ***person4*** and ***person8***, the objects labeled as person with the shortest presence in scene time. ***Person2*** participates first in group1 (Event 1) as shown in Figure 41(a) (orange bounding box), moves away to gesture to group2 (Event 4) as shown in Figure 41(b), and then participates in group2 (Event 6) as shown in Figure 41(c) (**SR2**). These events can be clearly noticed in the layout, which separates ***person2*** meetings in one initial long segment, a set of small segments in the middle, and another long segment at the end of the appearance bar. The *Appearance Bars View* shows that ***person2*** meets ***car3*** three times, twice alone and once with ***person5*** and ***person6***. In the first meeting, he/she is gesturing to group1 positioned next to the car (see Figure 41(b)) (**SR2**). Then, in the second time he/she is going to meet group1 and walks alone next to the car. Finally, in the third time, he/she, together with ***person5*** and ***person6*** walk to the bottom left part of the scene (see Figure 41(c)). The meetings between ***person2*** and ***car3*** are captured due to the minimum time threshold adopted, which defines the minimum period of time for which the bounding boxes must intersect to be considered as a meeting.

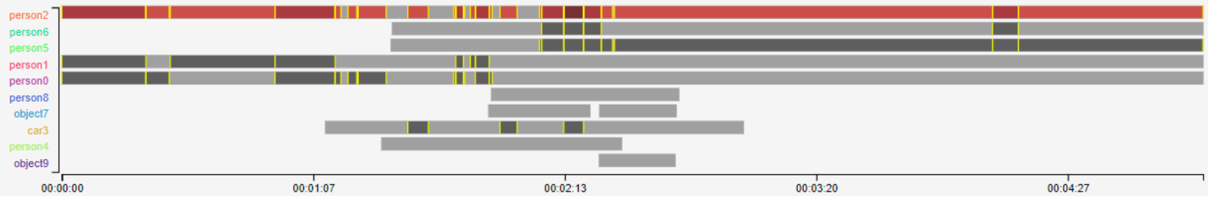


Figure 40 – ***Person2*** meetings in **ParkingLot1** *Appearance Bars View*, highlighting the time portions which correspond to each meeting, as well as the distribution of meetings over time (red intensities). The layout shows that ***person2*** first participates in meeting with ***person0*** and ***person1***, and thus participate in meeting with two another people (***person5*** and ***person6***).

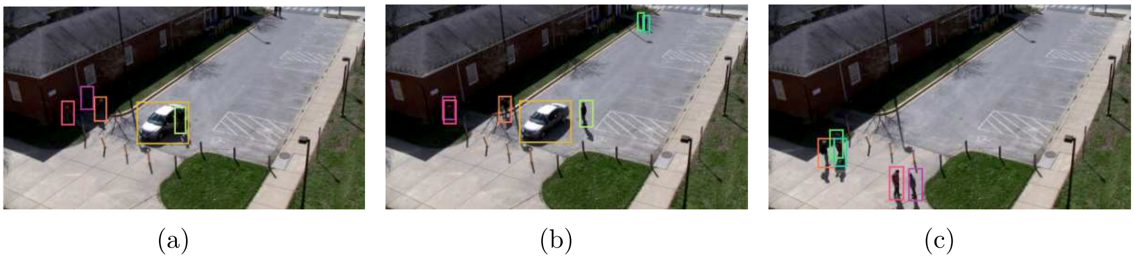

(a) (b) (c)

Figure 41 – Frames showing events involving ***person2***. (a) ***person2*** (orange bounding box) participating in group1 (Event 1); (b) ***person2*** moving away from group1 to gesture to group2 (Event 4); (c) ***person2*** participating in group2 (Event 6).

The layout in Figure 42 shows that ***person6*** enters the scene meeting ***person5*** (Event 3) and participates in this meeting during all its presence in scene, suggesting that these objects are always together. It also participates in a meeting with ***person2*** twice, and ***car3*** once, when in meeting with ***person5***. Although one notices, by watching the video, that when ***person6*** met ***person2*** they stayed together until the end of the video, this meeting is not shown in the layout. When looking at the bounding boxes during this meeting (illustrated in Figure 41(c)), it is possible to notice that they do not intersect with each other, but both ***person6*** and ***person2*** bounding boxes intersect with ***person5*** one (**SR2**). Our meeting detection procedure only calculates meetings between two objects with bounding boxes intersection, and ***person5*** represents a "connection object", which characterizes the meeting among these three objects.

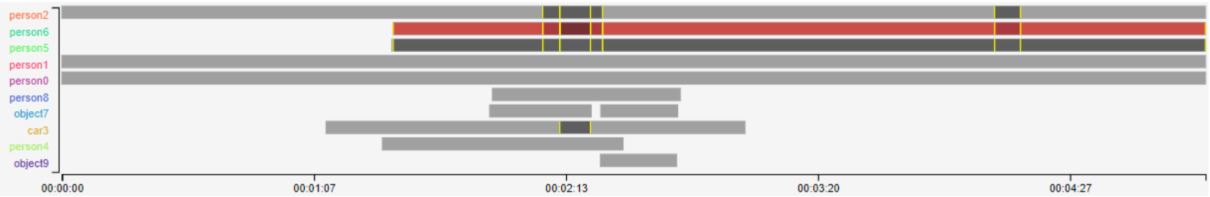


Figure 42 – ***Person6*** meetings in **ParkingLot1** *Appearance Bars View*, highlighting the time portions which correspond to each meeting, as well as the distribution of meetings over time (red intensities). The layout shows that ***person6*** meets ***person5*** during all its presence in scene, indicating they were together.

The layout highlights multiple interactions of several objects with a single object (***car3***), as shown in Figure 43 in which the *Distinct Objects Meet* sorting is applied. It is possible to notice that ***car3*** is the object that meets more distinct objects during the video. The sorting tools provide ways to facilitate the analysis of strategic situations such as vehicle theft or suspicious meetings among people. ***Person4*** starts and finishes its presence participating in a meeting with ***car3***, which suggests that he/she was in the car and leaves it (Event 5) and then gets in the car again (Event 8). There is a moment in which ***car3*** meets most of the objects at the same time (***person2***, ***object7***, ***person8***, ***person5*** and ***person6***), which is highlighted in the layout by the darkest red portion of ***car3*** appearance bar. In Figure 44 it is also possible to notice a fifth object (***person4*** - light green bounding box), close to ***car3***, that seems to participate in the meeting, but due to its position in the scene, was not considered by the layout (**TR2**, **SR2**). The system allows the user to define what can be considered as a meeting, by setting a threshold which defines the minimum time for which the objects bounding boxes must intersect, allowing several degrees of analysis.

Figure 45(a) shows the selection of a region in the *Brush View* that can be a potential parking region and the result in the *Appearance Bars View* is shown in Figure 45(b). One notices that almost all the identified objects cross the region at some moment. It is also possible to notice that ***car3*** is positioned in the selected region in most of its presence, which suggests that it is parked there (Event 2). ***Person4*** occupies the region

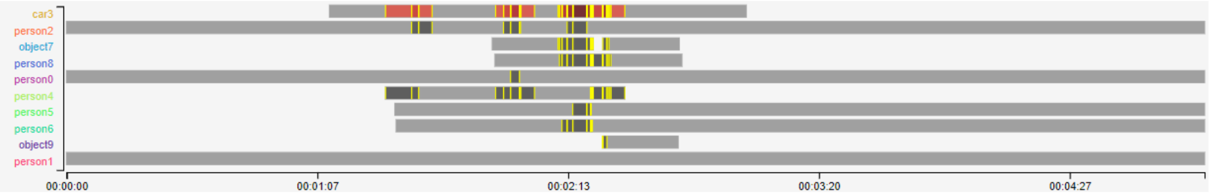


Figure 43 – ***Car3*** meetings in **ParkingLot1** *Appearance Bars View*, highlighting the time portions which correspond to each meeting, as well as the distribution of meetings over time (red intensities). The layout shows that ***car3*** is the object that meets more distinct objects during the video and ***person4*** was in the car since he/she starts and finishes his/her presence meeting the car.

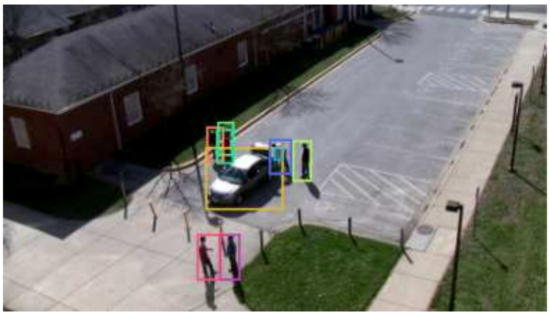
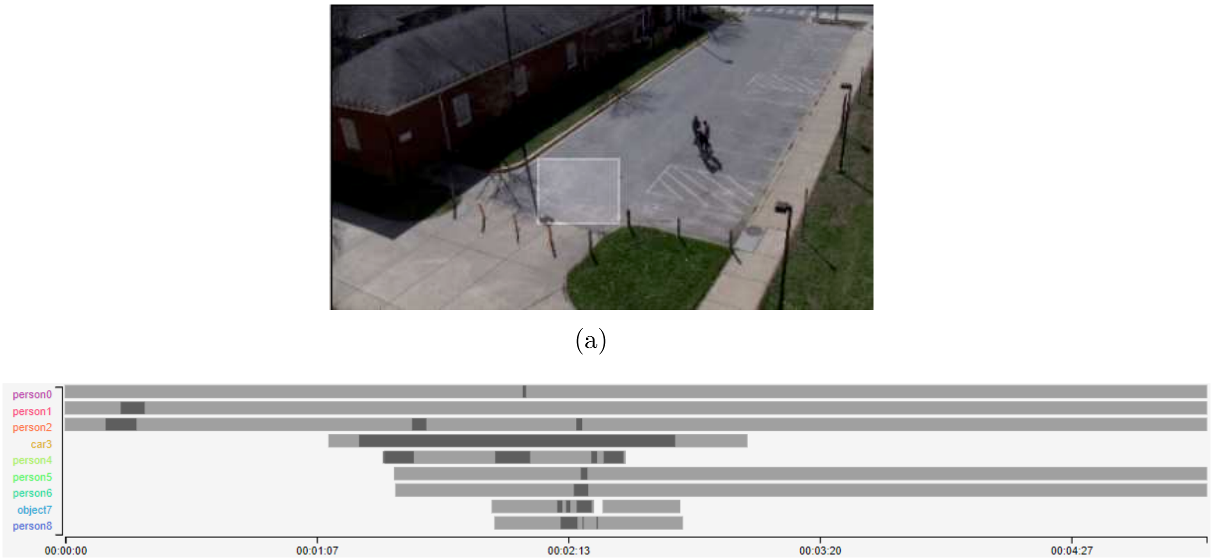


Figure 44 – A frame from the moment in which ***car3*** meets most of the objects at the same time.

at the beginning/end of its bar, indicating it entered/left the scene by this region. On the other hand, the layout shows that ***person0***, ***person1***, ***person2***, ***person5***, ***person6***, ***object7*** and ***person8*** cross the region at certain moments (**SR3**).

In Figure 46(a), a potential region to be used as scene entrance/exit was selected. When filtering objects by this region, it is possible to notice three objects (***car3***, ***person5*** and ***person6***) occupying this space at their trajectories beginnings (see Figure 46(b)), which suggests that all these objects entered the scene by crossing this region (Events 2 and 3). ***Car3*** also cross this region at the end of its appearance bar, indicating that it leaves the scene by this region (Event 8).

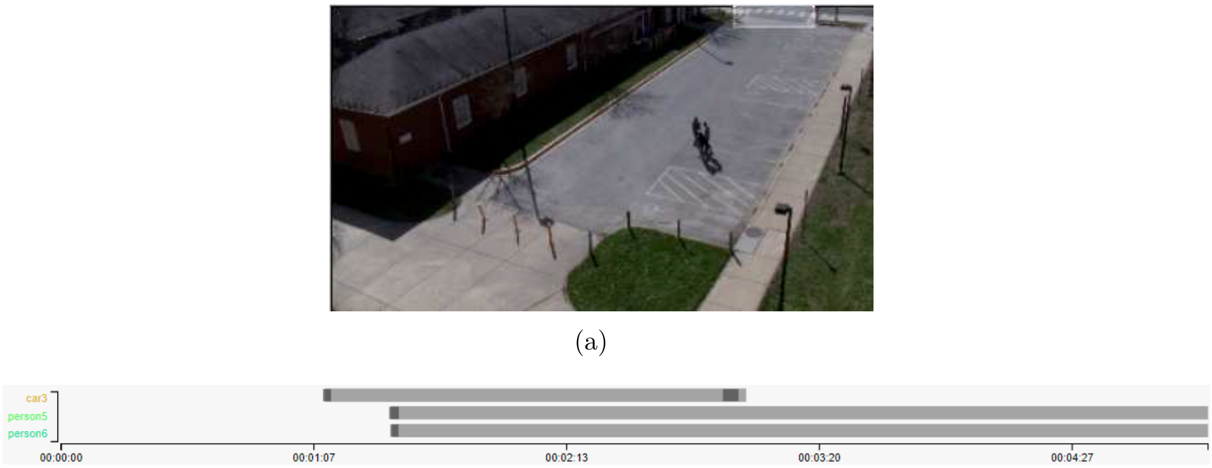
Figure 47 shows the objects average speed when they are moving in the scene, considering intervals of 1s. The blue shade variation in the bars segments suggests a low acceleration/deceleration in the objects speed in most of their presence in the video, only ***car3*** shows two acceleration peaks (**TR3**). By watching the video, one notices that the two acceleration peaks consists of the moments in which the car moved to enter and leave the parking lot, and the nature of a car movement explains these peaks. ***Person8*** shows a movement pattern similar to ***car3***: an acceleration at the beginning of the bar, a deceleration in the middle and an acceleration in the end followed by a deceleration. When watching the video, one notices that ***car3*** and ***person8*** behave similar entering



(a)

(b)

Figure 45 – Selection of a potential parking region in **ParkingLot1** video. (a) region of interest selected in *Brush View*; (b) result in *Appearance Bars View*.



(a)

(b)

Figure 46 – Selection of a potential scene entrance/exit region in **ParkingLot1** video. (a) region of interest selected in *Brush View*; (b) result in *Appearance Bars View*.

the scene moving in direction to some scene position, stopping for a while and then moving out the scene. At the beginning of their bars, ***person0***, ***person1*** and ***person2*** present an homogeneous speed decreasing, and these objects are involved in a meeting at this moment (see Figure 40), thus their speed can indicates a group movement pattern (Event 1). By watching the video, it is also possible to notice ***person0***, ***person1*** and ***person2*** walked together and then stopped near a facility (see Figure 41(a)), justifying the observed patterns.

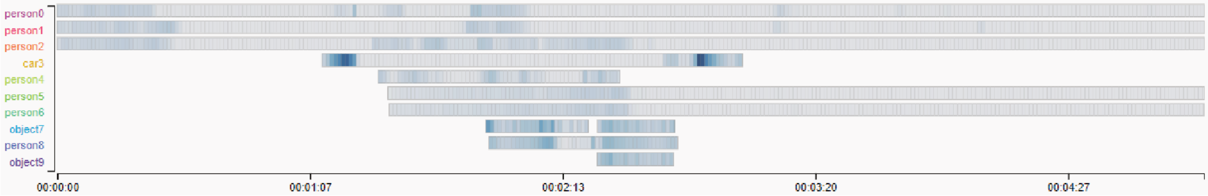


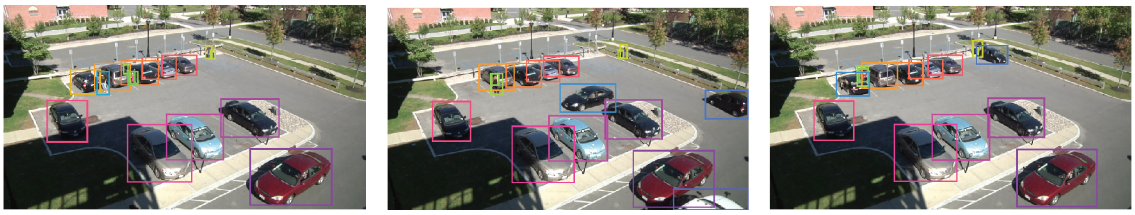
Figure 47 – Objects average speed in **ParkingLot1** video, showing that objects present a low acceleration/deceleration in most of their presence in scene, except for ***car3***, which shows an acceleration peak.

4.3 ParkingLot2

The second video from VIRAT dataset is the *VIRAT_S_040103_00_000000_000120.mp4*, which we name here as **ParkingLot2**. In the **ParkingLot2** video, cars move into the parking lot, park, enter/leave the parking lot, people move around, get in/out of cars, carry objects and load/unload objects from cars.

4.3.1 Overview

The video was recorded in a parking lot in USA, and is composed of 3594 frames and a frame rate of 29.94 frames per second, resulting in a 2 minutes video. Table 3 shows a description of the main events occurring in **ParkingLot2** video. Figure 48 shows three key frames from **ParkingLot1** video, representing some of the events described on Table 3. We manually adjusted the detections of this video in order to correct some detection failures and correctly represent the objects trajectories on the scene. Most of the objects identified in the scene are cars, but many people are also identified.



(a) (b) (c)

Figure 48 – Examples of key frames corresponding to events in **PakingLot2** video. (a) Event 2; (b) Event 6; (c) Events 7 and 8.

4.3.2 Analysis

Figure 49 shows the layout initial view for the **ParkingLot2** video, and it was generated in 3 milliseconds (**GR2**). It is possible to notice that most objects in the scene

Table 3 – Description of the main events occurring in **ParkingLot2** video.

Event	Frame Interval	Description
1	0-2064	A person (**man1**) crosses the parking lot carrying a luggage and stops in the upper part of the scene with the luggage.
2	179-508	A person (**man2**) carrying a bag leaves a car and walks out the parking lot, leaving the scene by the left part of the scene.
3	419-1047	The car from which man2 left leaves the parking lot by the right part of the scene.
4	1138-1467	One child enters a SUV car by the right side.
5	1468-1946	Woman carries another child to the left part of the SUV car.
6	1796-2395	A sedan and a hatch car enter the parking lot simultaneously, and park into two different places.
7	2065-3353	Man1 puts the luggage into the hatch car, enters it, and the hatch car leaves the parking lot by the bottom right part of the scene.
8	2186-2934	Woman, carrying the child, transport objects from the sedan car to the SUV car, and enters it.
9	2664-3594	A person (**man3**) leaves the sedan car, picks a child seat in the backseat, and meets the person in the SUV car, holding the child seat.

were labeled as cars and only bars corresponding to objects identified with this label occupy the entire layout timeline, which suggests that these objects are cars parked in the parking lot. The *Appearance Bars View* shows 26 identified objects, and it is possible to notice that 9 objects labeled as car remain on the scene during all the video duration, which leads to the conclusion that at least 9 objects are always in the parking lot. Some objects remains in scene for a short period of time: ***person15***, ***person16***, ***person17***, ***person18*** and ***car23*** (**GR1**, **TR1**). This significant difference among the presence is an aspect that may deserve a deeper investigation. In this case, the scene is a parking lot and there are cars parked, and some objects quickly cross peripheral regions of the scene, causing this significant difference.

Figure 50 shows the objects average speed layout, considering intervals of 1s. The absence of blue shade variation in the first nine objects bars segments suggests no acceleration/deceleration in the objects speed during all the video duration. Thus one can conclude that these objects do not move on the scene. By watching the video it is possible to notice that these objects are cars that remained parked in the parking lot. Three objects labeled as cars (***car9***, ***car21*** and ***car22***) present acceleration peaks, at the beginning and/or end of their bars. The peaks represent the moments when these cars enter/leave the scene. The rest of these bars show segments with a low acceleration/deceleration, which suggests that they are entering/exiting or looking for a parking space, and also show segments with no acceleration/deceleration, which suggest that the cars are parked at these moments. Figure 51 highlights the bars segments representing acceleration peaks and low/no acceleration/deceleration. ***Car9*** starts the video parked and then leaves the

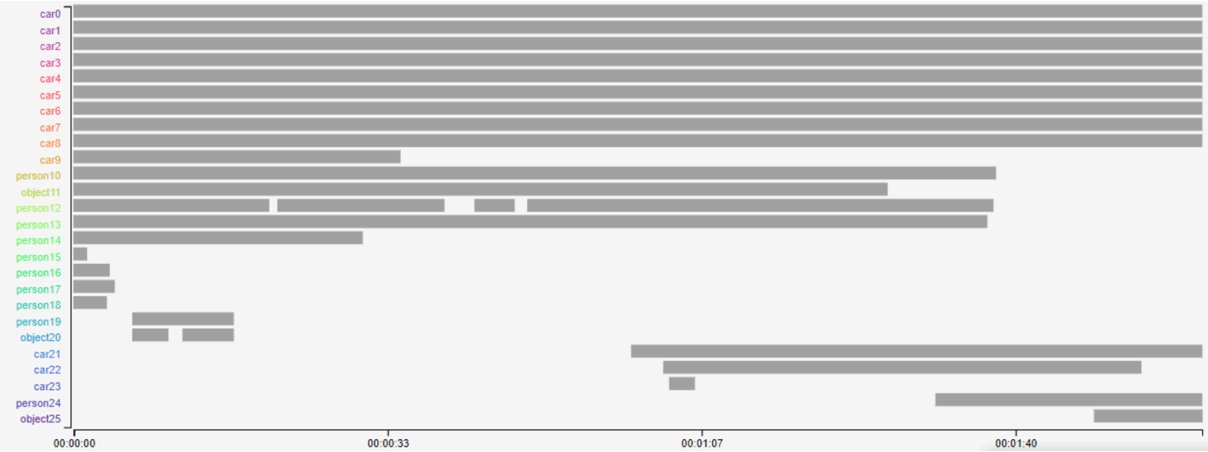


Figure 49 – **ParkingLot2** *Appearance Bars View* layout.

scene (Event 3). ***Car21*** and ***car22*** enter the scene almost simultaneously showing acceleration peaks, which suggests they are the sedan and hatch car involved in Event 6. Thus, it is possible to notice when the cars remain parked and which ones move to enter/leave the scene during video duration. Slower movements indicate that a car is looking for a parking space or parking, which is normal behavior for the location. In this layout, fast movement patterns would be abnormal, which could be an escape or a theft, for example, and the layout is able to quickly highlight those patterns (**TR3**).

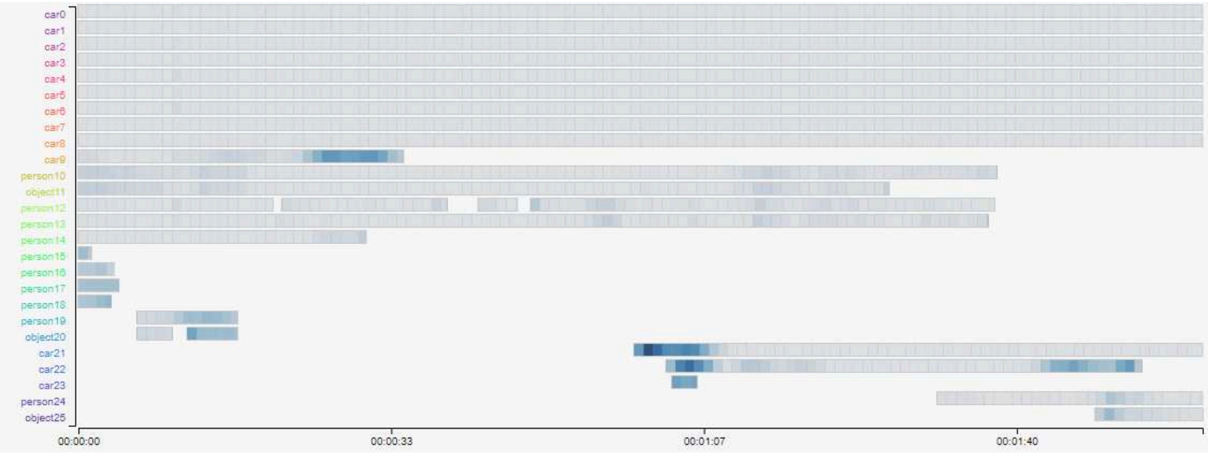
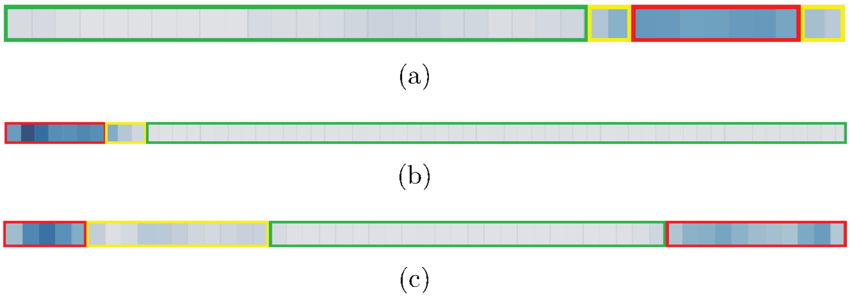


Figure 50 – Objects average speed in **ParkingLot2** video, showing that objects present a low acceleration/deceleration in most of their presence in scene, except for ***car9***, ***car21*** and ***car22***, which show acceleration peaks.

The result of highlighting all meetings in *Appearance Bars View* is shown in Figure 52. Several objects (21) start the video or enter the scene participating in meetings, and some of them participate in meetings during all their presence in scene. It is possible to notice that most of objects (16) participate in meetings during all their presence in scene (**TR2**). Among these objects, ***person12*** has four bar portions. Investigating the


(a)

(b)

(c)

Figure 51 – Bars segments representing acceleration peaks and low/no acceleration/deceleration. (a) ***car9*** bar; (b) ***car21*** bar; (c) ***car22*** bar. The red rectangles represent what we describe as acceleration peaks, the yellow rectangles represent moments with low acceleration/deceleration and the green rectangles represent moments without acceleration/deceleration.

frames corresponding to these gaps it is possible to notice that they occurs due to the person occlusion when she/he gets too close to the car, which is however masked by the continuity idea suggested by the layout, not hampering the analysis (**SR1**).

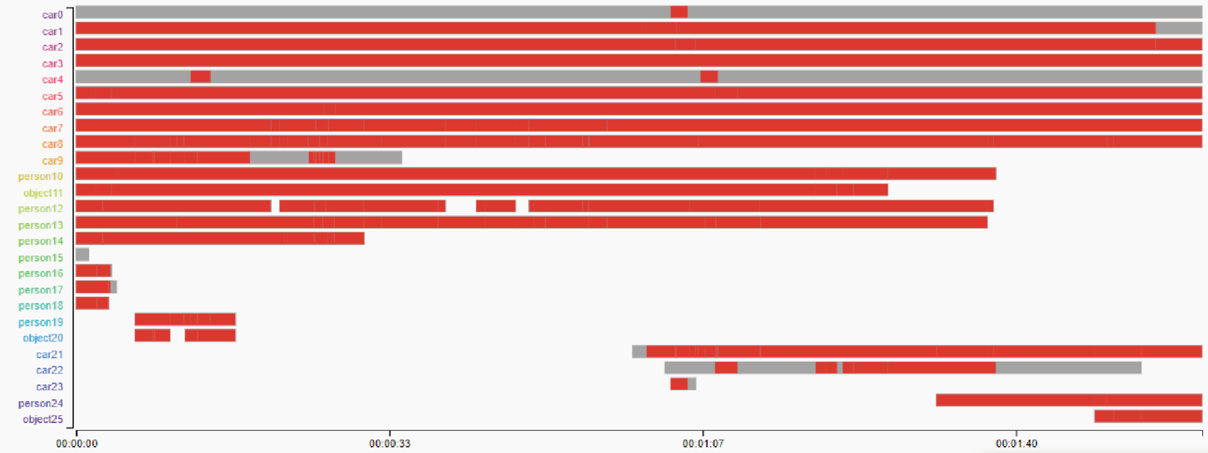


Figure 52 – *Show All Meetings* functionality applied to **ParkingLot2** video, highlighting all the moments in which objects participated in meetings in the *Appearance Bars View.*

Observing ***person10*** meetings (see Figure 53), one can notice that he/she meets ***object11*** during all its presence in scene, and ***person10*** meets ***car5*** and ***car22*** while meeting ***object11***. Observing ***person10*** and ***object11*** average speeds (see Figure 50), it is possible to notice that these two objects have a similar speed variation pattern, which reinforces the idea that they move together. ***Object11*** finishes its presence participating in a meeting with ***person10*** and ***car22***, and then ***person10*** also finishes its presence. Only ***car22*** remains a little longer in scene, representing the sequence of actions described in the Event 7. Figure 54 shows examples of frames in *Frame View* corresponding to Event 1 (Figure 54(a)) and Event 7 (Figures 54(b), 54(c) and 54(d)) highlighting the events occurrence with the white circle and confirming that ***person10***, ***object11*** and ***car22*** are involved in this events (**SR1**).

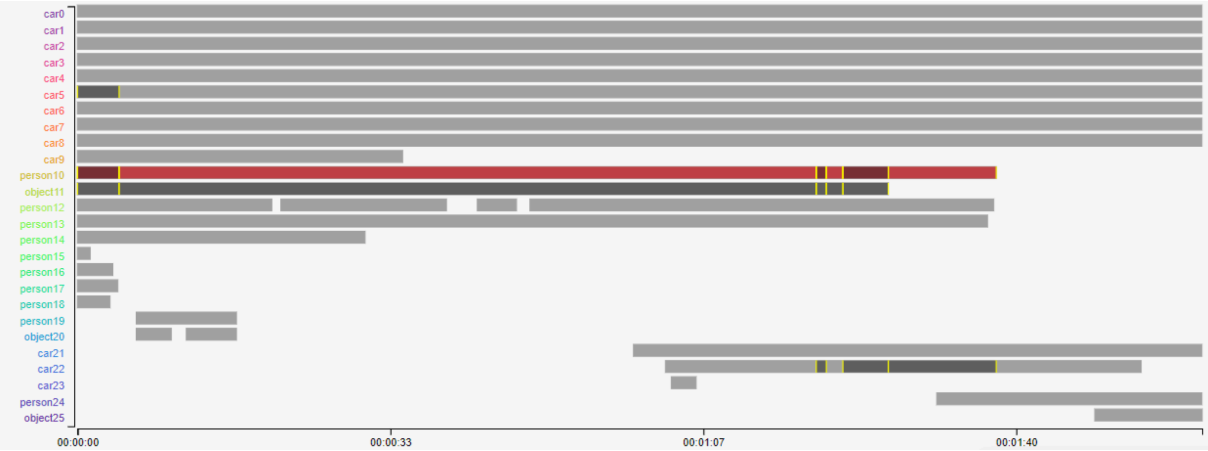


Figure 53 – ***Person10*** meetings in **ParkingLot2** *Appearance Bars View*, highlighting the time portions which correspond to each meeting, as well as the distribution of meetings over time (red intensities). The layout shows that ***person10*** meets ***object11*** during all ***object11*** presence, which suggests they were together and ***person10*** finishes its presence meeting ***car22***, indicating that ***person10*** enters the car.

The *Distinct Objects Met* sorting (see Figure 55) shows that the objects which met more distinct objects are labeled as cars. To analyze the more significant meetings involving these objects, we defined a minimum meeting time of 5 seconds. Figure 55 shows the ***car8*** meetings and it is possible to notice that it meets ***car7*** during all its presence in scene and as shown in Figure 50, they do not present speed variation, so one can infer that they are parked side by side. ***Car8*** also meets ***car21*** and ***car9*** for a while, indicating they also park next to ***car8*** for a time period. Figure 56 shows ***car7*** meeting ***car8*** and ***car6*** during all their presence in scene, indicating they are parked side by side. All the objects labeled as person which meet ***car7*** also meet ***car8*** (***person12***, ***person13*** and ***person14***), probably due to the fact they are parked together. By watching the video, one notices that these people interacted directly with ***car8***. ***Car21*** enters the scene and parks next to ***car8***, as previously concluded. Figure 57 shows that after first meeting ***car8***, ***car21*** meets ***person13*** and ***person12*** and these objects finishes their presence participating in this meeting. Figure 55 also shows these two objects finishing their presence participating in a meeting with ***car8***, which means they may have gotten into one of these cars (Event 8).

The system allows the user to define a meeting minimum time and in this case, this functionality allows the user to distinguish between significant meetings and object that just passes by another ones. Long meetings between two cars should mean they are parked nearby and long meetings between a car and a person means that the person

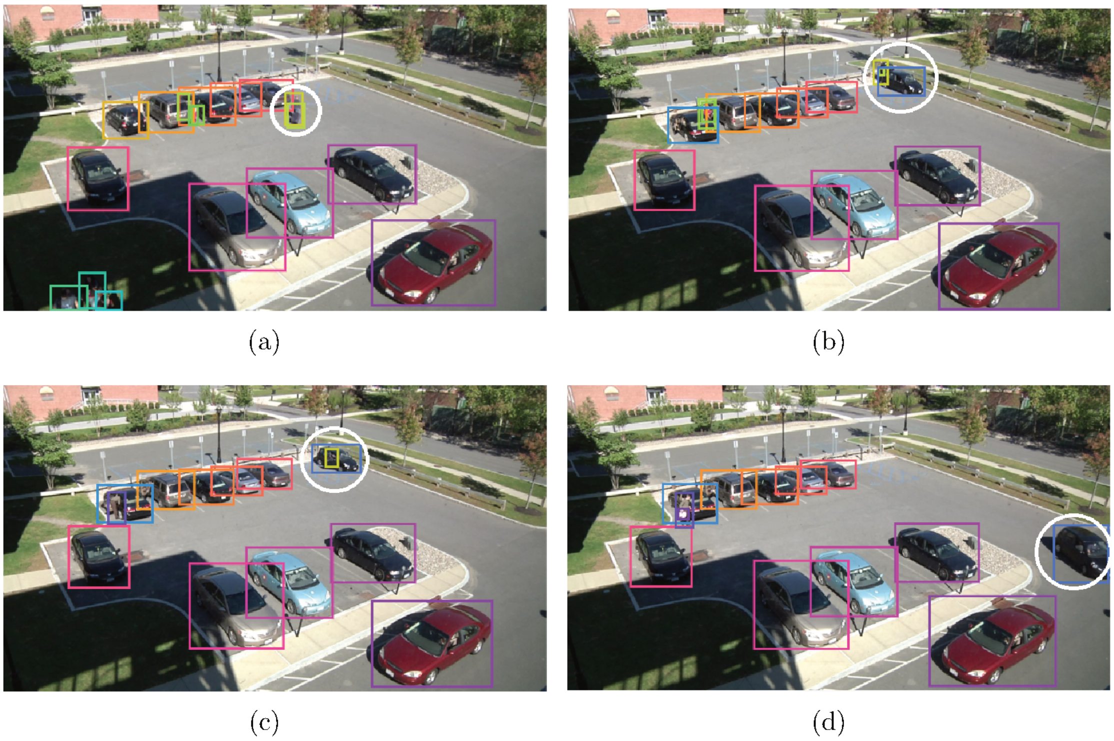


Figure 54 – Frame examples illustrating the events in **ParkingLot2** video. The white circles highlight the objects that participate in the described events. (a) shows a man carrying a luggage passing near a car (Event 1), (b) shows the man putting the luggage into the hatch car, (c) shows the man getting into the car and (d) shows the car leaving the scene (Event 7).

interacts with the car, not just passing by it. As the scene is a parking lot, it can be interesting to investigate the meetings in which these objects participate, as well as analyzing occupations in certain regions of the scene, for example, to analyze which parking spaces are most used, to investigate possible vehicle depredations, among other tasks.

Observing ***car9*** meetings (see Figure 58), considering a minimum time of 1s, it is possible to notice that ***person19*** and ***object20*** start their presence meeting ***car9*** and this meeting ends before the end of their bars, which indicates that they get out of ***car9*** and move away from it (Event 2). ***Car9*** quickly meets some objects before leaving the scene (***car8***, ***car7***, ***person13*** and ***car6***), indicating that it passes by these objects while leaving the parking lot (Event 3) (**TR2**). When exploring the video layouts, it was possible to identify this behavior pattern occurring in several videos, represented by short portions of meetings in the appearance bar almost sequentially, with little or no overlapping in the time axis. This pattern is always related to an object passing by a set of other objects, one by one. The amount of overlapping is related to the objects speed in scene. In this video, it was a car passing by other parked cars, but a person passing by other people also produced such pattern.

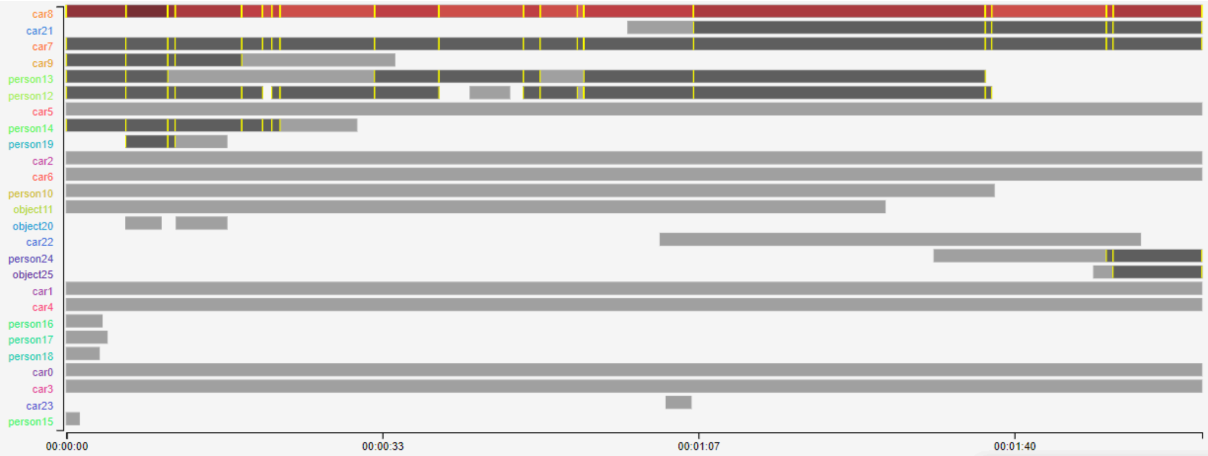


Figure 55 – ***Car8*** meetings in **ParkingLot2** *Appearance Bars View* with minimum 5 seconds of duration. The layout shows that ***car8*** meets ***car7*** during all their presence, indicating they were nearby.

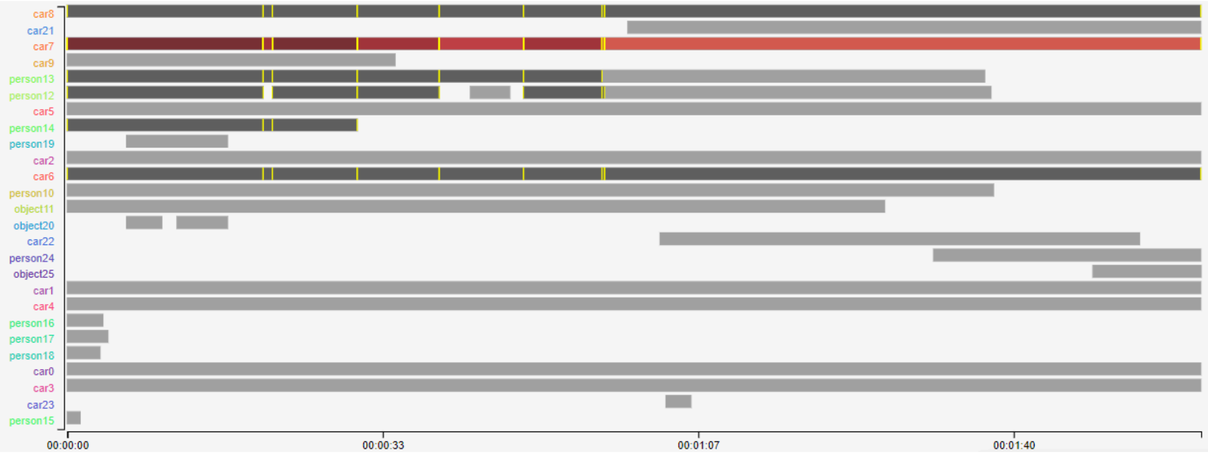


Figure 56 – ***Car7*** meetings in **ParkingLot2** *Appearance Bars View* with minimum 5 seconds of duration. The layout shows that ***car7*** meets ***car8*** and ***car6*** during all their presence, indicating they were close to ***car7***.

Person12 meets ***car8*** during almost all its presence (see Figure 59). While participating in this meeting, ***person12*** also meets another two cars (***car7*** and ***car21***), each one at different moments of its presence. Observing the *Frame View* it is possible to notice that ***car7*** is parked to the right of the ***car8*** and to the left of ***car21***. This suggests that ***person12*** was in the right side of ***car8*** and then move to the left side. ***Person12*** also participates in meeting with ***person13*** while it meets the cars, suggesting they are together (Event 5). ***Person14*** finishes its presence participating in a meeting with ***person12***, which also meets ***car7*** and ***car8*** at this moment, so we can infer that ***person14*** enters in one of these cars (Event 4) (**TR2**, **SR2**). An interesting pattern is produced when a person starts/ends his/her presence participating in a meeting with another car

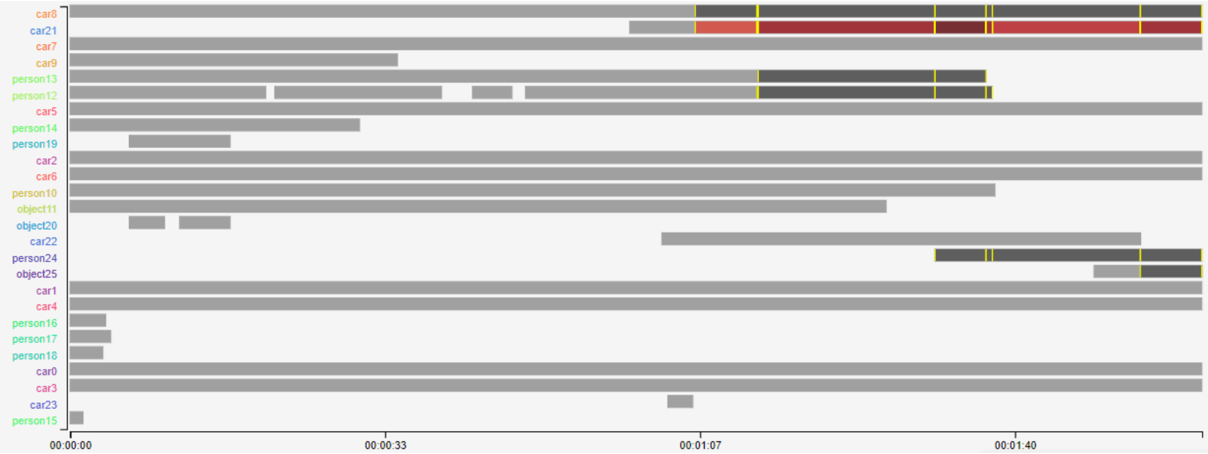


Figure 57 – ***Car21*** meetings in **ParkingLot2** *Appearance Bars View* with minimum 5 seconds of duration. The layout shows that ***car21*** meets ***car8*** during almost all its presence, indicating ***car21*** remains close to ***car8*** after entering the parking lot.

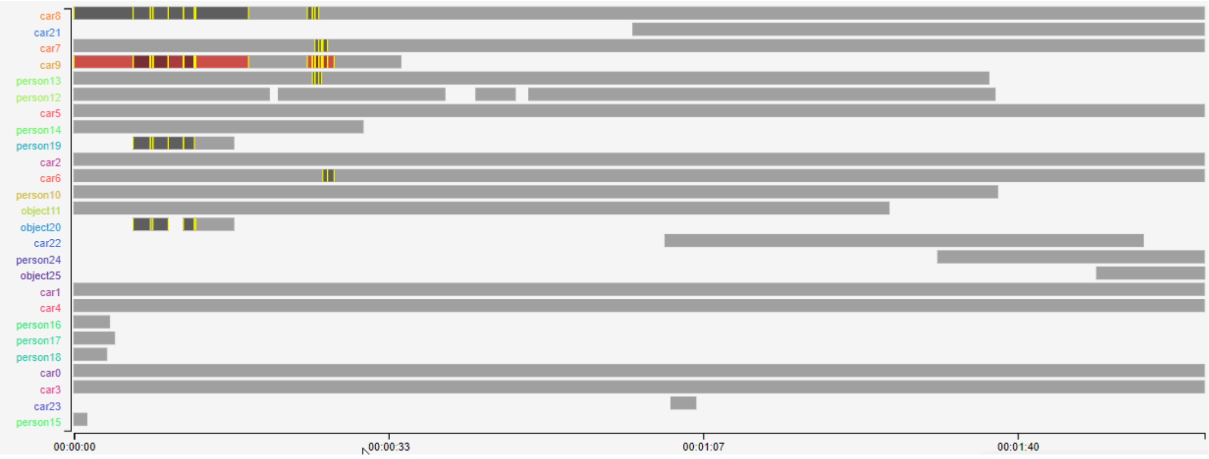


Figure 58 – ***Car9*** meetings in **ParkingLot2** *Appearance Bars View* with minimum 1 second of duration. The layout shows that ***car9*** quickly meets some objects before leaving the scene, indicating that it passes by these objects while leaving the parking lot.

already in scene. In this case, the pattern is related to a person entering in a car. The layout is able to produce behavior patterns can assist users in several analysis, improving the surveillance agent decision making.

Event 9 can be noticed in layout highlighting ***person24*** meetings (see Figure 60). One can notice that ***person24*** starts its presence participating in meeting with ***car21*** and this meeting remains until the end of the video, suggesting it was in this car and remained close to it. ***Object25*** meets ***person24*** during all its presence, and as the meeting is between an object labeled as object and another object labeled as person, it can suggests that the person was carrying the object. ***Person24*** also meets ***car8*** at the

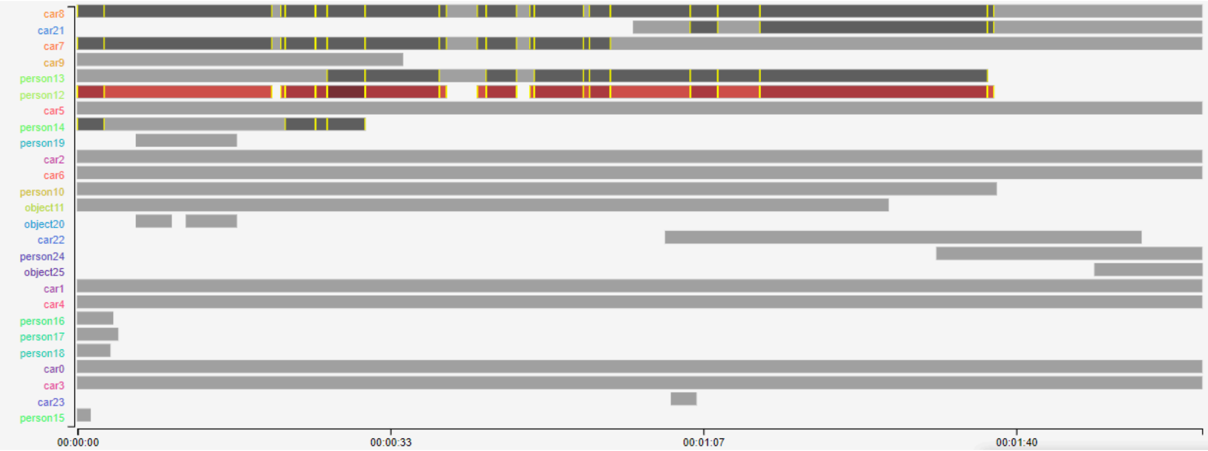


Figure 59 – ***Person12*** meetings in **ParkingLot2** *Appearance Bars View* with minimum 1 second of duration. The layout shows that ***person12*** meets ***car8*** during almost all its presence, indicating he/she was near this car, and ***person12*** finishes its presence meeting ***car8*** and ***car21***, indicating he/she enters in one of these cars.

end of its presence and after starts meeting ***object25***, suggesting ***person24*** approached ***car8*** carrying ***object25*** (TR2).

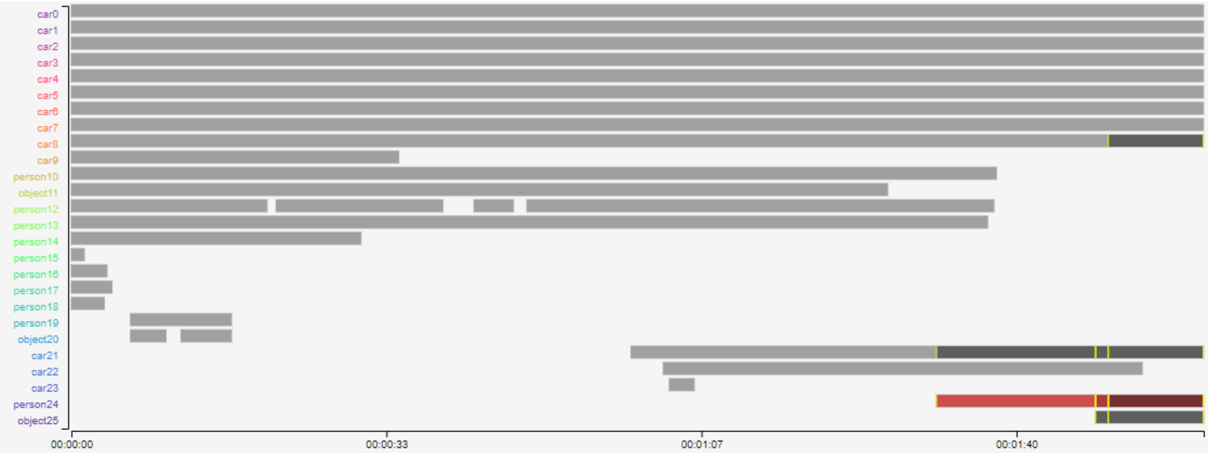


Figure 60 – ***Person24*** meetings in **ParkingLot2** *Appearance Bars View*, highlighting the time portions which correspond to each meeting, as well as the distribution of meetings over time (red intensities). The layout shows that ***person24*** starts its presence meeting ***car21***, which suggests that ***person24*** was in the car.

Chapter 5

Conclusions

In this chapter we summarize the conclusions about this Master's project, outlining our contributions, limitations and suggestions of directions for future research.

5.1 Contributions

In this work, we developed a surveillance video visual analysis methodology with focus on the exploration of objects behavior in surveillance scenarios. Our solution employs three coordinated layouts which highlight the dynamics and distribution of each object presence in scene, representing several aspects of objects behavior. A set of interaction tools provided means to explore several objects behavior aspects, specially interactions among identified objects. These tools allowed to distinguish between different types of interaction among objects, to identify the number of objects involved in a meeting, as well as in which scene regions these interactions occur. Finally, it was possible to investigate the objects average speed in the scene, allowing the identification of several patterns related to these speeds. The methodology uses the result of automatic or manual detection/tracking procedures as input, making possible the use of any technique or combination of techniques. We also developed a system implementing the proposed methodology, which is publicly available online[1].

We performed several experiments involving different surveillance scenarios in order to validate the proposal. These experiments demonstrated that the proposed methodology allows to identify instants corresponding to specific objects actions, which may be potentially related to the occurrence of relevant events. The *Appearance Bars View* was able to reveal several aspects related to each object behavior during its presence in scene, such as instants when an object moves, when it occupies a certain region of the scene, when the objects interact among themselves, which objects were involved in the interaction, and for how long these interactions occur. The layout also produced some patterns that provided us a quick identification of some types of events. For example,

[1] https://github.com/cibelemara/objects-behavior-visual-analysis-system

when an object labeled as a person begins/ends its presence meeting an object labeled as car, in most cases it represented a person entering/leaving a car. Sorting the objects appearance bars according to a specific aspect, such as time in meetings or number of distinct objects met, allowed to rank the objects by a certain analysis aspect, focusing the analysis on specific objects, which could potentially be involved in strategic events in the scene. Highlighting the objects average speed in appearance bars produced interesting movement patterns associated to a single or multiple objects. These patterns revealed common and/or abnormal behaviors related to the objects movement in the scene, and allowed the identification of patterns types related to certain objects activities, such as parked cars, cars entering/leaving the parking lot, among others. The *Brush View* allowed to explore people movement and behavior in potentially critical regions in the scene, in which relevant events may occur, such as forbidden places, entrances/exits, store cashes and ATMs, among others.

5.2 Limitations and Future Work

During the evaluation of the methodology, we were able to identify some limitations in our proposal. One of them is that is dependent of the results of a specific detection/tracking technique, and thus the ability to faithfully represent the objects behavior is totally related to the accuracy of the chosen technique. Multiple detection/tracking techniques can be combined and extra annotations can be manually added to improve the accuracy of detection/tracking.

Another limitation is the representation of a large number of objects, since it generates a large number of appearance bars to observe. Time and space filters can partially solve this problem, since they focus the analysis on a specific time period or scene region, making it possible to decrease the number of objects analyzed.

Future work related to this research include:

- **Perform a user study:** it is important to perform a user study with security agents and other surveillance experts in order to evaluate the system ability to communicate the video content, in terms of object behavior, as well as which analysis tasks these users are able to perform using our proposed layouts. It is also important to collect their feedback about the system itself, for further improvement;

- **Improve meetings detection procedure:** in our approach, the definition of a meeting is based on the bounding boxes intersection of only two objects, and identify meetings involving three or more objects, although possible, is not always straightforward. In the experiments, it was noted that some objects met/interacted but the meeting was not considered, because, although the objects interacted close to each other, their bounding boxes did not intersect. A new meetings detection approach could consider additional multiple aspects related to the object bounding boxes, such as their shape or area, as well as the object type, refining and improving the meetings detection;

- **Adapt the strategy for Real time surveillance scenarios:** our methodology was developed for post-event analysis, using previously collected surveillance footage. However, several recent CCTV systems work in real time scenarios, in which a video stream is uninterrupted captured. We believe however that the proposed methodology can be adapted to this scenario, using real time object detector/tracker techniques, in addition to modifications on the layout generation procedure.

Bibliography

AFROZ, S.; MORSHED, B. I. Web visualization of temporal and spatial health data from smartphone app in smart and connected community (scc). In: IEEE. **2018 IEEE International Smart Cities Conference (ISC2)**. [S.l.], 2018. p. 1–6. <https://doi.org/10.1109/ISC2.2018.8656990>.

AHMED, S. A. et al. Localization of region of interest in surveillance scene. **Multimedia Tools and Applications**, Springer, v. 76, n. 11, p. 13651–13680, 2017. <https://doi.org/10.1007/s11042-016-3762-y>.

_____. Trajectory-based surveillance analysis: A survey. **IEEE Transactions on Circuits and Systems for Video Technology**, IEEE, v. 29, n. 7, p. 1985–1997, 2018. <https://doi.org/10.1109/TCSVT.2018.2857489>.

ANDRIENKO, N.; ANDRIENKO, G. Visual analytics of movement: An overview of methods, tools and procedures. **Information Visualization**, Sage Publications Sage UK: London, England, v. 12, n. 1, p. 3–24, 2013. <https://doi.org/10.1007/978-3-642-37583-5>.

BABIKER, M. et al. Automated daily human activity recognition for video surveillance using neural network. In: IEEE. **2017 IEEE 4th international conference on smart instrumentation, measurement and application (ICSIMA)**. [S.l.], 2017. p. 1–5. <https://doi.org/10.1109/ICSIMA.2017.8312024>.

BACH, B. et al. A review of temporal data visualizations based on space-time cube operations. In: **Eurographics conference on visualization**. [S.l.: s.n.], 2014.

_____. A descriptive framework for temporal data visualizations based on generalized space-time cubes. In: WILEY ONLINE LIBRARY. **Computer Graphics Forum**. [S.l.], 2017. v. 36, n. 6, p. 36–61. <https://doi.org/10.1111/cgf.12804>.

_____. Time curves: Folding time to visualize patterns of temporal evolution in data. **IEEE transactions on visualization and computer graphics**, IEEE, v. 22, n. 1, p. 559–568, 2015. <https://doi.org/10.1109/TVCG.2015.2467851>.

BAGHERI, S.; ZHENG, J. Y.; SINHA, S. Temporal mapping of surveillance video for indexing and summarization. **Computer Vision and Image Understanding**, Elsevier, v. 144, p. 237–257, 2016. <https://doi.org/10.1016/j.cviu.2015.11.014>.

CARD, S. K.; MACKINLAY, J. D.; SHNEIDERMAN, B. **Readings in information visualization: using vision to think**. [S.l.]: Morgan Kaufmann Publishers Inc, 1999.

CHEN, Q. et al. An ssd algorithm based on vehicle counting method. In: IEEE. **2018 37th Chinese Control Conference (CCC)**. [S.l.], 2018. p. 7673–7677. <https://doi.org/10.23919/ChiCC.2018.8483037>.

CHENG, H.-Y.; HWANG, J.-N. Integrated video object tracking with applications in trajectory-based event detection. **Journal of Visual Communication and Image Representation**, Elsevier, v. 22, n. 7, p. 673–685, 2011. <https://doi.org/10.1016/j.jvcir.2011.07.001>.

CLAES, S. et al. Design implications of casual health visualization on tangible displays. In: **Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems**. [S.l.: s.n.], 2015. p. 1839–1844. <https://doi.org/10.1145/2702613.2732830>.

CLEVELAND, W. S. **Visualizing data**. [S.l.]: Hobart Press, 1993.

COX, M. A.; COX, T. F. Multidimensional scaling. In: **Handbook of data visualization**. [S.l.]: Springer, 2008. p. 315–347. <https://doi.org/10.1007/978-3-540-33037-0_14>.

DAO, N.-Q. et al. Management of video surveillance for smart cities. In: **Handbook of Smart Cities**. [S.l.]: Springer, 2018. p. 285–310. <https://doi.org/10.1007/978-3-319-97271-8_11>.

ELHOSENY, M. Multi-object detection and tracking (modt) machine learning model for real-time video surveillance systems. **Circuits, Systems, and Signal Processing**, Springer, v. 39, n. 2, p. 611–630, 2020. <https://doi.org/10.1007/s00034-019-01234-7>.

FERNANDEZ-CARROBLES, M. M.; DENIZ, O.; MAROTO, F. Gun and knife detection based on faster r-cnn for video surveillance. In: SPRINGER. **Iberian Conference on Pattern Recognition and Image Analysis**. [S.l.], 2019. p. 441–452. <https://doi.org/10.1007/978-3-030-31321-0_38>.

GONG, M.; SHU, Y. Real-time detection and motion recognition of human moving objects based on deep learning and multi-scale feature fusion in video. **IEEE Access**, IEEE, v. 8, p. 25811–25822, 2020. <https://doi.org/10.1109/ACCESS.2020.2971283>.

HAMPAPUR, A. et al. Smart surveillance: applications, technologies and implications. In: IEEE. **Fourth International Conference on Information, Communications and Signal Processing, 2003 and the Fourth Pacific Rim Conference on Multimedia. Proceedings of the 2003 Joint**. [S.l.], 2003. v. 2, p. 1133–1138.

HÖFERLIN, M.; HÖFERLIN, B.; WEISKOPF, D. Video visual analytics of tracked moving objects. In: **Proc. 3rd Workshop on Behaviour Monitoring and Interpretation**. [S.l.: s.n.], 2009. v. 541, p. 59–64.

HOFFMAN, P.; GRINSTEIN, G.; PINKNEY, D. Dimensional anchors: a graphic primitive for multidimensional multivariate information visualizations. In: **Proceedings of the 1999 workshop on new paradigms in information visualization and manipulation in conjunction with the eighth ACM internation conference**

on Information and knowledge management. [S.l.: s.n.], 1999. p. 9–16. <https://doi.org/10.1145/331770.331775>.

HOTELLING, H. Analysis of a complex of statistical variables into principal components. **Journal of educational psychology**, Warwick & York, v. 24, n. 6, p. 417, 1933. <https://doi.org/10.1037/h0071325>.

HU, L.; NI, Q. Iot-driven automated object detection algorithm for urban surveillance systems in smart cities. **IEEE Internet of Things Journal**, IEEE, v. 5, n. 2, p. 747–754, 2017. <https://doi.org/10.1109/JIOT.2017.2705560>.

HU, W.-C. et al. Moving object detection and tracking from video captured by moving camera. **Journal of Visual Communication and Image Representation**, Elsevier, v. 30, p. 164–180, 2015. <https://doi.org/10.1016/j.jvcir.2015.03.003>.

HUANG, T. Traffic speed estimation from surveillance video data. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops**. [S.l.: s.n.], 2018.

INSELBERG, A.; DIMSDALE, B. Parallel coordinates: a tool for visualizing multidimensional geometry. In: IEEE COMPUTER SOCIETY PRESS. **Proceedings of the 1st conference on Visualization'90**. [S.l.], 1990. p. 361–378.

JOHNSON, B. Treeviz: treemap visualization of hierarchically structured information. In: **Proceedings of the SIGCHI conference on Human factors in computing systems**. [S.l.: s.n.], 1992. p. 369–370. <https://doi.org/10.1145/142750.142833>.

JOIA, P. et al. Local affine multidimensional projection. **IEEE Transactions on Visualization and Computer Graphics**, IEEE, v. 17, n. 12, p. 2563–2571, 2011. <https://doi.org/10.1109/TVCG.2011.220>.

JOSHI, K. A.; THAKORE, D. G. A survey on moving object detection and tracking in video surveillance system. **International Journal of Soft Computing and Engineering**, Citeseer, v. 2, n. 3, p. 44–48, 2012.

JYOTHI, R. A.; BABU, K. R.; BACHU, S. Moving object detection using the genetic algorithm for real times transportation. **International Journal of Engineering and Advanced Technology**, Blue Eyes Intelligence Engineering Sciences Publication, v. 8, p. 991–996, 2019. <https://doi.org/10.35940/ijeat.F8266.088619>.

KEIM, D. A. et al. Pixel bar charts: a visualization technique for very large multiattribute data sets. **Information Visualization**, SAGE Publications Sage UK: London, England, v. 1, n. 1, p. 20–34, 2002. <https://doi.org/10.1057/palgrave.ivs.9500003>.

KIM, I. S. et al. Intelligent visual surveillance—a survey. **International Journal of Control, Automation and Systems**, Springer, v. 8, n. 5, p. 926–939, 2010. <https://doi.org/10.1007/s12555-010-0501-4>.

KIM, K.-J. et al. Performance enhancement of yolov3 by adding prediction layers with spatial pyramid pooling for vehicle detection. In: IEEE. **2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)**. [S.l.], 2018. p. 1–6. <https://doi.org/10.1109/AVSS.2018.8639438>.

KO, S. et al. Marketanalyzer: an interactive visual analytics system for analyzing competitive advantage using point of sale data. In: WILEY ONLINE LIBRARY. **Computer Graphics Forum**. [S.l.], 2012. v. 31, n. 3pt3, p. 1245–1254. <https://doi.org/10.1111/j.1467-8659.2012.03117.x>.

KO, T. A survey on behaviour analysis in video surveillance applications. **Video Surveillance**, InTech, p. 279–294, 2011. <https://doi.org/10.5772/15302>.

KRUEGLE, H. **CCTV Surveillance: Video practices and technology**. [S.l.]: Elsevier, 2011.

KUMAR, D. et al. A visual-numeric approach to clustering and anomaly detection for trajectory data. **The Visual Computer**, Springer, v. 33, n. 3, p. 265–281, 2017. <https://doi.org/10.1007/s00371-015-1192-x>.

LEE, K. B.; SHIN, H. S. An application of a deep learning algorithm for automatic detection of unexpected accidents under bad cctv monitoring conditions in tunnels. In: IEEE. **2019 International Conference on Deep Learning and Machine Learning in Emerging Applications (Deep-ML)**. [S.l.], 2019. p. 7–11. <https://doi.org/10.1109/Deep-ML.2019.00010>.

LI, C. et al. Face detection based on yolov3. In: **Recent Trends in Intelligent Computing, Communication and Devices**. [S.l.]: Springer, 2020. p. 277–284. <https://doi.org/10.1007/978-981-13-9406-5_34>.

LI, H. et al. Secure video surveillance framework in smart city. **Sensors**, Multidisciplinary Digital Publishing Institute, v. 21, n. 13, p. 4419, 2021. <https://doi.org/10.3390/s21134419>.

LI, J. et al. Facial expression recognition with faster r-cnn. **Procedia Computer Science**, Elsevier, v. 107, p. 135–140, 2017. <https://doi.org/10.1016/j.procs.2017.03.069>.

LI, T. et al. Crowded scene analysis: A survey. **IEEE transactions on circuits and systems for video technology**, IEEE, v. 25, n. 3, p. 367–386, 2014. <https://doi.org/10.1109/TCSVT.2014.2358029>.

LI, Y. et al. Efficient ssd: A real-time intrusion object detection algorithm for railway surveillance. In: IEEE. **2020 International Conference on Sensing, Diagnostics, Prognostics, and Control (SDPC)**. [S.l.], 2020. p. 391–395. <https://doi.org/10.1109/SDPC49476.2020.9353137>.

LIANG, W.; NIU, H. W. M. T. A survey of visual analysis of human motion [j]. **Chinese Journal of Computers**, v. 3, p. 225–237, 2002.

LIN, K. et al. Abandoned object detection via temporal consistency modeling and back-tracing verification for visual surveillance. **IEEE Transactions on Information Forensics and Security**, IEEE, v. 10, n. 7, p. 1359–1370, 2015. <https://doi.org/10.1109/TIFS.2015.2408263>.

LIU, W. et al. Ssd: Single shot multibox detector. In: SPRINGER. **European conference on computer vision**. [S.l.], 2016. p. 21–37. <https://doi.org/10.1007/978-3-319-46448-0_2>.

LOSADA, A. G.; THERÓN, R.; BENITO, A. Bkviz: A basketball visual analysis tool. **IEEE computer graphics and applications**, IEEE, v. 36, n. 6, p. 58–68, 2016. <https://doi.org/10.1109/MCG.2016.124>.

LUNA, E. et al. Abandoned object detection in video-surveillance: survey and comparison. **Sensors**, Multidisciplinary Digital Publishing Institute, v. 18, n. 12, p. 4290, 2018. <https://doi.org/10.3390/s18124290>.

LUO, Y. et al. Pedestrian tracking in surveillance video based on modified cnn. **Multimedia tools and applications**, Springer, v. 77, n. 18, p. 24041–24058, 2018. <https://doi.org/10.1007/s11042-018-5728-8>.

MA, J. et al. Living liquid: Design and evaluation of an exploratory visualization tool for museum visitors. **IEEE Transactions on Visualization and Computer Graphics**, IEEE, v. 18, n. 12, p. 2799–2808, 2012. <https://doi.org/10.1109/TVCG.2012.244>.

MABROUK, A. B.; ZAGROUBA, E. Abnormal behavior recognition for intelligent video surveillance systems: A review. **Expert Systems with Applications**, Elsevier, v. 91, p. 480–491, 2018. <https://doi.org/10.1016/j.eswa.2017.09.029>.

MEGHDADI, A. H.; IRANI, P. Interactive exploration of surveillance video through action shot summarization and trajectory visualization. **IEEE Transactions on Visualization and Computer Graphics**, IEEE, v. 19, n. 12, p. 2119–2128, 2013. <https://doi.org/10.1109/TVCG.2013.168>.

MENDES, G.; PAIVA, J. G. S.; SCHWARTZ, W. R. Point-placement techniques and temporal self-similarity maps for visual analysis of surveillance videos. In: IEEE. **2019 23rd International Conference Information Visualisation (IV)**. [S.l.], 2019. p. 127–132. <https://doi.org/10.1109/IV.2019.00030>.

MOLCHANOV, V. et al. Pedestrian detection in video surveillance using fully convolutional yolo neural network. In: INTERNATIONAL SOCIETY FOR OPTICS AND PHOTONICS. **Automated visual inspection and machine vision II**. [S.l.], 2017. v. 10334, p. 103340Q. <https://doi.org/10.1117/12.2270326>.

NIKOUEI, S. Y.; CHEN, Y.; FAUGHNAN, T. R. Smart surveillance as an edge service for real-time human detection and tracking. In: IEEE. **2018 IEEE/ACM Symposium on Edge Computing (SEC)**. [S.l.], 2018. p. 336–337. <https://doi.org/10.1109/SEC.2018.00036>.

OH, S. et al. A large-scale benchmark dataset for event recognition in surveillance video. In: IEEE. **CVPR 2011**. [S.l.], 2011. p. 3153–3160. <https://doi.org/10.1109/AVSS.2011.6027400>.

PAULOVICH, F. V. et al. Least square projection: A fast high-precision multidimensional projection technique and its application to document mapping. **IEEE Transactions on Visualization and Computer Graphics**, IEEE, v. 14, n. 3, p. 564–575, 2008. <https://doi.org/10.1109/TVCG.2007.70443>.

QU, H. et al. A pedestrian detection method based on yolov3 model and image enhanced by retinex. In: IEEE. **2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)**. [S.l.], 2018. p. 1–5. <https://doi.org/10.1109/CISP-BMEI.2018.8633119>.

RAHIM, H. A. et al. Vehicle speed detection using frame differencing for smart surveillance system. In: IEEE. **10th International Conference on Information Science, Signal Processing and their Applications (ISSPA 2010)**. [S.l.], 2010. p. 630–633. <https://doi.org/10.1109/ISSPA.2010.5605422>.

RÄTY, T. D. Survey on contemporary remote surveillance systems for public safety. **IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)**, IEEE, v. 40, n. 5, p. 493–515, 2010. <https://doi.org/10.1109/TSMCC.2010.2042446>.

REDMON, J.; FARHADI, A. Yolov3: An incremental improvement. **arXiv**, 2018.

REN, S. et al. Faster r-cnn: towards real-time object detection with region proposal networks. **IEEE transactions on pattern analysis and machine intelligence**, IEEE, v. 39, n. 6, p. 1137–1149, 2016. <https://doi.org/10.1109/TPAMI.2016.2577031>.

SAWAS, A. et al. Trajectolizer: Interactive analysis and exploration of trajectory group dynamics. In: IEEE. **2018 19th IEEE International Conference on Mobile Data Management (MDM)**. [S.l.], 2018. p. 286–287. <https://doi.org/10.1109/MDM.2018.00053>.

SCHREDER, G. et al. Supporting cognition in the face of political data and discourse: A mental models perspective on designing information visualization systems. In: IEEE. **2016 Conference for E-Democracy and Open Government (CeDEM)**. [S.l.], 2016. p. 213–218. <https://doi.org/10.1109/CeDEM.2016.23>.

SREENU, G.; DURAI, M. S. Intelligent video surveillance: a review through deep learning techniques for crowd analysis. **Journal of Big Data**, SpringerOpen, v. 6, n. 1, p. 1–27, 2019. <https://doi.org/10.1186/s40537-019-0212-5>.

TAHA, A. et al. Human activity recognition for surveillance applications. In: **Proceedings of the 7th International Conference on Information Technology**. [S.l.: s.n.], 2015. p. 577–586. <https://doi.org/10.15849/icit.2015.0103>.

TIAN, S. et al. Pedestrian multi-target tracking based on yolov3. In: IEEE. **2020 39th Chinese Control Conference (CCC)**. [S.l.], 2020. p. 7558–7564. <https://doi.org/10.23919/CCC50068.2020.9189074>.

VELHO, L.; GOMES, J. d. M. Digital halftoning with space filling curves. **ACM SIGGRAPH Computer Graphics**, ACM New York, NY, USA, v. 25, n. 4, p. 81–90, 1991. <https://doi.org/10.1145/127719.122727>.

VERMA, K. K.; KUMAR, P.; TOMAR, A. Analysis of moving object detection and tracking in video surveillance system. In: IEEE. **2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom)**. [S.l.], 2015. p. 1758–1762.

VEZZANI, R.; CUCCHIARA, R. Video surveillance online repository (visor): an integrated framework. **Multimedia Tools and Applications**, Springer, v. 50, n. 2, p. 359–380, 2010. <https://doi.org/10.1007/s11042-009-0402-9>.

VISHWAKARMA, S.; AGRAWAL, A. A survey on activity recognition and behavior understanding in video surveillance. **The Visual Computer**, Springer, v. 29, n. 10, p. 983–1009, 2013. <https://doi.org/10.1007/s00371-012-0752-6>.

VROTSOU, K.; FORSELL, C.; COOPER, M. 2d and 3d representations for feature recognition in time geographical diary data. **Information Visualization**, SAGE Publications Sage UK: London, England, v. 9, n. 4, p. 263–276, 2010. <https://doi.org/10.1057/ivs.2009.30>.

WANG, J.; PAN, J.; ESPOSITO, F. Elastic urban video surveillance system using edge computing. In: **Proceedings of the Workshop on Smart Internet of Things**. [S.l.]: ACM, 2017. p. 1–6. <https://doi.org/10.1145/3132479.3132490>.

WANG, R. et al. A video surveillance system based on permissioned blockchains and edge computing. In: IEEE. **2019 IEEE International Conference on Big Data and Smart Computing (BigComp)**. [S.l.], 2019. p. 1–6. <https://doi.org/10.1109/BIGCOMP.2019.8679354>.

WANG, X.; LOY, C.-C. Deep learning for scene-independent crowd analysis. In: **Group and Crowd Behavior for Computer Vision**. [S.l.]: Elsevier, 2017. p. 209–252. <https://doi.org/10.1016/B978-0-12-809276-7.00012-6>.

WANG, Z. et al. Visual exploration of sparse traffic trajectory data. **IEEE transactions on visualization and computer graphics**, IEEE, v. 20, n. 12, p. 1813–1822, 2014. <https://doi.org/10.1109/TVCG.2014.2346746>.

WANNER, F. et al. Integrated visual analysis of patterns in time series and text data-workflow and application to financial data analysis. **Information Visualization**, SAGE Publications Sage UK: London, England, v. 15, n. 1, p. 75–90, 2016. <https://doi.org/10.1177/1473871615576925>.

XU, H.; LV, P.; MENG, L. A people counting system based on head-shoulder detection and tracking in surveillance video. In: IEEE. **2010 International Conference On Computer Design and Applications**. [S.l.], 2010. v. 1, p. V1–394.

YUNDONG, L. et al. Multi-block ssd based on small object detection for uav railway scene surveillance. **Chinese Journal of Aeronautics**, Elsevier, v. 33, n. 6, p. 1747–1755, 2020. <https://doi.org/10.1016/j.cja.2020.02.024>.

ZHANG, X. et al. Multi-target tracking of surveillance video with differential yolo and deepsort. In: INTERNATIONAL SOCIETY FOR OPTICS AND PHOTONICS. **Eleventh International Conference on Digital Image Processing (ICDIP 2019)**. [S.l.], 2019. v. 11179, p. 111792L. <https://doi.org/10.1117/12.2540269>.

ZHANG, Z. et al. Multi-scale visualization based on sketch interaction for massive surveillance video data. **Personal and Ubiquitous Computing**, Springer, p. 1–11, 2019. <https://doi.org/10.1007/s00779-019-01281-6>.

ZHU, J. et al. Mme-yolo: Multi-sensor multi-level enhanced yolo for robust vehicle detection in traffic surveillance. **Sensors**, Multidisciplinary Digital Publishing Institute, v. 21, n. 1, p. 27, 2021. <https://doi.org/10.3390/s21010027>.

ZHU, S.; HU, J.; SHI, Z. Local abnormal behavior detection based on optical flow and spatio-temporal gradient. **Multimedia Tools and Applications**, Springer, v. 75, n. 15, p. 9445–9459, 2016. <https://doi.org/10.1007/s11042-015-3122-3>.