POLITÉCNICA

Universidad Politécnica de Madrid

PhD program: Technologies and communication systems

**Modeling human behaviour to improve
retail efficiency and health monitoring**

Author:

Marcos Quintana González

Academic course: 2018-2019

POLITÉCNICA

Universidad politécnica de Madrid

Escuela Técnica Superior de Ingenieros en Telecomunicación

Grupo de Aplicación de Telecomunicaciones Visuales

Directors:

José Manuel Menéndez García

Federico Alvarez García

Abstract

This dissertation presents a model to express the most relevant features of the human behaviour, specifying two personalized instances for retail and health Monitoring domains.

These features are divided into static and dynamic. For the static ones facial features are the most relevant since they can express different attributes of the subjects such as age, gender, emotions ... For that purpose landmarks and pose are the most significant points of interest. However, the body pose can also reveal some interesting features of the current state of one person and should also be taken into account. Regarding the dynamic properties, the trajectories described by a subject in the monitored scenario can also express very useful insights. It is a basic task to delimit the area where the subject is located that is mandatory to perform the static inferences. The interactions of the user with the environment are a very relevant feature, and should be modeled as well.

Thus, initial model has been evolved for the domains previously mentioned. Retail is a scenario where the shoppers express a large amount of features that should be considered by the retailers. Concepts such as density of the areas of the establishment, focus of attention, interactions with the shelves, queues management or characterization of the visitors are included in the model to improve the efficiency of the physical shops. A completely different field such as health monitoring could lead to a completely different approach. But the trajectories of the patients, their facial properties, their interactions or their body pose are also meaningful in this domain to assist decision-making of the health professionals. In our work we have mainly studied the behaviour of the patients with cognitive diseases. This field requires monitoring in different scenarios such as the home of the patient, the clinic centre or the rehabilitation centre. A scalable system and its deployment is proposed to provide an accurate monitoring. In this area we shall also include medical features such as the treatment of the patients, their body pose evolution in physiotherapy sessions or measurements such as their body temperature or their heart rate.

Stated the architecture of every domain, we have gone beyond in two Computer Vision and Machine Learning tasks: human pedestrian tracking and facial analysis. For the first we have developed two solutions that have been tested on real scenarios and on controlled environments that are suitable for a deeper analysis. Feature descriptors such as HOG or LBP have been tested, and Machine Learning classifiers such as Adaboost or Support Vector Machines have been optimized. We propose the best combination for previously exposed techniques and we show results on the different scenarios proposed to validate our proposal.

A 3D visual imaging system is optimized by the proposal of a multi-camera RGB-D system able to capture facial properties at extreme poses. The data collection gathered allow 3D reconstruction of facial areas of 92 subjects captured, and a complete 3D reconstruction pipeline is proposed. Deep learning methods require a large amount of annotated data such as the output of our proposed system. We rely on that insight to perform an innovative data augmentation method for facial landmark detection that has been tested and discussed on two state of the art deep networks for this task. Head pose estimation is another relevant task for this dissertation and relying on 3D reconstruction and facial landmark detection we have implemented a method to validate the alignment of the data captured.

Finally a knowledge base model is provided to store structured data, conceptually based on hierarchical and semantic properties of the domains studied. The data models are considered as the output for the proposed fields, and they are built to improve the performance of professionals in both areas.

Contents

List of Figures	X
List of Tables	XII
1 INTRODUCTION	1
1.1 Problem statement	2
1.2 Motivation	3
1.3 Objectives	3
1.4 Scientific methodology	4
1.5 Contributions	6
1.6 Quality indicators	7
1.7 Thesis overview	8
2 RELATED WORK	9
2.1 Introduction	9
2.1.1 Retail	9
2.1.1.1 Business intelligence	9
2.1.1.2 Behaviour analysis	11
2.1.2 Health monitoring	11
2.1.2.1 Cognitive diseases	12
2.2 Telehealth (architecture)	12
2.3 Interactions	15
2.3.1 Human-machine	16
2.3.2 Human-object	17
2.4 Trajectories	18
2.4.1 Retail	18
2.4.2 Health monitoring	19
2.4.3 Human pedestrian tracking	20
2.4.3.1 WSN and multi-sensing tracking	23
2.5 Facial analysis and pose	24
2.5.1 Features of the consumers	24
2.5.2 Symptoms detection	24
2.5.3 Facial analysis	27
2.5.3.1 3D facial acquisition	28
2.5.3.2 Facial landmark detection	29
2.5.3.3 Head pose estimation	31
2.6 Conclusions	32

3	ARCHITECTURE FOR MODELING HUMAN BEHAVIOUR	33
3.1	Retail	34
3.1.1	Introduction	34
3.1.2	Scenario	35
3.1.3	Logical architecture	36
3.1.4	Physical architecture	38
3.2	Health monitoring	39
3.2.1	Introduction	39
3.2.2	Scenarios	42
3.2.3	Physical and Logical architecture	43
3.3	Conclusions	47
4	HUMAN PEDESTRIAN TRACKING	49
4.1	Background subtraction	49
4.1.1	Non-parametric methods	50
4.1.2	Shadow detection	51
4.2	Features	51
4.2.1	Histogram of Oriented Gradients (HOG)	52
4.2.2	Local Binary Patterns (LBP)	53
4.3	Classification	54
4.3.1	Adaboost	54
4.3.2	Support Vector Machine	56
4.4	Tracking solutions	57
4.4.1	Boosting	57
4.4.1.1	Modified K -means	57
4.4.1.2	Particle filter	58
4.4.2	Hierarchical tracking (HT)	59
4.4.2.1	Template ensemble	60
4.4.2.2	Backprojection map	60
4.4.2.3	Tracking	61
4.5	Testing scenarios	63
4.5.1	Training	65
4.5.2	ETSIT Hall	66
4.5.3	Retail store	67
4.5.4	Cognitive surveillance	67
4.5.5	Lab	67
4.5.5.1	Calibration	68
4.5.5.2	Feature selection	69
4.5.5.3	Multisensing solution	70
4.5.6	Baseline solutions	70
4.6	Results	72
4.6.1	Performance indicators	72
4.6.1.1	Probabilistic interpretation	73
4.6.2	ETSIT Hall	73
4.6.3	Retail store	74
4.6.3.1	Baseline comparison	75
4.6.4	Cognitive surveillance	76
4.6.4.1	Baseline comparison	76
4.6.5	Lab	77

4.6.5.1	Calibration	77
4.6.5.2	Pedestrian detection	78
4.6.5.3	Tracking	81
4.6.5.4	Baseline comparison	82
4.6.5.5	Multisensing solution	83
4.7	Conclusions	84
5	3DWF, FACIAL LANDMARK DETECTION AND HEAD POSE	87
5.1	3D Wide Faces Dataset 3DWF	87
5.1.1	Characterization	88
5.1.2	Multi-camera RGBD setup	88
5.1.2.1	Camera initialization	89
5.1.2.2	Capture	89
5.1.3	Devices optimization	90
5.1.3.1	Distance from model to the frontal camera	90
5.1.3.2	Light source	91
5.1.3.3	Cameras orientation	92
5.1.4	3D Reconstruction algorithm	94
5.1.4.1	Registration	94
5.1.4.2	Refinement	96
5.1.4.3	ROI	96
5.1.4.4	Noise filtering	97
5.1.4.5	Uniform distribution	98
5.2	UVA Dataset	99
5.2.1	Characterization	99
5.2.2	Data annotation and augmentation	100
5.2.2.1	Facial landmark annotation	101
5.2.2.2	Data augmentation	101
5.2.2.3	Bounding box calculation	102
5.3	Facial landmark detection	103
5.3.1	VanillaCNN	103
5.3.1.1	Architecture	103
5.3.1.2	Loss and error	104
5.3.1.3	Results	104
5.3.2	Recombinator Neural Network RCN	107
5.3.2.1	Architecture	107
5.3.2.2	Results	108
5.4	Head pose classification	109
5.4.1	Rigid motion	110
5.4.2	Euler Angles	111
5.4.3	Classification	111
5.4.3.1	Validation methods	111
5.4.4	Results	113
5.5	Conclusions	114
6	CONCLUSIONS AND FUTURE WORK	115
	Bibliography	119

A	DATA MODEL FOR RETAIL	131
A.1	Retail Establishment	133
A.2	Heatmap	134
A.3	Alert	134
A.4	Zone	135
A.5	Hotspot	136
A.6	Interaction	137
A.7	Staff	137
A.8	POS	138
A.9	Queue	139
A.10	Visitor	140
A.11	Satisfaction	141
B	DATA MODEL FOR COGNITIVE HEALTH MONITORING	142
B.1	Patient	144
B.2	Alert	144
B.3	EHR	145
B.4	Medication	146
B.5	Scale	147
B.6	Measurement	147
B.7	Symptom	148
B.8	Professional	149
B.9	Behaviour	150
B.10	Motion	151
B.11	Emotion	152
B.12	Routine	153

List of Figures

1.1	Mindmap to state the problem	2
1.2	Diagram for scientific methodology	4
1.3	Diagram for application-based methodology	6
2.1	Diagram for health monitoring from PERFORM	13
2.2	Diagram for health monitoring from WANDA	14
2.3	Information pyramid for motion analysis	15
2.4	Devices to assist patients	16
2.5	Interaction at the shelf zone	17
2.6	Components for human pedestrian tracking	21
2.7	Classification for target tracking in a WSN	23
2.8	Block diagram for sensing fusion	24
2.9	Block diagram for audience measurement	25
2.10	Framework for gait estimation	25
2.11	Diagram for image normalization	26
2.12	Data collection described in MultiPie	28
2.13	Landmark annotation established by MultiPie	28
3.1	General architecture for human behaviour	33
3.2	Use case diagram for retail	35
3.3	Basic interactions for retail	36
3.4	Logical architecture for retail	37
3.5	Physical architecture for retail	39
3.6	Use case diagram for health monitoring	41
3.7	Sensing devices for health monitoring	43
3.8	Architecture for health monitoring	48
4.1	Shadow detection in Background Subtraction	51
4.2	Gradient orientations in HOG	52
4.3	Block diagram for feature extraction in HOG	52
4.4	Neighbour sets for LBP	54
4.5	Classifiers from Adaboost	55
4.6	Hyperplane from SVM	56
4.7	Block diagram of Boosting for human pedestrian tracking	57
4.8	Block diagram of HT for human pedestrian tracking	60
4.9	Back projection map for HT	61
4.10	Hierarchical tracking	62
4.11	Visual imaging devices for human pedestrian tracking	64

4.12	Scheme designed to train classifiers for human pedestrian tracking	65
4.13	Visual results of Boosting Solution	66
4.14	Phases of visual results of Boosting Solution	66
4.15	Sample frames from retail	67
4.16	Sample frames from health monitoring	67
4.17	Areas of lab scenario	68
4.18	Chessboard used and visual results for the calibration in the lab scenario.	69
4.19	Multi-sensing deployment for lab scenario	70
4.20	Detection results to initialize TLD and KCF for retail	71
4.21	Detection results to initialize TLD and KCF for health monitoring	71
4.22	Detection results to initialize TLD and KCF for lab scenario	72
4.23	Trajectories estimated by muti-sensing solution for human pedestrian tracking	84
5.1	3D Sensors used to capture data	87
5.2	Graphic with features of subjects from 3DWF	88
5.3	Description of 3DWF scenario	89
5.4	Depth sensor triangulation	90
5.5	Visual evaluation of the distance from the model to the frontal camera	91
5.6	Surface covered with the different angles of the cameras.	93
5.7	Graphical explanation of 3D reconstruction for 3DWF	94
5.8	Block diagram of 3D reconstruction for 3DWF	94
5.9	Initial clouds of points obtained	95
5.10	Sample of a complete cloud of points from different perspectives.	96
5.11	3D face parts split for uniform sampling	98
5.12	Sample of the proposed clouds of points	99
5.13	NoTextureMapping	99
5.14	Graphic with the features of the subjects from the UVA Dataset	100
5.15	Flow diagram followed to pre-process the UVA Dataset.	101
5.16	Sample of the annotation of the facial landmarks for the UVA Dataset.	101
5.17	Geometry model for raycasting	102
5.18	Sample of bounding box calculation for the UVA Dataset	103
5.19	Architecture of VanillaCNN neural network	104
5.20	Learning curve for UVA Dataset with viewpoint of 10°	105
5.21	Learning curve for UVA Dataset with viewpoint of 30°	106
5.22	Learning curve for UVA Dataset + AFLW	107
5.23	Architecture of RCN	108
5.24	Comparison of the results for facial landmark estimation	108
5.25	Representation of the last layers of convolution filters from VainillaCNN	109
5.26	Representation of the last layers of max-pooling filters from VainillaCNN	109
5.27	Block diagram of head pose estimation for 3DWF	110
5.28	Graphical plot of Euler Angles obtained for the different markers of 3DWF.	111
5.29	Confusion Matrix for Head Pose Classification.	114
A.1	Base of knowledge for retail	132
B.1	Base of knowledge for health monitoring	143

List of Tables

1.1	Scientific publications upon the work performed on this dissertation.	8
2.1	Results of human-object interaction	18
2.2	Results to estimate roles of the shoppers	19
2.3	Accuracy of features for dyskinetic condition	27
4.1	Specification of parameters to train human pedestrian tracking	65
4.2	Results of Boosting for human pedestrian tracking	74
4.3	Performance indicators of Boosting for human pedestrian tracking	74
4.4	Results of Boosting for impact evaluation	74
4.5	Results of human pedestrian tracking for retail	75
4.6	Performance indicators for human pedestrian tracking for retail	75
4.7	Comparison of results of human pedestrian tracking for retail	75
4.8	Comparison of performance indicators for human pedestrian tracking for retail .	76
4.9	Results of human pedestrian tracking for health monitoring	76
4.10	Performance indicators of human pedestrian tracking for health monitoring . . .	76
4.11	Comparison of results of human pedestrian tracking for health monitoring	77
4.12	Comparison of performance indicators of human pedestrian tracking for health monitoring	77
4.13	RMS results for distortion correction	77
4.14	Calibration error from pixel to cm with selected points	78
4.15	Calibration error from pixel to cm with random points	78
4.16	Results of human pedestrian detection for lab scenario	79
4.17	Results of human pedestrian tracking for lab scenario divided into areas (3Ped) .	79
4.18	Performance indicators of human pedestrian tracking for lab scenario divided into areas (3Ped)	79
4.19	Results of human pedestrian tracking for lab scenario divided into areas (4Ped) .	79
4.20	Performance indicators of human pedestrian tracking for lab scenario divided into areas (4Ped)	80
4.21	Results for pedestrian detection for lab scenario divided into areas with HOG and LBP (3Ped)	80
4.22	Performance indicators for pedestrian detection for lab scenario divided into areas with HOG and LBP (3Ped)	80
4.23	Results for pedestrian detection for lab scenario divided in areas with HOG and LBP (4Ped)	80
4.24	Performance indicators for pedestrian detection for lab scenario divided into areas with HOG and LBP (4Ped)	81

4.25	Results of human pedestrian tracking for lab scenario	81
4.26	Performance indicators of human pedestrian tracking for lab scenario	81
4.27	Results of human pedestrian tracking for lab scenario divided into areas (3Ped)	81
4.28	Performance indicators of human pedestrian tracking for lab scenario divided into areas (3Ped)	82
4.29	Results of human pedestrian tracking for lab scenario divided into areas (4Ped)	82
4.30	Performance indicators of human pedestrian tracking for lab scenario divided into areas (4Ped)	82
4.31	Comparison of results of human pedestrian tracking for lab scenario	82
4.32	Comparison of performance indicators of human pedestrian tracking for lab scenario	83
5.1	Evaluation of the distance from the model to the frontal camera.	90
5.2	Main features of the light source employed in 3DWF setup.	91
5.3	Evaluation of the influence of the luminous flux.	92
5.4	Evaluation of the orientation of the lighting devices.	92
5.5	Evaluation of the orientation of the cameras.	93
5.6	Parameters and results for UVA Dataset with viewpoint of 10°	105
5.7	Parameters and results for UVA Dataset with viewpoint of 30°	106
5.8	Parameters and results for UVA Dataset + AFLW	106
5.9	Results for head pose classification.	113
A.1	Fields included in root table for Retail Establishment.	133
A.2	Fields included in table Heatmap for Retail Establishment.	134
A.3	Fields included in table Alert for Retail Establishment.	134
A.4	Fields included in table Zone for Retail Establishment.	135
A.5	Fields included in table Hotspots for Retail Establishment.	136
A.6	Fields included in table Interaction for Retail Establishment.	137
A.7	Fields included in table Staff for Retail Establishment.	137
A.8	Fields included in table POS for Retail Establishment.	138
A.9	Fields included in table Queue for Retail Establishment.	139
A.10	Fields included in table Visitor for Retail Establishment.	140
A.11	Fields included in table Satisfaction for Retail Establishment.	141
B.1	Fields included in table patient for health monitoring.	144
B.2	Fields included in table Alert for health monitoring.	144
B.3	Fields included in table EHR.	145
B.4	Fields included in table Medication for health monitoring.	146
B.5	Fields included in table Scale for health monitoring.	147
B.6	Fields included in table Measurement for health monitoring.	147
B.7	Fields included in table Symptom for health monitoring.	148
B.8	Fields included in table Professional for health monitoring.	149
B.9	Fields included in table Behaviour for health monitoring.	150
B.10	Fields included in table Motion for health monitoring.	151
B.11	Fields included in table Emotion for health monitoring.	152
B.12	Fields included in table Routine for health monitoring.	153

Abbreviations

3DWF 3D Wide Faces

3Ped Three Pedestrians Sequence

4Ped Four Pedestrians Sequence

AAM Active Appearance Model

AFLW Annotated Facial Landmarks in the Wild

AUC Area Under the Curve

BLE Bluetooth Low Energy

BNs Bayesian Networks

CHU Centralized Hospital Unit

CLM Constrained Local Model

cm centimeters

CMOS Complementary Metal–Oxide–Semiconductor

CNN Convolutional Neural Network

CPM Convolutional Pose Machines

DFT Discrete Fourier Transform

EHR Electronic Health Record

ETSIT Escuela Técnica Superior de Ingenieros de Telecomunicaciones

FK Foreign Key

FPGA Field-Programable Gate Array

fps Frames per second

GCC General Correlation Coefficient

GDPR General Data Protection Regulation

GMM Gaussian Mixture Model

GNB Gaussian Naive Bayes

HD High Definition

HF High Frequency

HOG Histogram of Oriented Gradients

HSV Hue Saturation Value

HT Hierarchical Tracking

I/O Input or output operations
ICT Information and Communication Technology
ISP Internet Service Provider
JCR Journal Citation Reports
KCF Kernelized Correlation Filter
kNN k Nearest Neighbours
LBP Local Binary Pattern
LDA Linear Discriminant Analysis
LF Low Frequency
LID Levodopa-Induced Dyskinesia
lm Lumen
MAD Median Absolute Deviation
MBLBP Multi-scale Block Local Binary Patterns
MEEM Multiple Experts Using Entropy Minimization
mm millimeters
NB Naive Bayes
PCA Principal Component Analysis
PD Parkinson's disease
PDM Point Distribution Model
PK Primary Key
POS Point of Sale
RANSAC Random Sample Consensus
RCN ReCombinator Neural Network
R & D Research and Development
RF Random Forests
ROI Region of Interest
RPM Remote Patient Monitoring
SD Standard Definition
STD STandard Deviation
SVM Support Vector Machine

TLD Tracking Learning Detection

UDysRS Unified Dyskinesia Rating Scale

UPM Universidad Politécnica de Madrid

UVA University of Amsterdam

WMSMU Wearable Multi-Sensor Monitor Unit

WSN Wireless Sensor Network

Chapter 1

INTRODUCTION

This dissertation is a scientific work which aims to achieve a notable impact for the Research Community. For that aim the candidate has been oriented by his directors through the application of a specific methodology relying on engineering and scientific principles. Therefore, its main objective is to enhance human knowledge in the areas of research involved:

1. **Multi-sensing ICT architectures** for human behaviour analysis.
2. **Computer Vision & Machine Learning** algorithms for human behaviour analysis.
3. **Data modeling** for human behaviour analysis.

Information and Communication Technology (ICT) has surged in importance in recent times due to the advances of different kinds of technology such as Signal Processing or Machine Learning. Recently developed systems allow automatic processing of large amounts of data in order to provide relevant and specific information for different aspects of human life. The work presented here is intended to extract features able to describe and quantify human behaviour and evaluate its feasibility in two uses cases scenarios: retail and health monitoring. For that aim, experts in both areas have been consulted so that the system has an adequate response for their needs, and provides accurate information for them to ease decision-making in both areas, and also increase efficiency of ICT systems already available.

This kind of systems require different levels of abstraction from the highest level (architecture) to the lowest (signal processing algorithms). The highest level should define the different modules involved and how they communicate with each other. Every one of them should be defined according to the following properties: input, scope, data storage, data flow, latency and output. In our case we focused the architectures proposed in the domain of the use cases previously mentioned. The top level of abstraction requires an effective and considerable level of communication with professionals of the target environment in order to design a system that fulfill their requirements. Those requirements are expressed in a data model that save the most relevant and valuable information to assist the professionals on their daily routines.

The lowest level requires a more specific knowledge of the signal processing field (in our case Computer Vision) and Machine Learning. The sensing technologies used and the algorithms to extract significant information from them for the problem addressed should be defined, implemented, deployed and tested.

1.1 Problem statement

The problem challenged is described in the title of this dissertation: *Modeling human behaviour to improve retail efficiency and health monitoring*. Initially, a generic model able to reflect and classify generic activities in human life which can be estimated by a system based on signal processing is required. A mindmap displaying the most relevant concepts of them is shown on figure 1.1.

Figure 1.1: Diagram showing the mindmap that defines the scope of the problem

It can be noted that concepts are classified in four categories:

1. Surveillance. Most of surveillance the systems in recent times are focused on localization of humans at a global scenario which can provide significant information concerning trajectories, most visited areas of the scenario, frequency of visitors or interactions by one/several person(s).
2. Facial analysis. Facial analysis allows low-level estimations such as facial landmarks, head pose or expression. With these kind of estimations upper abstractions such as characterization of humans (age, gender ...), focus of attention, or degree of satisfaction can be achieved.
3. Data science. Previous concepts rely on information gathered by a sensing system that may be processed through Machine Learning techniques such as regression or classification and annotated following a ground truth procedure that allows a proper automatic inference.
4. Expert output system. This concept is mostly focused to provide relevant and accurate information for the experts in the proposed field, once signals captured have been processed. Information related to user experience (preferences, satisfaction), its behaviour or its current state (measurements) may be significant for that aim.

This dissertation proposes two use cases whose specific properties may be specified:

1. **Retail.** Smart retail stores offer an exciting opportunity in terms of technology development. These establishments require ICT solutions whose main targets are to make it easier for visitors to find suitable products for their needs, help manage the workload of the employees, evaluate paths and interactions of the visitors in order to adapt the distribution of the establishment and characterize which kinds of visitors enter the shop, their frequency of visits and preferred timing.
2. **Health monitoring.** Health systems have opened a new door for ICT systems in recent years. Several subsystems such as telehealth, remote assistance or behaviour analysis have been implemented. The focus of this problem is mainly on behaviour analysis for patients with cognitive problems, but the other subsystems should also be included in the architecture. It should be remarked that in this use case it is required to evaluate the patient in different scenarios, and that they may require personalized solutions for their current state. Therefore scalability is a concept with a significant impact for this area.

The proposed approach aims to explore the different perspectives required to provide a robust solution for the problem proposed that defines all the processes required to cover all abstraction levels.

1.2 Motivation

The main motivation of a dissertation like this, is to tackle one problem following a scientific methodology (details on subsection 1.4) that make a contribution to the Scientific Community. In this scope, attending conferences and submitting and reviewing manuscripts for Scientific Journals is a new area explored in this work that provides an expansive knowledge and human experience for a PhD candidate. The proposed work includes all the phases of scientific methodology from background analysis to results discussion through hypothesis formulation, which makes it very profitable.

Analysis and modeling of two fields with a relevant interest for Research Community such as retail or health monitoring is very enriching. The first is more intended for an application-based and the second one helps to improve quality of life. Combining both sides with the suitability of a generic human behaviour model for the use cases proposed makes the topic very attractive. The relationship with experts in both areas is also an exciting opportunity. Both of them imply different backgrounds that alongside technology development compose a fascinating demand.

Finally experimentation with Computer Vision & Machine Learning algorithms make this work very challenging and rewarding. This step includes handling different types of devices for visual sensing acquisition and several frameworks to implement systems with high performance and accuracy.

1.3 Objectives

Several objectives have been formulated for the proposed topic:

1. **Modeling.** Proposed areas of knowledge should be explored alongside close collaboration with experts in order to have an enclosed area of research. Then the initial model is designed, tested and improved for human behaviour.
2. **System architecture.** The model contains different semantic entities that require some resources in terms of signal acquisition, data processing and presentation that may be overcome by an architecture whose modules cover those demands.

3. **Algorithm development and optimization.** A structure of SW for signal processing needs to be developed to perform the inferences demanded by the model through employing architectural resources. Different geometric, statistical or mathematical transformations need to be implemented by using different devices, libraries or frameworks.
4. **Deployment and testing on real scenarios.** Finally these developed technologies should be set up in the corresponding scenarios for the scope of this dissertation. Iterations through data capture and processing allow for the improvements in terms of efficacy for the system.

1.4 Scientific methodology

This kind of methodology is very specific for a dissertation like the one presented here. Steps followed are defined in fig. 1.2.

Figure 1.2: Steps of the scientific method. Extracted from [3].

1. **Ask a question.** The following questions have been formulated:
 - (a) Can human behaviour analysis be properly modeled for the proposed uses cases?
 - (b) Which technologies are the most suitable for this analysis?
 - (c) How can they be organized, connected, and implemented?
 - (d) Which outputs provide a significant knowledge for experts in the proposed areas?
2. **Do background research.** Research publications located on the different abstraction levels relevant for the proposed topic are exposed and analysed in Chapter 2. Several iterations have been performed in this task to update the state of the art for the tasks related to the subject.

The next loop (**Hypothesis, test, results and report**) has been repeated several times to keep tracking the evolution of the system.

3. **Construct an hypothesis, test your hypothesis by doing an experiment, analyze your data and draw a conclusion and communicate your results.** Several hypothesis have been built to target the proposed system. For every development task: architecture, human pedestrian tracking and facial analysis; an initial hypothesis was postulated and has been iterated through experiments, data analysis, conclusion and results sharing to arrive to the final thesis.

The last loop mentioned requires to be further explained in order to provide more details regarding the methodology followed to propose an adapted solution for the two application fields explored in this dissertation (retail and health monitoring). Both solutions proposed have been developed in parallel by the cooperation of two teams: the technical team and the experts team. For both processes the technical team has been leaded by the candidate and the directors of this dissertation. Regarding the expert teams, both of them have been built upon the working environment provided by two Research and Development (R & D) projects:

1. **ICT4Life (H2020).** This project includes institutions represented by experts in the different fields involved in the treatment of patients with cognitive diseases: neurologists, associations of patients and integrated care developers. The neurologists own a large knowledge related to the medical side of the diseases, the associations of patients are in direct contact with both patients and caregivers, and the integrated care developers are updated regarding most critical aspects of technology development in this area.
2. **LPSBigger(CIEN).** This project standardizes and optimizes SW production with an specific work package dedicated to the development of smart solutions for retail environment. In this work package the technical team has worked in collaboration with a company expert on this area. This company is in direct contact with stakeholders from the Internet Service Providers (ISP) and supermarket segments, that own a large knowledge related to the retailers requirements and user experiences in the stores.

In addition, we should mention that all the data collected for this dissertation have been collected by means of General Data Protection Regulation (GDPR) in the environments promoted by both R & D projects. The different steps of the methodology followed for the uses cases mentioned can be noticed in fig. 1.3.

Figure 1.3: Steps of the method followed to develop application-based solutions for retail and health monitoring

Development loop includes four steps: literature review, technical proposal, validation calls and expert input. Different iterations have been performed on this loop in order to achieve two reliable solutions for the use cases proposed. The main targets of the proposed methodology can be divided in three levels:

1. Level 0: Behaviour modeling.
2. Level 1: Requirements, data formats, latency, user experience and back-end.
3. Level 2: Architecture, sensing devices, algorithms, data model and interfaces.

Level 0 covers a wide area that is divided into more specific steps for the design of ICT systems in level 1, concluding in a more concrete approach with the elements included in level 2. The behaviour is modeled in order to enhance the user-experience at the scenarios concerned by the two use cases enumerated. This work provides a full description of the architecture to accomplish requirements formulated in the analysis step. The design includes the different natures of sensing devices and the hierarchical connections among them. Individually, every sensing device provide outputs whose logical contribution to solve the problem enunciated is determined by their data formats or their latency. The signal gathered by the sensing devices is analyzed through algorithms that partially feed the data model proposed, whose design is highly influenced by the knowledge of the experts. The interaction with the user may be performed by a dedicated interface for this kind of systems, whose implementation is out of the scope of this dissertation. Analogously, a back-end shall provide HW resources to host the architecture, establish communications with the sensing devices, run the corresponding algorithms, maintain the data model and hold the interfaces. But practical details about this area are not involved in the proposed work.

1.5 Contributions

The contributions of this dissertation can be divided in three groups:

1. Two dedicated **architectures** for the use cases proposed (**retail and health monitoring**). This work proposes two logical architectures based on signal processing and built upon the requirements discussed with the experts on both fields. Physic layer of the architecture is defined as well to cope those requirements, and finally the storage layer is prototyped by two specific data models.
2. **Human pedestrian tracking**. Deployment and optimization of a visual imaging system at the specific scenarios relevant for the proposed topic in order to infer human trajectories are presented. The proposed solution is discussed from the most suitable sensing devices to calibration methods, going through the most relevant feature descriptors and some classification algorithms.
3. **3D facial analysis**. Contributions in this field are enumerated below
 - (a) An optimized **RGB-D multi-camera** system.
 - (b) **3DWF**. Streams from 600 to 1200 frames of RGB-D data from 3 cameras for 92 subjects and clouds normalized to 2K points for 10 poses proposed and their corresponding 3D landmarks projected have been included in the proposed dataset. Demographic data such as age or gender is provided for every subject as well. Such a complete dataset in terms of the number of subjects and different imaging conditions is unique and the first of its kind.
 - (c) An innovative **data augmentation method for facial landmark detection**. This method is based on a 3D(mesh)-2D projection implemented by raycasting.
 - (d) **3D reconstruction workflow**. Adapted to facial properties and their normalization to provide meaningful features for cloud formats.

1.6 Quality indicators

The scientific work performed for this dissertation has lead to different publications in international scientific journals that are enumerated in table 1.1. Where Q stands for quartile of the Journal Citation Reports (JCR).

The scope of every of the publications enumerated is exposed below:

1. Ref. [104] introduces an architecture for human behaviour modeling in a retail environment.
2. Ref. [105] exposes the 3D facial dataset gathered and the deep learning experiments performed.
3. Ref. [47] proposes a multi-sensing architecture for human behaviour inference, specially regarding trajectory estimation.
4. Ref. [7] reveals and architecture for human behaviour modeling in a health monitoring environment.
5. Ref. [122] includes a complete ICT platform with integrated care purposes.

Journal	Title	Authors	Year	Q	JCR Area
Pattern Recognition Letters	Improving retail efficiency through sensing technologies	M. Quintana, J. M. Menéndez, F. Alvarez, J.P. López	2016	Q1	Computer Vision & Pattern Recognition
MDPI Sensors	3D Wide Faces (3DWF): facial landmark detection and 3D reconstruction over a new RGB-D multi-camera dataset	M. Quintana, S.Karaoglu, F. Alvarez, J. M. Menéndez, T. Gevers	2019	Q2	Electric & Electronic Engineer.
IEEE Access	A Multi-sensor Fusion Scheme to Increase Life Autonomy of Elderly People with Cognitive Problems	G. Hernández-Peñaloza, A. Belmonte-Hernández, M. Quintana, F. Alvarez	2017	Q1	Engineer.
IEEE Multimedia	Behaviour analysis through multimodal sensing to improve Parkinson and Alzheimer patients quality of life	F. Alvarez, . . . , . . . , M.Quintana	2018	Q1	Media Technology
Studies in Health Technology and Informatics	ICT Services for Life Improvement for the Elderly	A. Sanchez-Rico, . . . , . . . , M.Quintana	2017	Q3	Health Informatics

Table 1.1: Scientific publications upon the work performed on this dissertation.

1.7 Thesis overview

This dissertation is organized as follows:

1. **Chapter 2** enumerates and evaluates different methods relevant for this dissertation.
2. **Chapter 3** structures and communicates all the required modules to solve the proposed problem.
3. **Chapter 4** reveals two human tracking pedestrian algorithms deployed and tested on four different scenarios.
4. **Chapter 5** proposes a multi-camera RGB-D setup, a 3D reconstruction algorithm directed towards face modeling and an innovative data augmentation procedure related.
5. **Chapter 6** discloses the outcomes of this dissertation.
6. **Appendix A** proposes an expert-based semantic and hierarchical data model for retail.
7. **Appendix B** proposes an expert-based semantic and hierarchical data model for health monitoring.

Chapter 2

RELATED WORK

This chapter is intended to review all the relevant literature for the topic of this dissertation and has been divided by following a top-down approach. The first subsection is focused on the introduction of the context for retail and health monitoring. Once the main objectives of every domain are revealed it is required to disclose the architectures relevant to cope with them. Different global architectures are proposed in the literature for health monitoring domain, however we shall mention the lack of global architectures for the retail domain. Retail and health monitoring are two fields widely studied in recent times. Both of them are joined by the synergy of understanding human behaviour. For that purpose they rely on basic human features such as interactions, motion, pose or facial properties. Their two most significant tasks are selected to go further in the signal processing analysis: human pedestrian tracking and facial analysis.

2.1 Introduction

This subsection provides some insights from the background research performed on the retail and health monitoring domains in order to delimit their scope. Business intelligence has a great impact on smart stores, and therefore works located in this area may be analysed to state advances performed on the retail use case. In the case of health monitoring, this introduction is focused on the most suitable technologies, and some particularities for cognitive diseases.

2.1.1 Retail

The ambition of implementing smart retail systems is focused on two targets: increasing shoppers satisfaction and increasing sales. Both of them are related, but the first one is intended to analyse and study behaviour of the shoppers and the second one is originally related to business intelligence, marketing ... Our work is mostly oriented to the first target, but some concepts related to the second target will be exposed as well.

2.1.1.1 Business intelligence

Retail data is increasing exponentially in volume, variety, velocity and value with every year. Current reports summarizing average behaviour do not provide the useful insights needed to determine how individual customers will act in their next visit. In order for retailers to create a meaningful dialogue with customers, predictive analytics are required to provide the opportunity to significantly change the retail marketing industry. Predictive analytics estimate what a data

set could potentially signify in the future rather than merely explaining what data means. There are varying goals for data science in this field:

- Identification of shopping trends and cross-selling opportunities through visual data analysis. This kind of inference is directed towards human behaviour analysis and is where our work is located.
- Targeted campaigns using analytics to segment consumers, identify the most appropriate channels and achieve optimal region of interest. We provide a low-level data knowledge to increase accuracy in this direction.
- Personalized recommendations and multi-level reward programs based on purchase preferences, online data, smartphone apps, etc. These systems are focused on combination of user features preferences with segment of market targeted by the retail establishment to offer products based on personalized demand.
- Insights using product sensors that provide real-time information and determine their post-purchase use. The goal of this approach is more adapted to user experience.
- Demand-driven forecasting through a combination of structured and unstructured data. This high level of abstraction is strongly related with the proposed modeling in this dissertation, including a data structure that feeds these kind of systems.

Recommendation methods like the one implemented in [10] attempt to predict the next optimum user selection based on its features and previous experiences. Other methods like [109] implement statistical methods, Bayesian Networks (BNs [36]) in this case, to simulate a sales module and study response from the shopper. Stock management is another field where Data Science has contributed, it was proved by Xi et al [139], they proposed a new data model that increases efficiency of the Automated Storage and Retrieval System of retail stores.

Targets specified above require retailers to be proactive in managing and utilizing corporate data if they want to hold on their position on the market. That is where business intelligence comes in. There are different motivations to consider business intelligence relevant for smart retail:

- Identifying relationships among the information, and learning how different factors affect each other and the crux of the company.
- Companies need to simultaneously analyse multiple layers of information to better understand customer needs and behaviour, from global analysis to the individual point-of-sale record.
- A retailer will have many people in different locations with distinctive skills who need to use this information for varying purposes.

Different experiments have been performed to analyse intentions of the potential purchasers and/or to attract more consumers:

- [130] relies on congruity between the multi-channel retailers, land-based and online stores, obtaining strong implications within the data collected.
- [112] and [121] try to infer from shoppers their purchase intentions, the first of them by introducing simulated products, and the second one by analysing the behaviour of the user.

- Finally other methods such as [142] or [53] based on an incentive scheme, propose submission of coupons in almost real time through smart mobile devices, and day-ahead dynamic prices in order to attract more consumers.

We propose a data model that stores complete information from low-level sensing oriented algorithms. This model accurately exposes relationships among human behaviour properties in order to perform a specific data analysis that might be used as input for techniques such as the ones previously enumerated.

2.1.1.2 Behaviour analysis

Smart retail is a term used to describe a set of smart technologies which are designed to give the consumer a greater, faster, safer and smarter experience when shopping. For that aim different aspects of the consumers behaviour should be captured by the proposed system:

1. **Human-product interaction**
2. **Trajectory analysis**
3. **Features of the consumers**

They will be further explained in the next subsections.

2.1.2 Health monitoring

Remote patient monitoring (RPM) has enhanced the ability of clinicians to monitor and manage patients in nontraditional healthcare settings. More specifically, noninvasive technologies are now commonly being integrated into disease management strategies to provide additional patient information, with the goal of improving healthcare decision-making. Therefore these kind of monitoring systems could be defined in the following way "Usage of digital technologies to collect health data from individuals in different locations, such as the home of the patient, and electronically transmit the information to healthcare providers in a different location for assessment and recommendations [40][22]". Interventions are categorized based on technology using ([127]):

1. **Any smartphone or PDA device** (or associated software/ application/text messaging) that is used to transmit patient data to the physician/researcher. This task addresses data transmission.
2. **Any wearable device** worn or placed on the body part to record a particular physiological change (e.g., blood pressure monitors). The output from these kind of modules are physiological measurements.
3. **Any biosensor** device for recording data from biological or chemical reactions (e.g., pulse oximeters). In this case measurements are not only based on technology, but also in some bio-science matters.
4. **A computerized system** where data is entered by the patient (or caregiver) through an internet connection. In this case we have an interface (front-end) that allows human annotation.
5. **Multiple components** containing more than one of the categories of technology mentioned above (e.g., biosensor device and computerized system). Different levels of abstractions should be fused to provide more reliable and global information.

2.1.2.1 Cognitive diseases

Concerning cognitive diseases various solutions have been proposed to detect the symptoms associated mostly to PD and relying on motion features of the patients that will be presented and evaluated. Symptoms associated to cognitive diseases are highly correlated to the pose of the corresponding patient. Therefore, there are several works for Parkinson scope that rely on Computer Vision techniques to estimate the pose and analyse the correlation with the different symptoms specified and current state of the patient.

Levodopa is the gold standard therapy for PD, but its prolonged usage leads to additional motor complications, namely Levodopa-Induced Dyskinesia (LID). To assess LID and adjust drug regimens for optimal relief, patients attend regular clinic visits. However, the intermittent nature of these visits can fail to capture important changes in the condition of the patients. With the recent emergence of Machine Learning techniques achieving impressive results in a wide array of fields including computer vision, there is an opportunity for video analysis to be used for automated assessment of LID.

Advances in Computer Vision & Machine Learning have increased reliability in patient monitoring by visual imaging systems. Therefore, wearable devices and biosensors could be complemented by cameras and corresponding image processing solutions to provide more complete outputs by analyzing human behaviour. Going beyond in behaviour analysis, sensing-oriented systems could also provide more specific information such as symptoms or human-machine interaction to provide a hierarchical model. This hierarchical model (architecture) is located in the telehealth area, involving as well other natures of data required to provide useful information for health professionals.

2.2 Telehealth (architecture)

Telehealth is the remote exchange of data between a patient at home and their clinician(s) to assist in diagnosis and monitoring, and it is typically used to support patients with Long Term Conditions. This subsection will cover different architectures published to fulfill this paradigm. The first of them is the PERFORM [126] system, which consists of three subsystems:

1. **Wearable Multi-Sensor Monitor Unit (WMSMU)** which is physically attached to Parkinson's disease (PD) patient's body. The key role of this unit is to facilitate the monitoring of the daily motor activity of the patients, and status through the continuous recording of specific signals. The signals recorded through five sensors are transferred to the Local Base Unit where they are stored and processed.
2. **Local Base Unit** which is composed of a touch screen computer located in the environment of the patients along with the WMSMU and the test devices. It is mainly responsible for uploading, storing and processing of the raw signals coming from the test devices. Its outcomes are the identification and quantification of motor symptoms alongside the diary keeping of the patients (its entries store the timing of drug and food intakes).
3. **Centralized Hospital Unit (CHU)** is positioned in the clinician's setting. The CHU is dedicated to processing all the data from the patient, and assisting the treatment decisions of the clinicians. The CHU subsystem is responsible for further processing the classified results of the Local Base Unit in order to extract further knowledge and to generate alerts to inform the clinician of the condition of the patients.

The three core components of the CHU subsystem are:

- (a) Alert Manager
- (b) Information Manager
- (c) Interoperability Manager .

Fig. 2.1 explains the different layers of the proposed system. It can be noted that there exists different processing units (from sensing acquisition to Data Mining), different data containers (from user data to activity recognition) and different scenarios such as home of the patients or clinic ambulatory contemplated by the proposed architecture. The most important contributions of this work are: the proposal of hierarchical subsystems, and the CHU (Back-end) core components to communicate with patients, and ease information access for health professionals. Although a more specific information modeling for this kind of disease is not included in the manuscript.

Figure 2.1: Diagram showing the proposed workflow for PERFORM system.

WANDA is a three-tier, end-to-end remote monitoring system with extensive hardware and software components designed to cover the broad spectrum of the telehealth and remote monitoring paradigm [65]. The overall architecture is summarized in 2.2. The first tier of the architecture consists of a data collection framework, which is formed from a heterogeneous set of sensing devices whose measurements include weight, body fat, body water, blood pressure, heart rate, blood glucose, blood oxygen saturation and body movements. However, the manuscript does not provide any details about the architecture and deployment of those devices, nor its accuracy to measure bio features, and some of them require a either specific or complex setup. The data from these sensors are collected, processed, and transmitted via a smartphone-based gateway to the cloud (the second tier of the WANDA architecture). The large amount of data is stored and indexed using a scalable database, and it is easily accessible using a web interface. The last tier of the WANDA architecture is a backend analytics engine capable of continuously generating statistical models and predicting outcomes using various Machine Learning and Data Mining algorithms. In our case we believe that another relevant feature to evaluate current cognitive

state of the patient is behaviour analysis. For that reason we want to cover this matter with a dedicated framework that provides additional information to the team of health professionals.

Figure 2.2: Diagram showing the proposed workflow for Wanda system.

A framework for motion analysis that describes how to transform the data from one level of the information pyramid to the next, is introduced in [110] and illustrated in fig. 2.3. The diagram presented provides a more concrete visualization of how different framework processes relate to each other and to the different abstraction levels of the information pyramid. The framework consists of four general processes:

1. **Symbolization**, which creates an intermediate data representation: symbols.
2. **Context analysis**. The relationships between symbols are investigated through context analysis. These relations may be expressed through patterns, sequences of symbols or rules.
3. **Expert system**, which is responsible for mapping patterns or symbols to human concepts or linguistic descriptions of the system.
4. **Characterization**. Symbols and patterns can also be used directly to characterize different aspects of the movement, independently of expert knowledge.

Figure 2.3: Diagram showing the information pyramid and an instance/framework proposed for motion analysis.

The second layer in the framework pyramid can be divided into two. The right side relates to processes that incorporate expert knowledge, namely context analysis and expert system, whereas the left side deals with characterization. Context analysis identifies patterns or structures in the symbolic data (e.g., in/out of bed). Symbolic data relates context analysis with time dimension and routines of the patient (e.g., they get out of bed after 9 AM). Having identified patterns of the patient, their meaning may be determined by including expert knowledge. The expert system is the process that makes the connection between patterns in the symbolic data and expert knowledge (including information about caregivers and persons related to the patient). At this level of abstraction the data conveys information about the daily activities of the patient. The left side of the second layer of the framework pyramid converts symbols for information through characterization. This process typically quantifies certain aspects of the symbolic data. This characterization of activities of the patient can be used to compare the well-being state before and after treatment, or determine the rate of recovery of the patient. The third layer in the information pyramid is concerned with converting information into knowledge. One common way to achieve this is through classification. In this example, one possibility would be to determine whether or not the patient is in good health based on their daily activities. The last layer in the information pyramid transforms knowledge into expertise. One possible way to achieve this is through Data Mining. We have found this work really relevant for our solution, but we believe a higher level platform including other clinical aspects such as specific scales or treatment should also be included in the model in order to provide a full perspective to health professionals in order to perform a proper evaluation of the patient.

2.3 Interactions

Once defined the ambit of both use cases related, it is mandatory to go beyond in the low-level available solutions in order to assist professionals from the mentioned fields. In this section an analysis of the literature related to the most relevant interactions performed in the corresponding scenarios is presented. For health monitoring we have found some significant works concerning the suitability of different kinds of devices to communicate with the patients. Regarding smart stores, interactions are oriented to the analysis of the synergies between the products offered by the establishment and their visitors.

2.3.1 Human-machine

In the health monitoring context, human-machine interaction is intended to identify new ways in which existing knowledge from the fields of medicine, nursing, psychology, cognitive science and computer science can be combined and used to support people with cognitive-related diseases. Existing assistive tools for that aim have been analysed in [138] (devices can be noted on 2.4) concluding that these tools may be replaced by mobile applications combined in a single suite of applications. The most relevant needs identified are the following ones:

1. **Memory support.** Very consistent with cognitive current state.
2. **Daily activities support.** Intended to evaluate motion as well.
3. **Social interaction support.** To increase mental satisfaction of the user (patient).
4. **Support to feel safe.** Thereby increasing the patients sense of freedom.

Figure 2.4: Diagram showing the different devices tested in [138] to assist patients.

Next features shall be included for a interface that provides a complete user-experience for the patients:

1. Multi-role. Caregivers and patients have different privileges to access and modify information stored.
2. Voice recorders and instructions. To ease patients daily activities and increase their autonomy.
3. Quiz games and music collections. To perform evaluations based on their knowledge evolution.
4. Photos albums with recording/narration. To improve quality of life.
5. Recommendation prompts and emergency procedures. Telehealth can help to avoid dangerous situations.

In the same way the information retrieved from the activity of the patients with these kind of apps can provide relevant information to assess their current state. This process has been implemented in [30] to perform early detection in cognitive decline by analysing semantic memory performance of 1002 elderly agricultural workers in rural France. Scores significantly correlated with baseline neuropsychological test scores, and stimulating activities (crossword puzzles, reading) were prospectively associated with increases in semantic memory performance over a subsequent three-hour period of the same day. In our proposed health monitoring architecture we propose a module to store the most relevant paths that the patients follow through the application, and to compare them with the most common paths for every one of them in order to state its current cognitive state.

Figure 2.5: Interaction instance and setting parameters of the shelf zone from [75].

2.3.2 Human-object

The relationships between shoppers and the products offered in a retail environment are very valuable information in order to maximize their attraction by optimization of the distribution of the shop and the visibility of the products. The information provided by common RGB sensors is not enough to accurately extract that information with an acceptable computation complexity in this scope. Due to that fact different technologies have been employed to satisfy the needs of retailers. Liciotti et al [75] propose a multi-camera architecture with RGB-D sensors, requiring an initial configuration step where the shelf regions of the establishment should be specified by the retailer. The output of this initial step is a threshold to determine the interactions of the consumers. Visitors to the shop are tracked, and when they reach the specified zone if the hand exceeds the threshold it is considered as an interaction. The proposed system classifies captured interactions in the following way: positive (the product is picked up from the shelf), negative (the product is taken and then put back on the shelf) and neutral (the hand exceeds the threshold without taking anything). We consider the focus of attention also related to this topic and we think that head pose and gaze estimation should also be included to perform a more complete classification. An example of this interaction can be observed on fig. 2.5

A complex infrastructure of wireless embedded sensors is deployed in a store to acquire human-product interaction in the work implemented by Pierdicca et al [100]. The proposed system architecture consists of an active sensor network which can be arranged throughout the store and a series of smaller beacons which can be attached to the items of the store. The main components of the system are: BLE sensors (Estimote Beacons), smart devices (BLE integrated receiver is mandatory), master root (BLE Receiver, Internet connection management and data collection) and server (data management and data analysis). This system provides more accurate information by an output that combines local and global rates such as: total number of people, average visiting time, frequency of visiting for each area of the store, number of people passing by, average group number and number of interactions per person. But we postulate that visual imaging systems could provide more accurate information to improve the performance of mentioned rates. In the same context an extensive experiment was performed in [101] to evaluate the feasibility of stable landmarks in a retail environment, and to demonstrate its utility in the development of next generation apps. A clustering algorithm is implemented to perform non-intuitive feature combination of sensors like Accelerometer, Gyroscope, Magnetometer, Light, Sound, Wi-Fi, GSM signal strength etc. Heterogeneity on different parameters such as people (walking style) or time of the day have been tested on different devices, concluding that their order of influence is analogous to the one we have followed in our exposition, in a decreasing sense. Visual imaging systems have shown a great accuracy on the analysis of walking style at different indoor scenarios. An approach based on RFID sensors is proposed in [86] whereas real-time human-object interaction detection is performed. In the proposed scenario, for each inventory

Method	Precision	Recall	F-Score	Accuracy
Baseline x_1	0.740	0.901	0.813	0.792
Baseline x_4	0.837	0.242	0.375	0.597
Baseline x_5	0.851	0.565	0.683	0.737
Pair-wise $x_{1,4}$	0.738	0.925	0.821	0.798
Pair-wise $x_{4,5}$	0.857	0.650	0.739	0.771
Pair-wise $x_{1,5}$	0.723	0.972	0.829	0.799
$x_{1,4,5}$	0.742	0.972	0.841	0.817

Table 2.1: Evaluation of interaction algorithms to estimate human-object interaction from [86].

round r , n antennas are sequentially activated t seconds, eventually returning samples S with timestamp p from the tag population. Deployment of this system is complex and expensive, and we find it too sensitive if we analyse it in a cost-accuracy fashion. Three events are used as feature vectors in solutions analysed: x_1 (Received Signal Strength Indication, RSSI), x_4 ($-\Delta$ RFP—, Radio Frequency Phase difference) and x_5 (Number of antennas), to build a BN whose results are compared in table 2.1.

In our proposed work we do not include any specific solution to infer this kind of information, but we propose an architecture where this kind of sensing solution could be integrated, and we have also contemplated the data gathered so that it could be used for retail establishments in order to optimize its performance.

2.4 Trajectories

Trajectories followed by pedestrians, and consequently human pedestrian tracking, has already been named in the previous section as one of the mandatory steps required to delimit the Region Of Interest (ROI) and perform local analysis to extract significant features from the human behaviour at the scenarios defined. This section will initially state the application domain for this task in the two use cases presented, and finally some Computer Vision & Machine Learning techniques related for this task will be introduced.

2.4.1 Retail

The paths followed by the visitors at the retail establishments are critical to evaluate the attractiveness of the different areas in the store, and perform further local analysis. The feasibility of these kind of systems with imaging systems was initially proved by Makela et al. [79]. They propose a customer behaviour tracking solution based on 3D data. Experiments with people tracking, and analysis of the trajectories in a department store show that the use of inexpensive 3D sensors and lightweight computation allows classifying shopping behaviour into three classes (passers-by, decisive customers and exploratory customers) with a promising accuracy of 80 %. Surveillance space is divided in regions, and an adjacency matrix is built in order to separate small groups. The following aspects are extracted from the behaviour of the shoppers: audience structure, flow pattern and scene character. As previously mentioned, division and its outputs is a very promising contribution that we have in our work. In [107] Machine Learning methods are tested on real-world digital signage data from the perspective of the observer to predict consumer behaviour in a retail environment. Optimum performance is achieved by SVM (Support Vector Machine [24]), and the application of the solution is oriented towards the purchase decision process. Most relevant results obtained are shown in table 2.2. (NB is the Naive Bayes Classifier[67]

Method	Initiator	Influencer	User	Decider	Purchaser	Passive Infl.
NB	0.679	0.735	0.606	0.614	0.614	0.882
kNN	0.659	0.740	0.630	0.603	0.651	0.865
SVM	0.692	0.748	0.728	0.599	0.724	0.910
RF	0.651	0.724	0.611	0.635	0.653	0.861

Table 2.2: Evaluation of Machine Learning algorithms to estimate six roles of shoppers in a group of people extracted from [107].

, kNN k-Nearest Neighbour Classifier[58] and RF Random Forests [18]). This work demonstrated how Machine Learning could contribute to estimate user preferences in proposed environment

This kind of imaging systems can be fulfilled with other sensing technologies. For instance RFID tags located in the shopping cart can be used in an analogous way that [70]. This setup allows estimation of the frequent path patterns that can be used to build an optimization model for the benefit of the establishment.

A local and more refined analysis for customers is also very relevant for this work, especially concerning Point of Sale (POS) where all purchases are finalised. Movements of customers in this area have been studied in this study [63]. Its contribution is focused on the proposal of the following values for every module of the store: speed, stops, duration of stay and number of visits. Simple Markov [81] chains are desired to model the transitions among those modules. Not only individual movements are analysed, contextual information is also inferred from the evaluation of the social relations among them. A new index is introduced in this work: see-buy rate, which refers to an approximate probability to purchase a specified commodity for customers, the index can be calculated through integrating the shopping paths data and transactions data. This index provides very useful information for retailers, but we believe it could be refined further to include features related to interactions of consumers with products.

2.4.2 Health monitoring

Prior studies brought to light in [52] have found that monitoring movements of one person through the home allows for both prediction of acute activities and longer-term assessment of changes in daily activities that may indicate important health changes [99]. Furthermore, continuous monitoring of mobility may allow for detection of acute emergency events such as falls [92]. An overview of location-sensing techniques has been described by Hightower and Borriello [49] as falling under one of three categories:

1. Triangulation, which uses multiple distance measurements between known points
2. Proximity measurements, detecting closeness to a known set of points
3. Scene analysis, which uses a measure of a view from a given perspective.

Each of these techniques can require the person being monitored to comply with wearing a particular monitoring device, or the technique can be passive (e.g., an imaging system) and not require any compliance on the part of the individual. We believe that a combination of both in a general architecture can benefit health professionals (and patients as well) to gather more information and improve the quality of life of patients with a personalized treatment.

2.4.3 Human pedestrian tracking

This subsection is focused on the assessment of the state of the art for algorithms related to human pedestrian tracking in a video sequence. Visual tracking is a hard problem since it may have a proper response under many different and varying circumstances upon the features of the multiple scenarios where it can be used.

A straightforward application of target tracking is surveillance and security control, provided by video surveillance systems. The main critical aspect with a great influence on the accuracy of the algorithms that will be explored is handling variations in illumination. The scope of this work is limited to indoor scenarios, therefore those variations are limited as well, but elements such as shadows or natural light entering through windows affect this matter. Those elements and the motion of the subjects may provoke changes in their appearance and viewpoint that should be considered.

An interesting survey on this topic is provided in [116]. This work proposes the following classification for tracking algorithms:

1. **Tracking Using Matching.** The trackers perform a matching of the representation of the target model built from the previous frame.
 - (a) Kalman Appearance Tracker [84]. The target motion is modeled by a 2-D translation at a single scale and searched in an area around the previous target position. In the subsequent frame, candidate windows around the predicted target position are reduced and compared to the template. This technique is adopted in our solution and will be explained in more detail in Section 4.4.2.3.
 - (b) Mean Shift Tracking [23]. The tracker performs histogram matching rather than using any spatial information about the pixels, making it suited for radical shape changes of the target. For each new frame, the tracker compares candidate windows with the target region on the basis of the Bhattacharyya [15] metric between their histograms. This technique is adopted in our solution and will be explained in more detail in Section 4.4.2.3.
2. **Tracking Using Matching with Extended Appearance Model.** The idea of this class is to maintain an extended model of the target's appearance or behaviour over the previous frames. However, this comes at the expense of having to search for the best match both in the image and in the extended model of appearance variations.
3. **Tracking Using Matching with Constraints.** Due to major successes for sparse representations in the object detection and classification literature, this kind of tracking algorithms reduce the target representation to a sparse representation, and performs sparse optimisation.
4. **Tracking Using Discriminative Classification.** A different view of tracking is to build the model on the distinction of the target foreground against the background. It is implemented by building a classifier to distinguish target pixels from the background pixels, and updating the classifier by new samples coming in.
 - (a) Tracking, Learning and Detection [56]. It uses labeled and unlabeled examples for discriminative classifier learning. The method is applied to tracking by combining the results of a detector and an optical flow tracker. This method has been used as a baseline solution and its adoption is described in Section 4.5.6.

5. **Tracking Using Discriminative Classification with Constraints.** This category of visual tracking confronts the difficulty in precise sampling by confining it to the target area. Its aim is to pursue the correct classification of pixels which is different from finding the best location of the target.

- (a) Kernelized Correlation Filters [46]. KCF is a tracking framework that utilizes properties of circulant matrix to enhance the processing speed. Data matrix is diagonalized with the Discrete Fourier Transform [119] (DFT), and linear regression is formulated analogously to a correlation filter. For kernel regression, a new Kernelized Correlation Filter (KCF) is derived by combining the results of a detector and an optical flow tracker. This method has been used as baseline solution and its adoption is described in Section 4.5.6.

We can differentiate between two kind of algorithms for human/object tracking:

1. **Offline.** Some information is provided to initialize the algorithm regarding location at a time t of the object of interest. We have tested its performance as a baseline solution to estimate its results in the proposed scenarios.
2. **Online.** Prior features learned from the object of interest (e.g., Machine Learning based human detection). Our proposed solution can be categorized in this group.

Offline algorithms are usually employed for object tracking. For multi-target human tracking online algorithms are the most suitable. In past years, different studies combined discriminative and generative methods for tracking [19] [72]. In these studies, offline trained detectors and standard tracking techniques are combined, and the detectors are used in a tracking model based on appearance/observation. Some methods solve the problem by proposing a global optimization problem, integrating in that optimization response of the detectors, and models concerning motion and appearance [87] [12]. In others (such as ours), a data association problem is formulated to link detections through time [114] [13]. Online systems are built on some typical models:

1. **Background subtraction.** The aim of this model is to segment the area(s) of interest.
2. **Detection.** This model aims to accurately describe the appearance of the area(s) of interest.
3. **Tracking.** Last model is intended to provide a definition for the motion of the area(s) of interest through predictions based on a probabilistic inference.

The breakdown of these models and relations among them can be noted in the diagram of fig. 2.6. It can be observed from the diagram how detection and tracking procedures are constantly updated and related to each other.

Figure 2.6: Most important components of trackers extracted from [116].

In [8], weak classifiers are trained to discriminate between pixels on the target vs. the background. Online AdaBoost [35] is used to train a strong classifier. Meanshift is applied on the classifier’s confidence map to localize the target. In this approach, mean-shift tracker is adopted, but the appearance model consists of a pool of templates found by the pedestrian detector. The template ensemble is updated over time adding or removing templates; however, as the templates are found by the pedestrian detector and not by the tracker itself, the templates are independent of each other. Appearance of each template is not updated over time with the new detections, limiting the noise introduced in case of false detections. We consider that during an occlusion the appearance model may provide little information about the object position and the pedestrian behaviour may be unpredictable during the gap. Therefore, a different approach is taken by adaptively growing the search area of the tracker to enable reacquisition of the tracker when the occlusion ends. In our case we are using an overhead camera which enlarges the area covered, and reduces the occlusions.

We have chosen a fast online method to track multiple pedestrians [141], this method uses the meanshift tracker, which searches for the detection whose appearance best matches the target via a gradient ascent procedure. In [23], Mean-Shift tracking is coupled with online adaptive feature selection. Features are selected to separate the target from the background; the most discriminative features are used to compute weight images. Then the Mean-Shift tracker is applied to each of the weight images to localize the target. Based on the new target detection, foreground/background samples are selected to compute a new set of discriminative features.

More recently, deep learning solutions has been employed for this problem. This kind of solutions for online tracking divide the problem in human detection and human tracking. Human detection is solved in [124] by implementing a part-based pedestrian detector, for each of the six parts (head, body and limbs) an independent Convolutional Neural Network (CNN) classifier is trained. Authors rely on the fact that occlusions may have various patterns depending on the part of the body occluded, and therefore an extensive part pool containing various semantic body parts is built to have different responses for every one. Other methods such as [125] rely on high-level annotations to infer those semantic properties. A novel task-assistant CNN is proposed to jointly learn pedestrian classification, pedestrian attributes and scene attributes. Some other works such as [91] use CNNs to obtain domain-independent representations for visual tracking. The proposed architecture includes five hidden layers in the top of the pipeline (three convolution and two fully connected) shared for the different scenarios, and then two fully connected layers with one branch for every scenario in the training step. For the testing step, only the branch corresponding to the scenario is employed, and the network is updated (tracking) by the negative samples (short-term period) and the positive samples (long-term period). Another remarkable solution for human/object tracking is presented in [78]. In this case a hierarchical approach is implemented to track the target region (offline tracking) by adaptively learning correlation filters on each convolutional layer. The convolutional feature maps are resized to a fixed larger size with bilinear interpolation to encode target appearance, and the outputs of each convolutional layer are used as multi-channel features in the frequency domain to calculate the correlation response maps that determine the target location for every frame.

Deep learning solutions presented for online tracking are complex in terms of HW resources, and although efficiency measurements are not provided by this work, we should take into account that trajectory estimation is usually implemented in real-time for retail and health monitoring environments. In addition, deep nets for human pedestrian tracking are more suitable for environments that owns a large variance in terms of lightness, occlusions or scale. For this dissertation we have delimited human pedestrian tracking for the two use cases presented, and therefore we have discarded this kind of solutions.

2.4.3.1 WSN and multi-sensing tracking

Visual tracking algorithms previously showed offer a notable performance regarding efficacy, but they can fail during some specific cases due to illumination changes and occlusions. Taking this into account, nowadays new tracking algorithms have been proposed by using different sensing acquisition solutions. There are two main steps in this type of algorithms:

1. **Signal processing.** This step is focused on the proper acquisition, synchronization and pre-processing of the different sensing devices involved.
2. **Communications.** This step is focused on the way that the intermediate nodes connected to the sensing devices interact with each other.

First studies related to this kind of tracking are summarized in [28]. This work proposes a very interesting classification among the steps previously introduced and the different techniques that can be implemented to cover them. The classification proposed is presented in fig. 2.7

Figure 2.7: General Classification for Target Tracking in Wireless Sensor Network exposed in [28].

More recently research in this field such as [61] has been proposed to fuse information from different types of sensing devices. The block diagram of the solution proposed can be explored on fig. 2.8. From the diagram it can be noted that the signal provided by a laser range finder is used to implement a Mean-Shift algorithm whose output is combined at the tracking level with the output of a Particle Filter [95] relying on the sensing input of an RGB [123] camera. The algorithm performs human and object tracking at the same time with interesting results. In our case we believe that since the probabilistic motion model on top of the visual tracking solution is Kalman filtering, the multi-sensing solution proposed should be classified as collaborative signal processing and more concretely data filtering performed by this filter. For that reason we have chosen the solution proposed in [11] to fuse information from visual and other natures of sensing devices. This work implements a Wireless Sensor Network based on adaptive fingerprinting and multi-sensing fusion to perform human localization and tracking, therefore it matches requirements for a precise multi-person multi-sensing solution for indoor tracking.

Figure 2.8: Block diagram proposed in [61].

2.5 Facial analysis and pose

The main static features expressed by humans are related to the facial properties, and both head and body pose. In this section the main contributions in those areas for the use cases presented are inspected to complete a full view regarding the signal processing techniques that can contribute to perform human behaviour modeling. In the retail domain we have focused the literature review in the feature of the consumers, and for health monitoring it is oriented towards the detection of the symptoms.

2.5.1 Features of the consumers

The main target of a retail establishment is to attract the attention of the consumers and evaluate their satisfaction with the products and services provided. To measure that satisfaction different methods relying on different properties have been implemented. All the solutions explored for that aim implement a preprocessing step focused on the localization and normalization of the ROI, where the object of analysis (face of the subject in our case) is located. Güven et al provide an innovative application in this field for audience measurement [57]. What they assume is that the face occurrences are limited in this scope and one classifier is able to capture all of them by the combination of MBLBP (Multi-scale Block Local Binary Patterns [74]) and Gentle Boost [35] in order to perform face detection for extreme poses as well. This publication confirmed that real-time performance can be achieved without losing a significant rate of efficacy to count people in a limited area such as POS. This achievement is raised by implementing a classifier with a constant window size to find faces, and the reduction of the number of weak classifiers. Khryashchev et al [59] propose an interesting block diagram for video analysis with audience measurement purposes expressed in fig. 2.9

We find the workflow proposed for local analysis significant, however we should also mention the lack of upper abstraction levels that provide more contextual information regarding global features of the establishment.

2.5.2 Symptoms detection

A complete home-based solution for motion video capture technique is tested in [85]. Authors study motor function in humans by assessing joint angles, position, velocity, and acceleration of a limb in 3D. In the experimental study, Videoanaliz Biosoft 3D complex (Biosoft Ltd., Moscow, Russia) is used [115]. For that aim authors build a testing environment involving the next devices: IR backlight, Test object and a set of self adhesive spherical light returning elements. The software performs automated identification of labels on video, computation of their coordinates

Figure 2.9: Block diagram for the standard audience measurement solution of [59].

in 3D, construction of a 3D model of a subject and kinetogram. They assume the following PD symptoms are the most evident and cardinal from the point of view of home-based healthcare:

1. Muscular rigidity
2. Brady- and akinesia
3. Rest tremor
4. Postural instability

They propose a model that follows the M3 architecture [128] for smart spaces, supporting service intelligence in this healthcare scenario for motion video detection.

It was previously stated that the symptoms associated to PD are highly correlated to the pose of the patient. The first work explored in this direction [113] presents a method to combine a Gaussian Process Regression [106] (to determine the contribution of discriminative model and initialize the model) and a Particle Filter [95] to track the human pose in video sequences. The initial step is to build the discriminative model that allows a matching procedure with the silhouette descriptor to define the corresponding human pose. The second of them [102] uses depth sensors to extract skeleton features based on video analysis. They use the Euclidean distances between the centroids of legs of the patients and estimation of the stride length from maximal values of these differences to characterize the gait of a Parkinson patient. Our work proposes and offers results on a real scenario for the first step of this kind of analysis (human tracking), establishes a proper architecture for the deployment of these type of sensing solutions and a data model to store information inferred for the second one of them.

Finally another evolved sensor-free vision-based solution [6] is proposed for gait analysis by capturing twenty videos from PD patients uploaded to Youtube, its block diagram is shown in fig. 2.10.

Figure 2.10: Block diagram proposed in [6].

Initially they rely on the real-time 2D pose algorithm shown in [21] to extract the skeleton information such as the position of the ankles or knees (the most relevant for gait analysis) of every subject. Unfortunately, in normal video recordings, the device may not be placed in a single location for the entire duration of the recording, and therefore it is subject to continuous changes in viewing angles. For that reason a normalization procedure is required to process the frames composing the videos. It consists of projecting the frame of reference of every patient coordinates onto the camera's frame of reference. For this procedure, the viewing angle of the camera needs to be known, and it is calculated by inferring the head pose of the subject using DeepGaze [98] to identify the yaw of the head with respect to the camera. We believe head pose is a key feature for cognitive patients analysis and for that reason we have explored this field using 3D information. With the yaw information, a camera matrix is automatically populated by the proposed system. The process is summarized in fig. 2.10

Figure 2.11: Diagram showing the method followed in DeepGait to normalize the images.

The knee and ankle positions for the right and left limbs and the head position in every video frame are utilized to analyse the three prominent features of the Parkinsonian Gait: shuffling steps, slow gait and gait asymmetry. To model previously named features a model based on prior positions is built, results have shown great differences between those scores for Healthy Gait and Parkinsonian Gait.

Another solution explored [29] uses Microsoft Kinect sensor to track the limb and neck movements of a patient performing two motor tasks. By implementing a new motion segmentation algorithm, kinematic features were extracted from the videos and classified using SVMs. The depth channel is used to extract the virtual skeleton, which consists of the positions of 20 anatomical landmarks of the human body (which we will refer to as 'joints' for simplicity). The movement of all joints except the 3 joints that participated directly in the performance of the task were used to build the feature vector based on length, quantized length distribution and speed extracted from the motion segments. Two tasks were tested with features extracted: binary classification of the dyskinetic condition (present or not), and quantified assessment of Dyskinesia severity and its agreement with the evaluation of clinician professionals. Results for dyskinetic classification were evaluated by using Area Under the Curve (AUC) [89] and General Correlation Coefficient (GCC) [55] and they are shown in table 2.3. Results gathered reveal a high correlation between clinical symptoms of the patients and current accuracy of available techniques to estimate them.

Another notable work [71] goes further in the Machine Learning algorithms, and simplifies the data acquisition. The proposed method uses Convolutional Pose Machines (CPM) [134], composed by CNNs to predict pose by iteratively refining joint predictions, and apply it to each video frame. This work demonstrates the hability of CNNs to perform visual tasks significant for symptom inference. Head and neck annotations from CPM were used to initialize a face bounding box, which was tracked by using Multiple Experts using Entropy Minimization (MEEM) object

Feature	Average AUC	Average GCC
Average motion length	0.882	0.805
Motion length distribution	0.862	0.703
Average motion speed	0.906	0.789

Table 2.3: Evaluation of features for dyskinetic condition from [29].

tracker [140]. With this method the following joints are located:

1. Left and right shoulders
2. Left and right elbows
3. Left and right wrists
4. Left and right hips
5. Left and right ankles
6. Trunk, Neck and top of the head

The severity of Dyskinesia was rated using the Unified Dyskinesia Rating Scale (UDysRS). The communication and drinking task UDysRS ratings contained seven subscores, however the face subscore was omitted, we believe that information could provide a relevant improvement in this field. A total of 32 features were extracted for each joint, and their influence on LID was evaluated by Spearman Correlation [90]. The communication task had higher correlations than the drinking task. The top correlations for subscores of the communication task were achieved by the trunk, right arm and neck (face not evaluated).

2.5.3 Facial analysis

We have already exposed the influence of facial properties for both use cases included in this dissertation. Concepts such as visitors characterization, patients head pose or focus of attention have been previously mentioned. In addition, importance of depth sensing devices has also been proved by different works in order to acquire 3D visual imaging properties, and therefore they shall be included in the architecture proposed. This section is intended to describe current algorithms related to facial analysis by using Computer Vision techniques with special emphasis on depth sensing devices and 3D reconstruction. This topic has been widely covered, with different applications such as face recognition, age or gender estimation, gaze prediction or emotions analysis. In our case we will focus on two tasks that we believe are the basis for most of the facial analysis applications: facial landmarks detection and head pose estimation.

Facial landmarks, also known as facial feature points or facial fiducial points are the points which possess most of the meaningful information regarding the different properties that are expressed by human beings in the frontal region of their heads. Facial feature points are different from keypoints for image registration [97], keypoint detection is usually an unsupervised procedure and facial landmark detection uses to be a supervised procedure. Landmark detection usually starts from a rectangular bounding box returned by face detectors such as traditional ones like [129] which implies the location of a face. This bounding box can be employed to initialize the positions of facial feature points. In our case we have used a face detector built upon the Histogram of Oriented Gradients (HOG [27]) feature descriptor, combined with a Support Vector Machine (SVM [24]) classifier.

The first noteworthy advance in the modern times regarding this subject was achieved by the publication of a extensive database in [39]. Authors claim that facial appearance varies significantly with a number of factors, including identity, illumination, pose, and expression. To support the development and the comparative evaluation of face recognition algorithms, it is important the availability of facial image data spanning conditions in a carefully controlled manner. Therefore, authors collected a database under different conditions for the factors with a large influence on facial appearance previously mentioned. They built a setup where 13 cameras were located at head height. Two additional overhead cameras were located above the subject, simulating a typical surveillance camera view. A summary regarding the variation of different elements for data capture is expressed on fig. 2.12

Figure 2.12: Variance of visual appearance factor for data collection of [39].

Multi-PIE database contains 337 subjects, imaged under 15 view points and 19 illumination conditions in up to four recording sessions. This work provided the foundation for the benchmark [108] that has established the main metrics and standards to normalize the evaluation of the different methods implemented for this task. One of the standards established by this work is the annotation of the facial landmarks, this can be observed on fig. 2.13:

Figure 2.13: Landmark annotation established by [39].

2.5.3.1 3D facial acquisition

Recently different approaches have been published to optimize acquisition systems capable to represent accurately 3D facial attributes. Ref. [32] collected 2000 2D facial images of 135 subjects as well as their 3D ground truth face scans. The authors proposed a dense face reconstruction method based on dense correspondence from every 2D image gathered to a collected neutral high resolution scan. Another hybrid solution (Florence Dataset) for reconstruction is presented in [9], but in this case a complex capturing system (3dMD [5]) is required, and the number of

subjects (53) is also smaller than the one proposed here. UHDB31 [66] presented an evolution of this work by increasing the number of poses and number of subjects which resulted in a more complete dataset. Costs of the set up are quite high and the number of subjects are still lower than ones captured in this work.

In our case we provide a neutral high resolution scan as well, but we believe that the impressive results of recent works published implementing deep learning for classification and segmentation with normalized input clouds [103] [73] motivate a new research line. And therefore we postulate a new challenge, and propose an initial reconstructed and normalized set to adopt this line for facial analysis.

Pandora dataset focused on shoulder and head pose estimation is introduced in [17] for driving environments. But this dataset only contains images from 20 subjects and they only have one camera which does not allow precise 3D reconstruction for extreme poses. Ref. [60] proposes a 3D reference-free face modeling tested on a set of predefined poses. The authors perform an initial data filtering, and employ face-pose to adapt the reconstruction process. In our case we use 2D face detection projection, and afterward implement the proper 3D filtering techniques, exploiting information from 2D facial landmark detection in order to perform a more reliable 3D face reconstruction. Other techniques have been proposed by using just a single RGB sensor, but in this case they require either a 3D Morphable Model (3DMM) initially proposed by Blanz and Vetter [16], which can be trapped in local minimum and can not generate very accurate 3D geometry, or a 2D reference frame and displacement measurement, like in [136].

2.5.3.2 Facial landmark detection

Ref. [132] makes a good analysis concerning this topic. In this work methods analyzed for face landmark detection are classified in the following groups:

1. **Constrained Local Model-Based Methods.** Constrained Local Model-Based Methods (CLM-based methods) fit an input image for the target shape through optimizing an objective function, which is comprised of two terms: shape prior $R(p)$ and the sum of response maps $D_i(x_i; I)$, ($i = 1, \dots, N$) obtained from N independent local experts [111]

$$\min_p R(p) + \sum_{i=1}^N D_i(x_i; I). \quad (2.1)$$

A shape model is usually learned from training facial shapes, and it is taken as the prior knowledge to refine the configuration of facial feature points. Each local expert is trained from the facial appearance around the corresponding feature point, and it is employed to compute the response map which measures detection accuracy.

$$s = s_0 + P_s \alpha = s_0 + \sum_{i=1}^n \alpha_i s_i \quad (2.2)$$

Where s_i ($i = 0, \dots, n$) can be estimated by the Principal Component Analysis (PCA) on all aligned training shapes. In addition, s_0 is the mean of all these shapes and $s_1, \dots, s_n$ are the eigenvectors corresponding to the n largest eigenvalues of the covariance matrix of all aligned training shapes.

2. **Active Appearance Model-Based Methods.** An Active Appearance Model (AAM) [37] can be decoupled into a linear shape model and a linear texture model. The linear

shape is obtained in the same way as in the CLM framework . To construct the texture model, all training faces should be warped to the mean-shape frame by triangulation or thin plate spline method; the resultant images should be free of shape variation, called shape-free textures. Each shape-free texture is raster scanned into a grey-variation, z_i is normalized by a scaling u and offset v :

$$a = \frac{z_i - v\vec{1}}{u} \quad (2.3)$$

Where a is a texture representation in the reference frame, u and v represent the variance and the mean of the texture z_i respectively and $\vec{1}$ is a vector of all 1s with the same length as z_i . The texture model can be generated by applying PCA on all normalized textures as follows:

$$a = a_0 + Pa\alpha\beta = \sum_{i=1}^m a_0 + \beta_i a_i \quad (2.4)$$

Where a_0 is the mean texture in the reference frame, Pa is the texture projection matrix in the texture model of AAM, $\alpha = (\alpha_1, \dots, \alpha_N)^T$ are shape parameters in the Point Distribution Model (PDM) and $\beta = (\beta_1, \dots, \beta_N)^T$ are the texture parameters in the texture model of AAM.

The coupled relationship between the shape model and the texture model is bridged by PCA on shape and texture parameters:

$$\begin{pmatrix} W_s \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} W_s P_s T(s - s_0) \\ P_a^T(a - a_0) \end{pmatrix} \begin{pmatrix} Q_s \\ Q_a \end{pmatrix} \quad (2.5)$$

Where W_s is a diagonal weighting matrix measuring the difference between the shape and texture parameter, P_s is the shape projection matrix in PDM, s is a shape represented in the reference (mean-shape s_0) and the appearance parameter vector c governs both the shape and texture variation (Q_s and Q_a)

3. **Regression-Based Methods.** The aforementioned categories of methods mostly govern the shape variations through certain parameters, such as PDM coefficient vector α and AAMs. By contrast, regression-based methods directly learn a regression function from image appearance (feature) to the target output (shape):

$$M : F(I) \rightarrow x \in R^{2N} \quad (2.6)$$

Where M denotes the mapping from image appearance feature ($F(I)$) to the shape x , and F is the feature extractor.

Ref. [20] proposes a two-level cascaded learning framework based on boosted regression. This method implements a regression function that directly learns a vectorial output for all landmarks. Shape-indexed features are extracted from the whole image and fed into the regressor.

4. **Graphical Model-based Methods.** Graphical model-based methods mainly refer to tree-structure-based methods and Markov Random Field (MRF) based methods. Tree-structure-based methods take each facial feature point as a node and all points as a tree.

The locations of facial feature points can be optimally solved by dynamic programming. Unlike the tree-structure which has no loop, MRF-based methods model the location of all points with loops.

Zhu and Ramanan [143] proposed a unified model for face detection, head pose estimation and landmark estimation. Their method is based on a mixture of trees, each of which corresponds to one head pose view. These different trees share a pool of parts. Since tree-structure-based methods only consider the local neighbouring relation and neglect the global shape configuration, they may easily lead to an unreasonable facial shape.

5. **Deep Learning-Based Methods.** Luo et al [77] proposed a hierarchical face parsing method based on deep learning. They recast the facial feature point localization problem as the process of finding the label maps. The proposed hierarchical framework consists of four layers performing the following tasks respectively: face detector, facial parts detectors, facial component detectors and facial component segmentation.

Sun et al [120] proposed a three-level cascaded deep convolutional network framework for point detection in a coarse-to-fine manner. It can achieve great accuracy, but this method needs to model each point by a convolutional network which improves the complexity of the whole model. Ref. [50] improved the detection by following a coarse-to-fine manner where coarse features inform finer features early in their formation, such that finer features can make use of several layers of computation in deciding how to use coarse features. We have selected this method to test the data augmentation method presented in subsection 5.2.2 because of the novelty and efficiency of a deep net that combines convolution and max-pool layers to train faster than the summation baseline and yields more precise localization predictions.

Finally, the other selected solution to test the performance of the proposed augmentation method in subsection 5.2.2 is [137] as they imply an evolution from previous models. The proposed Tweaked Neural Network does not involve multiple part models, it is naturally hierarchical and requires no auxiliary labels beyond landmarks. They perform clustering of representations produced at intermediate layers of a deep CNN trained for landmark detection, showing surprisingly good results for the classification of different head poses and (some) facial attributes. They inferred from previous analysis that the first fully connected layer already estimates a rough head pose. With this information they can train pose specific landmark regressors.

2.5.3.3 Head pose estimation

Head pose estimation is a topic currently being widely explored, with applications such as autonomous driving, focus of attention modeling or emotion analysis. Fanelli et al [31] introduced the first relevant method to solve head pose by relying on depth sensing devices. Their proposal is based on random regression forests by formulating pose estimation as a regression problem. They synthesize a great amount of annotated training data using a statistical model of the human face. In an analogous way, Liu et al [76] also propose a training method based on synthetic generated data, but in this case they use a CNN to learn the most relevant features. To provide annotated head poses in the training process, they generate a realistic head pose dataset by rendering techniques, in which they fuse data from 37 subjects with differences in gender, age, race and expression. Some other lines of research such as the one followed by [98], pose the problem as the classification of human gazing direction, an approach that we have followed for our work. It proposes deep learning techniques to fuse different low resolution sources of visual information that can be obtained from RGB-D devices. The authors encode depth information

by adding two extra channels: surface normal azimuthal and surface normal elevation angle. Their learning stage is divided in two CNNs (RGB and depth inputs). The information learned by deep learning is employed to further fine-tune a regressor.

Analyzing all literature related to our work, we can conclude that our multi-camera RGB-D setup provides an affordable capturing system which is wider enough to perform 3D face reconstruction of extreme poses implying a reasonable cost and deployment. In addition facial landmark has already been widely studied, therefore in this topic it is more appropriated to provide a validation method related to 3D reconstruction techniques being presented here as well. Head pose estimation is highly correlated with facial landmark detection (specially in 3D domain) and we believe with a good performance in landmarks head pose could easily be solved.

2.6 Conclusions

This chapter has explored a wide spectrum of architectures, solutions and algorithms to provide the reader a state of the art of the topic of this dissertation. There are two main fields where conclusions have been obtained:

1. **Architecture.** We could not find relevant publications for retail environment describing a global architecture including different kind of sensing devices. For that aim we have addressed one publication [104] including a logical architecture for this domain. Analogously, we find very relevant publications in the domain of health monitoring for human-machine interaction or platforms to exchange information between clinicians and patients or care-givers, but we could not find a concrete deployment of different kind of sensing devices for the corresponding scenarios in order to automatically infer behavioural properties of the patients. We have addressed this issue by another publication [7], that is complemented with a full integrated care platform here [122]. Both physical and logical architectures proposed for retail and health monitoring domains will be detailed in the next chapter upon the requirements and scenarios defined in the same chapter as well.
2. **Computer Vision & Machine Learning.** We find very relevant a further development for human pedestrian tracking, since it is one common task for both use cases (and for most of the domains where human behaviour is analysed). This area has been widely studied, but the lack of more detailed results at the scenarios included in this dissertation are another reason to perform a visual imaging system design, implementation and deployment. An interesting approach for health monitoring following this line can be found on [47].

Analogously, facial properties are one of the most studied topics in visual imaging, but we could not find any 3D available dataset which is in proper shape to optimize some deep nets architectures. And we believe it could be the smile to get further in the analysis of this kind of properties for health monitoring. Pose properties have already been exploited in this area, but we believe further studies of facial properties could add a great value. Another reason to focus research in this area is to enhance human characterization with depth sensors that are widely used in the retail domain. Unfortunately, due to privacy restrictions is very difficult to capture this kind of data in the real scenarios linked to the use cases proposed for this dissertation. Therefore, we have captured them under controlled environments and we have made publicly available the corresponding dataset and the techniques employed to gather and validate it through this publication [105].

-

Chapter 3

ARCHITECTURE FOR MODELING HUMAN BEHAVIOUR

This chapter is intended to describe the different components involved and their relationships in order to develop a system able to model human behaviour in indoor scenarios. The scope of this work is mainly focused on signal processing techniques but other areas such as interfaces or data modeling are relevant for this topic, and therefore they are also considered in the architecture proposed. The following diagram shows the architecture designed:

Figure 3.1: Diagram showing the general architecture proposed for the problem stated.

It can be noted that the signal processing tasks proposed are divided between dynamic and static. The first is only concerned with spatial analysis of the data captured by the corresponding sensing devices, and the second considers time properties. Static features are grouped in pose and

facial analysis. Body pose provides information regarding location and orientation of the different parts of the human body, and head pose regarding the specific part where humans express the most relevant features to analyse their current state. This estimation is highly significant and it is connected to facial analysis because it is a complementary task of facial landmark detection. The orientation and correlation of the most characteristic points of human faces provide information with great potential to evaluate their current state. Locations of those points are the basic features used for most of the facial analysis tasks such as age/gender estimation, head pose/gaze estimation or emotion estimation.

Dynamic features are grouped in two blocks: motion and interaction. For the first of them we find trajectories described by human pedestrians the most relevant of the tasks. It is also a complementary task for static algorithms since it provides the location of the persons in the scenario at a certain time, which allows the local analysis of the corresponding region of interest. Interaction blocks provide the information of how humans relate within the different parts of the scenario. In this case, we believe that the meaning of the interaction is significantly dependent on the application field, and therefore we have classified the interactions depending on their nature (human-device, human-object and human-human)

Outputs of all these tasks are stored in a knowledge base designed to cover the most relevant information for a certain scenario. We consider this area highly dependent on that scenario and for that reason its structure has not been provided in the proposed diagram, but it has been defined for the two use cases explored that will be defined in the next sections: retail and health monitoring. In the same fashion user interfaces capture information from the signal processing tasks to show their estimations to the user, also to interact with the knowledge base to provide data regarding its usage (and give feedback to the system regarding the interests of the user), and finally to expose more accurate data to the user after the application of the corresponding Data Science algorithms.

3.1 Retail

This section introduces a use case to improve the quality and efficiency of the retail establishments in order to increase their attractiveness [104]. The architecture proposed goes through relevant tasks from a logical and physical perspective to infer properties of consumers related to their behaviour, mostly based on visual imaging systems but also taking into account signals captured by other sensing technologies. That inference adds a great value for the retailers, so that they can adapt the distribution and appearance of their products to facilitate their goals and simplify the campaigns directed to the purchasers.

3.1.1 Introduction

The traditional brick-and-mortar shops are actively looking for completely new ways to attract customers. To achieve that objective valuable insights should be provided to help clients meet their goals through introducing new ways to understand the movements of the customers and their intentions to offer more personalized services. The trend is exponentially increasing for brands and network aggregators, and needs of media planners are moving towards inferring the level of engagement of customers in order to attract them to new products that satisfy their needs. The ambition is to focus on the issues at hand unbiased opinions, and on the steps necessary for success. Future stores should offer what is relevant for the customers at the time they demand it.

The solution proposed should be able to infer all features from shoppers and different actors should be defined for that aim. The system should capture different situations and adapt its

response for every one, they are specified on the use case diagram of fig. 3.2.

Figure 3.2: Diagram showing the actors and their interaction with the system for the retail use case.

Four roles can be observed in the use case diagram: retail users, retail staff, retail management and system administrator. The first is the main case of study, and the system captures data related to their visits to the store, their interactions with the different products offered there and the products purchased by them. The tool provided by the system helps the workers of establishment as well. For the regular staff, they can receive alerts when it is the time to adapt the distribution of one section to the preferences of the shoppers, or when the system detects long queues in the available POSs and a new one should be opened. Management staff is able to check the data provided by the system on the dashboard, or they can update the available data regarding the distribution of the different sections in the store. Finally the system manager should update the configuration of the sensing equipment in order to optimize its performance.

3.1.2 Scenario

Automatic visual data analysis is a very challenging problem. In order to detect humans in video streams and automatically infer their features, interactions or intentions based on their behaviour, different computer vision algorithms are required, very often combined with Machine Learning techniques. The first goal of video analytics in the proposed area of research is to track all the movements of the people that enter inside one store. The correlation between time and space for all the customers on their shopping experience should be accurately stored by the system. Location of potential costumers allows the system to establish a higher abstraction level and provide deeper assessments related to the impact, emotions or synergies incited by the properties of the store and the products displayed for the shoppers.

Once the roles and requirements for the system have been defined, the next step is to define a proper deployment. For that aim one sample image has been annotated and it is shown on fig. 3.3.

Figure 3.3: Diagram showing a sample with the basic interactions captured by the system on a real scenario.

Five kinds of inferences can be noted on the sample from real scenario:

1. **Identification.** The first task is to properly identify a shopper when they enter the shop in order to associate all the inferences captured to their id. For that aim one WSN captures the digital signature of the mobile device of the user, and since all the sensing devices are calibrated under the same scenario all the signals are able to identify that shopper.
2. **Tracking.** By fusing the signals captured by one overhead camera and the WSN, the system is able to determine all the trajectories associated with the path of the consumer.
3. **Product interactions.** Small sensing devices such as beacons or RGB-D cameras specifically located on the shelves being analysed are able to capture the interaction that one visitor has with the different products on display in the store. The first of these is steered towards direct contact between specific visitors and the products, and the second provides an estimation regarding the attention that the visitors pay to the different sections of the shelf based on their current location and head pose estimation.
4. **POS.** RGB-D cameras located in the POS are able to infer the number of visitors in the queue, and therefore optimize the performance of this important section of the retail establishment.
5. **Facial analysis.** The same devices previously mentioned provide a signal that allows the analysis of satisfaction and the characterization of the shoppers through facial analysis techniques.

3.1.3 Logical architecture

This subsection is intended to provide an intelligent organization for all the devices, processes, and tasks previously mentioned for the optimization of the performance of a retail store. The proposed logical architecture is presented in the fig. 3.4. Five layers can be observed in the diagram:

Figure 3.4: Diagram showing the logical architecture for retail.

1. **Data collection.** Initially data is collected through the sensing devices deployed in the scenario: overhead cameras, RGB-D cameras and a WSN.
2. **Pre-processing.** Previously mentioned sensing devices should be calibrated and synchronized in spite of providing accurate raw data for the next steps of the system. Non meaningful data is filtered and the sampling rate is fixed in favor of the maximization of the trade-off between efficiency and efficacy.
3. **Signal processing.** Identification and tracking are the two global modules identified in order to properly estimate the location of the visitors, and to provide a procedure to associate all features inferred to the corresponding person. A further three modules more specific shall run in the store:
 - (a) **Facial analysis (static inference).** This module provides an estimation of the degree of satisfaction of the users when they are about to leave the store.
 - (b) **POS (human-human interaction).** This module helps the store management to optimize the performance of the section where the cash registers are located.
 - (c) **Product interactions (human-object interactions).** This module estimates when the shoppers approach different shelves, and how they check the features of the available products. Therefore, it allows the management staff to obtain a more accurate measurement regarding the distribution of the shop.
4. **Big data.** Estimations from the previously mentioned signal processing algorithms are stored in a knowledge base layer with two different policies regarding update frequency: real-time or batch.
 - (a) **Real-time.** The most critical entities regarding efficiency (real-time) are located on this side to feed alerts for the staff of the establishment, and to respect privacy policies of the shoppers on the analysis of the whole establishment (video is not stored, just the locations and the digital signature). All those entities are processed in the sensing-hub (trajectories by the processing unit and the queues by the sensing server)

- i. Trajectories
 - ii. Sections density
 - iii. Queues
- (b) **Batch.** Batch information is more concerned with the distribution of the store and the optimization of sales (preserving security for billing information as well). In this case the modules located here are the heaviest computationally, and the most complex regarding efficacy. For the analysis of 3D facial 3D information, it would require high computation resources on the concrete scenario to process it, and therefore it is more suitable to send it to the back-end for its processing. It is programmed for the retail store to send it once the establishment has closed, the data is filtered (storing just the region of the head), divided in assets (to preserve privacy) and compressed (to reduce its size) on the sensing-hub environment. The following tasks are considered to be processed in batch mode:
 - i. Interactions
 - ii. Facial features
 - iii. Billing

- 5. **Services.** The dashboard is intended to show results estimated by the signal processing algorithms stored on the knowledge base, and also some higher level inference related to busyness intelligence executed on top of gathered data.

In this section we have shown some data containers without any structure since the diagram is highly oriented towards proper deployment and implementation details. A data model more oriented towards semantic/hierarchical structure of relevant data is detailed on Appendix A to complete the proposed solution.

3.1.4 Physical architecture

In this subsection the location of all the devices and its physical connection with the correlated elements of the system is explained. A graphical explanation can be found on the diagram of fig. 3.5.

Figure 3.5: Diagram showing the physical architecture for retail.

Two environments can be noted on the figure:

1. **Sensing-hub.** A private network (including WSN) is managed by the router and the processing capabilities are divided between a portable device (e.g., FPGA/Raspberry Pi) for the processes associated to the overhead camera, and a local server for the processes related to the RGB-D camera and the WSN. Trajectories (with the corresponding IDs), outputs of the queues optimization module, and the RGB-D data are sent to the Back-end.
2. **Back-end.** This environment hosts the knowledge base, and all the batch processes related to the sensing devices previously mentioned. Business intelligence processes are also located in this environment and all the information is included on a dashboard (including the consumer models) offered to the user through the interfaces provided by the system.

3.2 Health monitoring

The objective of this use case is to create an integral platform to support healthcare by providing them with relevant knowledge of the status of the patient [122]. The use case is mainly focused on elderly patients with illnesses related to cognitive diseases such as Parkinson or Alzheimer. A key step and the main focus of this section is the creation of a complete architecture to establish the basis of the data workflow and infrastructure developments concerning sensing deployment [7], data hosting and processing.

3.2.1 Introduction

As the population ages, it is important to develop unobtrusive, affordable and computationally inexpensive off-the-shelf smart home technologies to be used for elder-care, particularly for elderly

people experiencing health problems. Every patient has their own evolution, therefore it is needed to provide information so that health professionals and caregivers are able to analyse more accurate data in spite of improving treatments for the patients.

Different roles have been detected that interact with proposed system such as:

1. **Patient.** This work is focused on elderly patients (+65) that have been diagnosed with Parkinson, Alzheimer or any similar disease related to dementia.
2. **Caregiver.** The person who takes care of the patient. Three kinds of caregivers have been considered (depending on the relations between them and with the patient):
 - (a) Professional caregiver: A professional who is paid and lives with the patient or takes care of the patient by hours every day.
 - (b) Emotional caregiver: A relative who lives with the patient, and in many cases the partner who is also elderly too. Relative who lives with the patient but he/she works out of the house (son, daughter or the partner) or relative who lives near the patient and looks after the patient (a daughter or son normally).
 - (c) Logistical caregiver: He/She is a younger relative. He/She does not live with the patient but provides logistical support; for instance escorts them to the doctor or carries out administrative procedures (son, daughter or grandchildren). When there is no caregiver, emergency situations are reported to social services that will take on the role of a caregiver (with limitations regarding Welfare State and social help in every country).
3. **Professional.** In this case there are several health areas related to aforementioned illnesses. We have considered professionals with medical degrees (psychiatrist, psychologist, physiotherapist or general practitioner). Other related professionals such as rehab personal, nurses or social workers have also been taken into consideration for its design.

The functionalities associated to every one of them can be observed in the fig. 3.6.

Figure 3.6: Diagram showing the actors and their interaction with the system for health monitoring.

Functionalities previously mentioned should be understood in the scope of events and alerts defined for the system. For that aim these concepts are defined below.

1. **Trigger event:** The events, which are defined on the platform, must be measured in order to be detected and raise an alert. We have three types of events:
 - (a) Informational event: For instance clear diagnosis or medication (type and hours). Someone (normally a caregiver) must provide the initial information and updates. Regarding appointments with the doctor, someone (normally a caregiver) provides the information and there is just a reminder. One change (updated) in any of the previously mentioned matters will trigger an alert.
 - (b) Yes or no event: There is an event which must be detected. Either it is detected or not detected. For instance: Take medication: yes/no; go out the house: yes/no. In some cases the caregiver or professional would like to be reported when the event happens or when the event does not happen. Happening or not will be an alert. For instance: take medication, a caregiver could be alerted when patient takes medication or be alerted just when patient does not take medication.
 - (c) Measured event: Regarding circumstances that must be measured. For instance temperature, movement, understood as distance, length, speed, gravity centre. In this case we have to establish the alert level (derived consequences) at the moment when the event has been measured.(e.g., 38 degrees are measured for body temperature, consequences: person moves slower, person takes more time to do a normal activity).
2. **Alert:** At the moment when the event takes a measured value (yes or no, degree, slower speed, more time) an alert is generated. We have two kinds of alerts:

- (a) Online alerts. This kind of alert is in real time for emergency situations. For instance, when a patient has a temperature. An online alert is generated in order to resolve the situation. If there is no caregiver, an online alert (emergency situation) is reported to emergency services.
- (b) Offline alerts. This kind of alert is not in real time and a situation to report other professionals. For instance: Medication change.

It is important to bear in mind that one alert can be online for caregiver and offline for the rest of the professionals. The same event can generate two kinds of alerts having in consideration the corresponding role.

Analogously in this scope **Electronic Health Record (EHR)** should be defined as well: 'EHR means a repository of patient data in digital form, stored and exchanged securely, and accessible by multiple authorized users. It contains retrospective, concurrent, and prospective information and its primary purpose is to support continuing, efficient and quality integrated healthcare' [44]. In our proposal we have considered the following entities to be included in the record:

1. General clinical data.
2. Disease clinical data.
3. Social.
4. Health measurements.

3.2.2 Scenarios

The medical environment demands a personalized treatment taking into consideration different factors such as: the patient, the caregiver (there in or not), is the patient living alone?, current state of the disease (initial, moderate or severe) ... For the scope of this work scenario means physical spaces and physical conditions. For instance, a house with several rooms, or a flat with just one room (a kitchen in the same space or not), whether there is a corridor or not. Or if we are talking about a therapy room, a description of the spaces should be taken into consideration including some specific details such as the description of existing columns or the existence of metal materials, and its location.

Three kind of scenarios have been selected for the proposed architecture:

1. **Home:** This is the scenario where the patients spend most of their time, and also the one where the monitored area is smallest. Therefore, it is the scenario where the number of inferences that can be captured by the system is larger, and the biggest variability among them can be found. For that reason this is the scenario where the sensing equipment and its deployment is the most complex.
2. **Hospital:** This scenario is where most of the medical decisions take place, the one that populates the largest area and where policy constraints are the most restrictive. The patients in this scenario are usually in a special state.
3. **Rehabilitation center:** This is a very specific scenario where the patients have periodic sessions to perform concrete exercises that improve their mobility and that should be monitored by the system to measure their progress.

The next figure shows the roles, interfaces and sensing devices involved in every scenario:

Figure 3.7: Diagram showing the distribution of the sensors among the scenarios included in health monitoring.

Regarding roles it can be observed that the caregiver is only present in the home, and the medical professionals are present in the other two scenarios. It should be taken into account that in the rehabilitation center the profiles are very specifically related to physiotherapy and social areas, and in the hospital we can find all the profiles previously mentioned. We have also added the user administrator role for all the scenarios, this role is in charge of optimizing the performance of the system. Patients are the principal subjects in the system and therefore they are present for all the scenarios.

Regarding inferences related to sensing devices, the home is the simplest scenario to deploy them, and therefore is the one where most of them can be found. Body pose is missing because its contribution is mainly focused on the exercises that the patients perform in the rehabilitation center. Facial analysis is highly related to the smart tv interface, because when the patients are interacting with the TV they fulfill environment requirements needed to perform a proper analysis of this area. Finally web interface is mostly directed to medical professionals in order to annotate or check the data of the patients and mobile interfaces are directed towards patients and caregivers.

3.2.3 Physical and Logical architecture

This subsection defines the physical and logical architecture of the system. Different scenarios have been presented in the previous subsection, but in this case we present a scalar architecture with all the relevant elements of the system that can be adapted to any of them. Finally on Appendix B a semantic/hierarchical data model where all data gathered is stored will be presented. Fig. 3.8 shows all the elements of the system and connections among them. From the diagram it can be easily observed that in the proposed architecture two different environments coexist: the sensing hub and the back-end.

1. **Sensing hub:** This environment is orchestrated by the sensing server that manages all the I/O operations from the private network built to manage all the data transfers concerning the different devices involved and transfer its results to the global environment (back-end). In this case the package sent to the back-end overnight contain the processed trajectories,

skeleton and 3D assets contained in the head of the visitors and the constants measured by the health bracelet. Different layers are required to capture the data from the corresponding sensing devices, synchronize that data, process it to obtain annotations for the tasks previously presented and connect the different devices involved in this process. They are detailed below:

- (a) **Intelligent devices:** This is the highest layer in the abstraction of the local environment and it is focused on the connection among the devices (implementing corresponding data flow), the processing of the data captured and finally transmission of the results.
 - i. **Sensing Server:** The local server is connected through an USB to the RGB-D camera in spite of capturing corresponding 3D data and performing the basic pre-processing operations to get relevant pose information (skeleton and head model) and then transmit it to the back-end. It is also connected to the router that receives trajectory information from the processing unit and the anchor node (including also the health measurements). This information is also transmitted to the back-end.
 - ii. **Router:** All the details of the private network are stored in the router. Most relevant features of this device are related to the I/O operations from the sensing devices (including the addresses and the ports) and the security implemented to protect the data of the patients.
 - iii. **Processing unit:** Finally the processing unit allows the fulfillment of the privacy rules performing visual tracking algorithms in real-time and storing just the information of the trajectories in the system. This device has two I/O interfaces, first of them to capture the current image from the overhead camera, and second one of them to perform visual tracking.
- (b) **Inferences:** This layer located on the second position of the abstraction queue is intended to perform the filters and algorithms related to signal processing techniques. In the general architecture exposed previously we can differentiate between static and dynamic techniques, in this case we have followed this criterion as well.
 - i. **Static:** Static algorithms are focused on the analysis of the body pose and expression by analysing the signal provided by the corresponding RGB-D devices for this use case. Both of them require expensive computing capabilities and therefore they are captured and filtered in the sensing-hub environment, but the heavy processing with Machine Learning techniques is located on the back-end.
 - A. **Skeleton:** Raw data captured from the RGB-D sensor is filtered in order to build the skeleton of the patients represented by 25 joints of their bodies. Based on the location and orientation of those points the system is able to evaluate the evolution of the patients in the rehabilitation center.
 - B. **Facial analysis:** The Smart TV interface allows the definition of a scene where facial analysis can be executed for the patients. Initially the facial landmarks and head pose are estimated. The first of them correlated with the current position of the patient and the TV provide an interesting clue to evaluate the attention focus of the patient. Finally, based on the results of both, the current expression of the patient can be estimated. This is a very relevant information for the professionals in order to evaluate the progress and current state of the patient.

- ii. **Dynamic:** Dynamic algorithms fuse signals of a different nature such as video or WSN and health bracelet measurements. Both of them are extremely critical (input for online alerts) and therefore they run on real-time using intelligent devices previously presented.
 - A. **Trajectories (motion):** This is the basic inference required to evaluate the motion of the patients and pre-processing step for all the static inferences that require the definition of the region of interest. In this case, fusing different sensing signals to improve the results and the response of the system to problems such as occlusions. Online alerts will be triggered when there is a potential danger in the current location of the patient (e.g., Patient living alone and leaving the house)
 - B. **Health measurements (human-device interactions):** The bracelet is designed to capture basic health measurements such as body temperature or heart rate. This data is very relevant for the medical professionals from a historical perspective, but also any abnormal values could trigger online alerts to the nurse related to the patient as soon as possible.
- (c) **Sensing devices:** Devices that allow signal acquisition are defined on this section.
 - i. **Overhead camera:** This camera provides a global view of the room being monitored. It has been selected because it offers a global view from an aerial perspective that allows dealing with occlusions in a simpler manner. In our proposal it is directly connected to a processing unit that runs the corresponding algorithm and the system does not perform any storage of the video captured.
 - ii. **RGB-D camera:** This device has a reduced scope, but provides more detailed information in spite of executing an exhaustive analysis of a determined region. The sensing server captures the data from the camera and performs initial filtering to get the skeleton and the information of the head of the patient for a further analysis.
 - iii. **WSN:** Devices included in this network, in the monitored indoor area complement the signal provided by the overhead camera, and allow capture of the data from the bracelet due to its reduced scope.
 - iv. **Bracelet:** Last device is intended to periodically check health constants of the patients and activate the emergency system in case they are in a critical range.
- (d) **Device connections:** The different nature of the sensing devices previously enumerated requires different connection interfaces, that are enumerated below with the corresponding data flows involved.
 - i. Ethernet:
 - A. Overhead camera-processing unit.
 - B. Processing unit-router.
 - C. Router-sensing server.
 - ii. WiFi:
 - A. WSN mote-anchor node.
 - B. Anchor node-router.
 - iii. Bluetooth:
 - A. Bracelet-anchor node.
 - iv. USB:

A. RGB-D Camera-sensing server.

- (e) **Configuration:** The sensing-hub previously defined demands an initial configuration for its deployment in a real scenario. That configuration must have two basic concerns: communication channels and sensing devices calibration
 - i. **Communication channels:** Several devices attached to a private network involving different connection interfaces require the configuration of the ports and addresses where the data flow during the capture. All those parameters are transferred from the sensing server to the router in order to optimize communications, that should also be synchronized by time (by using timestamps) and identities (digital signature) to allow batch processing of the heaviest processes and high level inference from a Data Science perspective.
 - ii. **Sensing devices calibration:** Cameras should also be calibrated in order to perform translation from pixels to a unit of length included in the metric system (from points belonging to the cloud of points in case of 3D data). In the case of the WSN, a calibration is required from the motes to obtain accurate measurements, and finally both of them should be jointly on the same coordinate system to obtain preliminary results that support fusion of the signal processing algorithms.
- 2. **Back-end:** This environment is focused on the heavy Machine Learning processes of the system (in our case the ones concerning 3D information), the automatic/manual annotation of the corresponding results and/or the measurements in the data model. In addition, it supports, from the server perspective, the user interfaces included in the system.
 - (a) **Data model:** Hostage of all the relevant data for the improvement of patients quality of life is stored on this environment. Processes that interact with the database are divided in batch and real-time processes depending on their efficiency.
 - i. **Real-time:** These processes require a compromise between efficacy and efficiency to provide quality results in the time they are demanded by the system. Online alerts are highly dependent on the estimations of the system for the data stored in the entities enumerated below.
 - A. **Trajectories:** This entity is directly accessed by the processing unit once it estimates trajectories of the tracked patients every given unity of time (We have fixed this parameter to one minute for this use case).
 - B. **Electronic Health Record (EHR):** This entity is directly accessed by the corresponding anchor node that extracts the measurements from the bracelet. In this case we have defined the optimum period to extract the constants to be one hour.
 - ii. **Batch:** These processes either require high computational resources to provide accurate results, or require manual annotation from the clinic professionals side.
 - A. **EHR:** Different information such as treatment of the patients, or scales associated to their current state should be annotated by the corresponding clinical professionals.
 - B. **Rehabilitation:** This module is updated by both signal processing modules taking the skeleton as input using Machine Learning techniques, and also manual annotation of the corresponding scales from the corresponding physiotherapy professionals.

- C. Expression: This module is updated by a signal processing module that takes as input the 3D information of the head of the patient, extracts the corresponding facial landmarks, estimates the head pose, and based on their results infers the expression of the patient.
 - D. Digital Tracking: Once every day this module is updated with the information from the interfaces regarding the modules of the system that the patient has interacted with. In the case of Smart TV, the head pose also provides an estimation regarding the degree of attention of the patient. All of them include the time the patient spent and the path that they followed to complete the desired task.
- (b) **Interfaces:** Multi-platform interfaces (Android, IOS, Windows, MAC) should be implemented for the presented system, so that the different actors can interact with them. Every user of the system has a corresponding role linked that determines which information he or she is able to access or update.

3.3 Conclusions

With this chapter the following tasks have been addressed:

1. The **roles** that interact with the system, the functionalities demanded by every one of them and the scenarios of the use cases have been structured.
2. The **devices** required to perform specific behaviour modeling at the required domains have been specified.
3. The **architecture** defines the logical components involved, the physical connections among them and the data flow.
4. **Configuration** of the sensing devices included in the architecture

The system is able to provide visual data, whose processing for the tasks selected by the background research is detailed in the two following chapters.

Figure 3.8: Diagram showing the physical and logical architecture for health monitoring use case.

Chapter 4

HUMAN PEDESTRIAN TRACKING

This chapter introduces two complete solutions for visual pedestrian tracking in a static video surveillance scenario (there is only one camera and it has a fixed location). Several techniques and algorithms will be introduced at the beginning by following the structure for online systems exposed in subsection 2.4.3: background subtraction, detection and tracking. In our case, we have split detection step in features and classification to study their influence isolated. At that point, most of the relevant techniques for every step have been presented, and therefore solutions combining and optimizing those said techniques and algorithms can be built upon the corresponding training procedures. The proposed solutions have been tested on four real-life scenarios (ETSIT hall, retail store, cognitive surveillance and lab) whose main features will be presented. The combination of camera locations, solutions, techniques involved and obtained results are discussed for every scenario, specially for lab scenario where constraints are the least.

4.1 Background subtraction

The first step implemented for both video analysis solutions proposed aims to discriminate between static and dynamic elements in the scene. For that purpose static elements inherent to the scenarios should be modeled through a statistical distribution. That defined distribution will allow the determination of the moving objects present in the scene, considered as Regions of Interest (ROIs), for an algorithm whose aim is to define pedestrian trajectories.

Regarding practical details, the illumination in the scene could change gradually even in an indoor scenario (daytime/night) or suddenly (switching different light sources in the scene). A new object could be brought into the scene or a present object removed from it. In order to adapt to changes, the solution proposed in [145] has been initially adopted. The training set is updated by adding new samples and discarding the old ones. A reasonable time period of 600 frames has been chosen (sequences are recorded at a frame rate of 15 fps (Frames per second), therefore it corresponds to $T = 40$ sec.). Therefore at time t we have

$$X_T = x^t, \dots, x^{t-T} \quad (4.1)$$

For each new sample we update the training data set X_T and re-estimate $\hat{p}(\vec{x}|X_T, BG)$, the probability of every pixel ($\vec{x}$) with the prior knowlegde of the history modeled by a mixture of K Gaussian distributions (X_T) defining the inherent elements of the scene (BG), Background).

However, among the samples from the recent history there could be some values that belong to the Foreground (FG) objects and we should denote this estimate as $\hat{p}(\vec{x}|X_T, BG + FG)$. We use Gaussian Mixture Model (GMM) [118] with $M = 3$ components

$$\hat{p}(\vec{x}|X_T, BG + FG) = \sum_{m=1}^M \hat{\pi}_m N(\vec{x}; \hat{\mu}_m, \sigma_m^2 I) \quad (4.2)$$

Where $\hat{\mu}_1, \dots, \hat{\mu}_M$ are the estimates of the means and $\hat{\sigma}_1, \dots, \hat{\sigma}_M$ are the estimates of the variances that describe the Gaussian components. The covariance matrices are assumed to be diagonal and the identity matrix I has proper dimensions. The mixing weights denoted by $\hat{\pi}_m$ are non-negative and add up to one. Given a new data sample $\vec{\mu}^{(t)}$ at time t [144] defines the following recursive equations to update the gaussian model by the $\leftarrow$ operator:

$$\begin{aligned} \hat{\pi}_m &\leftarrow \hat{\pi}_m + \alpha(o_m^{(t)} - \hat{\pi}_m) \\ \hat{\pi}_m &\leftarrow \hat{\pi}_m + o_m^{(t)}(\alpha/\hat{\pi}_m)\vec{\delta}_m \\ \hat{\sigma}_m^2 &\leftarrow \hat{\sigma}_m^2 + o_m^{(t)}(\alpha/\hat{\pi}_m)(\vec{\delta}_m^T \vec{\delta}_m - \hat{\sigma}_m^2) \end{aligned} \quad (4.3)$$

Where $\hat{\pi}_m$ is the estimate of the means and $\hat{\sigma}_m$ is the estimate of the variance that describe the Gaussian components. The estimated mixing weights denoted by $\vec{\sigma}_m = x_{(t)} - \hat{\mu}_m$ are non-negative and add up to one. Instead of the time interval T that was mentioned above, here constant α describes an exponentially decaying envelope that is used to limit the influence of the old data. We keep the same notation bearing in mind that approximately $\alpha = 1/T$. For a new sample the ownership $o_m^{(t)}$ is set to 1 for the ‘close’ component with largest π_m and the others are set to zero. A sample is defined as ‘close’ to a component if the Mahalanobis distance [83] from the component is less than c_{thr} . In our implementation we have empirically found that the best results are obtained with $c_{thr} = 25$. The squared distance from the m -th component is calculated as: $D_m^2(\vec{x}^{(t)}) = \vec{\delta}_m^T \vec{\delta}_m / \hat{\sigma}_m^2$. If there are no ‘close’ components a new component is generated with $\hat{\pi}_{M+1} = \alpha$, $\hat{\mu}_{M+1} = \vec{x}^{(t)}$ and $\hat{\sigma}_{M+1} = \sigma_0$ where σ_0 is some appropriate initial variance. If the maximum number of components is reached we discard the component with smallest $\hat{\pi}_m$.

4.1.1 Non-parametric methods

Another work of Zikvovic [146] improves the efficiency of background subtraction by density estimation and it has been included in this work. For that aim a uniform kernel is used to count the number of samples k from the data set X_T that lie within the volume V of the kernel. The volume V is a hypersphere with diameter D . The density estimate is given by:

$$\hat{p}_{non-parametric}(\vec{x}|X_T, BG + FG) = \frac{1}{TV} \sum_{m=t-T}^t K\left(\frac{\|\vec{x}^{(m)} - \vec{x}\|}{D}\right) = \frac{k}{TV} \quad (4.4)$$

Where the kernel function $K(u) = 1$ if $u < 1/2$ and 0 otherwise. The volume V of the kernel is proportional to D^d where d is the dimensionality of the data.

In practice T is large and keeping all the samples in X_T would require excessive memory and calculating the previous equation would be too slow. It is reasonable to choose a fixed number of samples $K \ll T$ and randomly select a sample from each sub-interval T/K . This might give too sparse sampling of the interval T .

The X_T also contains samples from the foreground. Therefore, for automatic learning we also keep a set of corresponding indicators $b^{(1)}, \dots, b^{(T)}$. The indicator $b^{(m)}$ has a value 0 if the

sample is assigned to the foreground. The background model considers only the samples with $b^{(m)} = 1$ that were classified as belonging to the background:

$$\hat{p}_{non-parametric}(\vec{x}|X_T, BG) = \frac{1}{TV} \sum_{m=t-T}^t b^{(m)} K\left(\frac{\|\vec{x}^{(m)} - \vec{x}\|}{D}\right) \quad (4.5)$$

If this value is greater than the threshold c_{thr} the pixel is classified as background.

4.1.2 Shadow detection

To detect shadows in the scene (reported in fig. 4.1), this algorithm works in the HSV (Hue, Saturation and Value) color space [69]. The main reasons are that HSV color space corresponds closely to the human perception of color [48] and it has revealed increased accuracy when distinguishing shadows. In fact, a shadow cast on a background does not significantly change its hue [26]. Moreover, we have exploited saturation information since it has been experimentally evaluated that pixels located in the shadow often possess lower saturation. The resulting decision process is reported in the following equation:

$$SP_k(x, y) = \begin{cases} 1 & \text{if } \alpha \leq \frac{I_k^V(x, y)}{B_k^V(x, y)} \leq \beta \quad \wedge \quad I_k^H(x, y) - B_k^H(x, y) \leq \tau_H \quad \wedge \\ & \wedge \quad I_k^S(x, y) - B_k^S(x, y) \leq \tau_S \quad \wedge \quad F(x, y) = 1 \\ 0 & \text{otherwise} \end{cases} \quad (4.6)$$

Where $I_k(x, y)$ and $B_k(x, y)$ are the pixel values at coordinate (x,y) in the input image (frame k) and in the background model (computed at frame k), respectively (H, S and V represent Hue, Saturation and Value coordinates from the HSV color space respectively). The foreground mask $F_k(x, y)$ is set to 1 if the point at coordinates (x,y) is detected as probably in motion, and therefore, has to be used both for moving object detection and for shadow detection. The use of β prevents the identification as shadows of those points where the background was slightly changed by noise, whereas τ takes into account the ‘power’ of the shadow, i.e. how strong the light source is w.r.t. the reflectance and the irradiance of the objects. Thus, the stronger and higher the sun (in the outdoor scenes), the lower τ should be chosen. In our case best results were obtained with $\tau = 0.5$. A sample of those results is shown on fig. 4.1.

Figure 4.1: Left image shows result of background subtraction with shadows detected in grey intensity. Right image shows the final result of background subtraction.

4.2 Features

This section will go through the feature descriptors selected for human pedestrian tracking: HOG and LBP.

4.2.1 Histogram of Oriented Gradients (HOG)

Histogram of Oriented Gradients (HOG) [27] is a feature descriptor whose essential thought relies on the fact that the appearance and shape of any object can be described by the distribution of its gradient intensity. Those gradient intensities are characterized by the direction of the edges that define the shape of the object. It can be noted on fig. 4.2:

Figure 4.2: Image courtesy of [27]. The image on the left side shows the original image and the right image shows gradient orientations around the edges of the pedestrian. We can also see in an augmented region of the image, belonging to the shoulder of the pedestrian, magnitudes of the eight gradient directions considered by HOG

The method is based on evaluating well-normalized local histograms of image gradient orientations in a dense grid. In practice this is implemented by dividing the image window into small spatial regions ('cells'), for each cell accumulating a local 1D histogram of gradient directions or edge orientations over the pixels of the cell. The combined histogram entries form the representation. For better invariance to illumination, shadowing, etc., it is also useful to contrast-normalize the local responses prior to use them. This can be done by accumulating a measure of local histogram 'cells' over somewhat larger spatial regions ('blocks') and using the results to normalize all of the cells in the block. Tiling the detection window with a dense (in fact, overlapping) grid of the HOG descriptors and using the combined feature vector in a conventional Machine Learning based window classifier gives our human detection chain.

Figure 4.3: Image courtesy of [27]. Main steps of HOG feature extraction algorithm are shown.

Thus Initial implementation Dalal and Triggs summarized in fig. 4.3 considers a patch cropped out of an image and resized to 64×128 , in our case we have used a patch of 32×64 (in the first implementation also 16×32) to fit it to the average sizes of the ROIs in the testing scenarios that will be shown on the next subsections (the patches need to have an aspect ratio of 1 : 2). Gamma correction is skipped as a preprocessing step, since the performance gains are minor. Therefore, the initial step is desired in order to calculate the horizontal and vertical gradients. This is easily achieved by filtering the image with the following kernels $(-1, 0, 1)^T$ and $(-1, 0, 1)$. Next the magnitude and direction of gradient are calculated using the following formula:

$$\begin{aligned}\|\vec{g}\| &= \sqrt{g_x^2 + g_y^2} \\ \theta &= \frac{g_y}{g_x}\end{aligned}\tag{4.7}$$

The next step is the calculation of histogram of gradients, In our case we are using a block size of 16×32 (4 blocks per sample image). Each block of the image is divided into 8×8 cells (8 cells per block) and a histogram of gradients is calculated for each of them. An 8×8 image patch contains $8 \times 8 \times 3 = 192$ pixel values. The gradient of this patch contains 2 values (magnitude and direction) per pixel which adds up to $8 \times 8 \times 2 = 128$ numbers. These 128 numbers are represented using a 9-bin histogram which can be stored as an array of 9 numbers. The next step is to create a histogram of gradients in these 8×8 cells. The histogram contains 9 bins corresponding to gradient angles $\{0, 20, 40, \dots, 160\}$. It is not only the most compact representation, calculating a histogram over a patch makes this representation more robust to noise. Individual gradients may have noise, but a histogram over an 8×8 patch makes the representation much less sensitive to noise.

The gradients of an image are sensitive to overall lighting. We want our descriptor to be independent of lighting variations. In other words, we would like to ‘normalize’ the histogram so they are not affected by lighting variations. A 16×32 block has 8 histograms which can be concatenated to form a 72×1 vector. In order to normalize those 8 histograms, the window is moved by 8 pixels and a normalized 72×1 vector is calculated over this window and the process is repeated.

Finally to calculate the feature vector for the entire image patch, the 72×1 vectors are concatenated into one giant vector. The size of that vector will be composed of 576 features (72 features/block $\times$ 8 overlapped blocks/patch).

The HOG descriptor has a few key advantages over other descriptors. Since it operates on local cells, it is invariant to geometric and photometric transformations, except for object orientation. Such changes would only appear in larger spatial regions. Moreover, as Dalal and Triggs discovered, coarse spatial sampling, fine orientation sampling, and strong local photometric normalization permits the individual body movement of pedestrians to be ignored so long as they maintain a roughly upright position.

4.2.2 Local Binary Patterns (LBP)

Local Binary Patterns (LBPs) are a texture descriptor made popular through the work of Ojala et al. [94]. LBPs compute a local representation of texture constructed by comparing each pixel with its surrounding neighbourhood of pixels.

We start the derivation of our gray scale and rotation invariant texture operator by defining texture T in a local neighbourhood of a monochrome texture image as the joint distribution of the gray levels of $P(P > 1)$ image pixels:

$$T = t(g_c, g_0, \dots, g_{P-1})\tag{4.8}$$

where the gray value g_c corresponds to the gray value of the center pixel of the local neighbourhood and $g_p(p = 0, \dots, P - 1)$ correspond to the gray values of P equally spaced pixels on a circle of radius R ($R > 0$, in our implementation $P = 8$ and $R = 1.0$) that form a circularly symmetric neighbour set. If the coordinates of g_c are $(0,0)$, then the coordinates of g_p are given by $(-R \sin(2\pi p/P), R \cos(2\pi p/P))$. Fig. 4.4 illustrates circularly symmetric neighbour sets for various (P, R) . The gray values of neighbours which do not fall exactly in the center of pixels are estimated by interpolation.

Figure 4.4: Image courtesy of [94]. Circularly symmetric neighbour sets for different (P, R) .

As the first step towards gray scale invariance we subtract, without losing information, the gray value of the center pixel (g_c) from the gray values of the circularly symmetric neighbourhood g_p ($p = 0, \dots, P-1$) giving:

$$T = t(g_c, g_0 - g_c, g_1 - g_c, \dots, g_{P-1} - g_c) \quad (4.9)$$

This is a highly discriminative texture operator. It records the occurrences of various patterns in the neighbourhood of each pixel in a P -dimensional histogram. For constant regions, the differences are zero in all directions. On a slowly sloped edge, the operator records the highest difference in the gradient direction and zero values along the edge, and for a spot the differences are high in all directions. Signed differences $g_p - g_c$ are not affected by changes in mean luminance, hence the joint difference distribution is invariant against gray scale shifts. We achieve invariance, with respect to the scaling of the gray scale by considering only the signs of the differences instead of their exact values:

$$T \sim (s(g_0 - g_c), s(g_1 - g_c), \dots, s(g_{P-1} - g_c)) \quad (4.10)$$

$$s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (4.11)$$

By assigning a binomial factor 2^p for each sign $s(g_p - g_0)$, we transform the previous equation into a unique $LBP_{P,R}$ number that characterizes the spatial structure of the local image texture:

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_0) 2^p \quad (4.12)$$

4.3 Classification

This section will briefly explain the two Machine Learning algorithms chosen in order to determine if the patches or windows contained in the ROIs previously detected, belong to a human or not by discriminating the feature space determined by the feature descriptors previously exposed.

4.3.1 Adaboost

Adaboost [35] (or ‘Adaptative Boosting’) is a Machine Learning ensemble meta-algorithm for primarily reducing bias, and also variance in supervised learning. The essential thought behind this binary classifier is that a set of weak learners create a single strong learner. A weak learner is defined as a classifier which is only slightly correlated with the true classification (it can

label examples better than random guessing). In contrast, a strong learner is a classifier that is arbitrarily well-correlated with the true classification. It is graphically exposed on fig. 4.5.

Figure 4.5: Image courtesy of [25]. Images on the Left column show the results of three weak classifiers. In the right column steps to build the final strong classifier can be noted.

In our case, we have chosen a two-class discrete AdaBoost Algorithm for **Boosting** solution based on the input from the first feature descriptor presented previously (HOG). Lets assume we have a set of N samples labeled: $(x_i, y_i) \quad \forall i = 1 \dots N$ with $x_i \in R^K, y_i \in -1, +1$. First some weights should be assigned to the different weak classifiers: $w_i = 1/N \quad \forall i = 1, \dots, N$.

Next steps are followed until the algorithm converges:

1. Choose $h_t(x)$:
 - (a) Find weak learner $h_t(x)$ that minimizes ϵ_t , the weighted sum error for misclassified points

$$\epsilon_t = \sum_{i=1, h_t(x_i) \neq y_i}^n w_{i,t} \quad (4.13)$$

- (b) Choose $\alpha_t = \frac{1}{2} \ln \left(\frac{1-\epsilon_t}{\epsilon_t} \right)$

2. Add to ensemble:

$$F_t(x) = F_{t-1}(x) + \alpha_t h_t(x) \quad (4.14)$$

3. Update weights:

- (a) $w_{i,t+1} = w_{i,t} e^{-y_i \alpha_t h_t(x_i)}$ for all i
 - (b) Renormalize $w_{i,t+1}$ such that $\sum_i w_{i,t+1} = 1$

4.3.2 Support Vector Machine

A Support Vector Machine (SVM) [24] is a non-probabilistic binary classifier formally defined by an optimal separating hyperplane. This technique learns a supervised linear model based on a training set previously labeled in order to predict the class that a new instance belongs to (in our case it is included in **Hierarchical Tracking (HT)** to predict if one image patch belongs to a human pedestrian or not). Therefore the essential thought behind this algorithm is to find the hyperplane that maximizes the distance to the closest point belonging to both classes previously defined: ‘maximum-margin hyperplane’. It is graphically depicted in fig. 4.6.

Figure 4.6: Image courtesy of [96]. Graphical representation of SVM maximum-margin hyperplane.

The optimal hyperplane can be represented in an infinite number of different ways by scaling of β and β_0 . As a matter of convention, among all the possible representations of the hyperplane, the one chosen is:

$$|\beta_0 + \beta^T x| = 1 \quad (4.15)$$

Where x symbolizes the training examples closest to the hyperplane, in our case those patches belonging to human pedestrians whose visual features (HOG and LPP) are very similar to the ones of the background or vice versa. In general, the training examples that are closest to the hyperplane are called support vectors. This representation is known as the canonical hyperplane. Now, we use the result of geometry that gives the distance between a point x and a hyperplane (β, β_0) :

$$\text{distance} = \frac{|\beta_0 + \beta^T x|}{\|\beta\|} \quad (4.16)$$

Finally, the problem of maximizing M is equivalent to the problem of minimizing a function $L(\beta)$ subject to several constraints. The constraints model the requirement for the hyperplane to classify correctly all the training examples x_i . Formally:

$$\min_{\beta, \beta_0} L(\beta) = \frac{1}{2} \|\beta\|^2 \mid y_i(\beta^T x_i + \beta_0) \geq 1 \quad \forall i \quad (4.17)$$

where y_i represents each of the labels of the training examples.

This is a problem of Lagrangian optimization that can be solved using Lagrange multipliers to obtain the weight vector β and the bias β_0 of the optimal hyperplane.

4.4 Tracking solutions

The problem of tracking visual features in indoor environments has been widely studied recently. In this matter it is essential to adopt principled probabilistic models with the capability of learning and detecting the objects of interest. In this section we present two solutions that have been implemented combining techniques explained previously and new ones that will be explained in this section. A diagram explaining the first of them can be noted on fig. 4.7, and the different phases of the second one can be noted on fig. 4.8.

4.4.1 Boosting

Initially static elements of the scene are substracted using the method previously exposed for background substruction. HOG is the feature vector that has been chosen in this solution to express shape features of the pedestrians. In the next step, Adaboost will determine which windows belonging to a ROI correspond to real humans. After that, a modified implementation of Kmeans (explained below) will build detection bounding boxes. And finally, Particle Filter (also explained in this subsection) will include temporal features in an statistical model in order to predict more accurately the location of the pedestrians in the scene.

Figure 4.7: Diagram showing the most relevant phases for the first solution implemented for visual pedestrian detection and tracking (Boosting).

4.4.1.1 Modified K -means

K -means clustering [42] is a method of vector quantization, widely used for cluster analysis in Data Mining. The aim of the K -means algorithm is to divide M points in N dimensions into K clusters so that the within-cluster sum of squares is minimized. Local optima solutions are sought so that no movement of a point from one cluster to another will reduce the within-cluster sum of squares.

In our case we will use the basis of this algorithm to group all the positive pedestrian window results from the corresponding classifier. Modifications will be directed towards splitting occluded pedestrians that fail on the same blob or ROI, and/or provide their accurate location on the scene. For that aim next modifications are proposed:

1. Variable number of clusters K (initialized with the number of ROIs detected m).
2. Maximum distance allowed between two positive windows (max_{Dis}) so that they belong to the same pedestrian.
3. Limited number of iterations in order to maintain efficiency.

Given a set of observations $(x_1, x_2, \dots, x_n)$, where each observation is a 2-dimensional integer vector (x and y coordinates of the window detected since width and height are constant), clustering aims to partition the n observations into $K \geq m$ sets. Initially the observations belonging to the same blob are grouped together:

$$(x_1, x_2, \dots, x_n) \in Cluster_a \quad \forall \quad a \in \{1 \dots K\} \Rightarrow d(x_i, x_j) \leq max_{Dis} \quad \forall \quad i, j \in \{1 \dots N\} \quad (4.18)$$

Where $d(x_i, x_j)$ is the Euclidean distance between x_i and x_j . If distance from one point is less than max_{Dis} to more than one point, the closest one will be chosen. max_{Dis} has been defined in the following way

$$max_{Dis} = \sqrt{max_{DisW}^2 + max_{DisH}^2} \quad (4.19)$$

Where max_{DisW} is the maximum distance regarding x coordinate, and max_{DisH} is the maximum distance regarding y coordinate. They are defined in the following way:

$$max_{DisW} = \begin{cases} \text{if } ROI_{width} \leq 3win_{width} \Rightarrow max_{DisW} = win_{width} \\ \text{if } ROI_{width} \leq 2win_{width} \Rightarrow max_{DisW} = win_{width}/2 \\ ROI_{width}/2 \text{ otherwise} \end{cases} \quad (4.20)$$

$$max_{DisH} = \begin{cases} \text{if } ROI_{height} \leq 3win_{height} \Rightarrow max_{DisH} = win_{height} \\ \text{if } ROI_{height} \leq 2win_{height} \Rightarrow max_{DisH} = win_{height}/2 \\ ROI_{height}/3 \text{ otherwise} \end{cases} \quad (4.21)$$

Finally two clusters will be merged if next condition is satisfied:

$$d(\mu_x, \mu_y) \leq max_{Dis} \quad (4.22)$$

Where μ_x and μ_y are the mean values of x and y distributions of all the observations located in the same cluster. The algorithm will iterate until no cluster is merged or the algorithm reaches the maximum number of iterations (in our implementation fixed to 10). And μ_x and μ_y will be taken as final coordinates for the bounding boxes of the pedestrian detected.

4.4.1.2 Particle filter

Particle filters [95] methods are a set of genetic, Monte Carlo algorithms used to solve filtering problems arising in signal processing and Bayesian statistical inference. The strength of these methods lies in their simplicity, flexibility, and systematic treatment of nonlinearity and non-Gaussianity.

The boosted particle iteration introduces two important extensions of the Multi-tracking Particle Filter. First, it uses Adaboost in the construction of the proposal distribution. This improves the robustness of the algorithm substantially. Second, Adaboost provides a mechanism for obtaining and maintaining the mixture representation.

In non-Gaussian state-space models, the state sequence $\{x_t; t \in \mathbb{N}\}$, $x_t \in \mathbb{R}^{n_x}$ is assumed to be an unobserved (hidden) Markov process with initial distribution $p(x_0)$ and transition distribution

$p(x_t|x_{t-1})$, where n_x is the dimension of the state vector. In our tracking system, this transition model corresponds to a standard autoregressive dynamic model. The observations $\{y_t; t \in \mathbb{N}^*\}$, $y_t \in \mathbb{R}^{n_y}$ are conditionally independent given the process $\{x_t; t \in \mathbb{N}\}$ with marginal distribution $p(y_t|x_t)$, where n_y is the dimension of the observation vector.

We denote the state vectors and observation vectors up to time t by $x_{0:t} \simeq \{x_0, \dots, x_t\}$ and $y_{0:t}$. Given the observation and transition models, the solution to the filtering problem is given by:

$$p(x_t|y_{0:t}) = \frac{p(y_t|x_t)p(x_t|y_{0:t-1})}{p(y_t|y_{0:t-1})} \quad (4.23)$$

In standard Particle Filtering, we approximate the posterior $p(x_t|y_{0:t})$ with a Dirac measure using a finite set of N particles $\{x_t^i\}_{i=1 \dots N}$. To accomplish this, we sample candidate particles from an appropriate proposal distribution $\hat{x}_t^i \sim q(x_t|x_{0:t-1}, y_{0:t})$. Introducing results of Adaboost and modified K-means lead us to the following expression for proposal distribution given by this proposed mixture.

$$q_B^*(x_t|x_{0:t-1}, y_{1:t}) = \alpha q_{ada}p(x_t|x_{t-1}, y_t) + (1 - \alpha)p(x_t|x_{t-1}) \quad (4.24)$$

where q_{ada} corresponds to the output of the detected pedestrians from the modified version of K-means. The parameter α is set dynamically without affecting the convergence of the Particle Filter (it is only a parameter of the proposal distribution and therefore its influence is corrected in the calculation of the weights).

4.4.2 Hierarchical tracking (HT)

Boosting was only based in spatial properties to merge detection and tracking steps, and did not introduce any hierarchy for the human pedestrian trackers. In the new solution implemented whose flow diagram can be noted on fig 4.8 color appearance properties will be used in order to avoid occlusions and provide a more accurate location of the pedestrians, and temporal properties of the trackers will be contemplated to build a hierarchical classification with two states: novices and experts based on the tracking solution exposed in [141]. In the training step the blobs or ROIs of the training set are extracted, and the feature vectors built from the windows contained are used to train an SVM classifier. In the testing step, first Mixture of Gaussians is applied in order to remove inherent regions of the scene, afterwards an SVM classifier predicts if the containing windows belong to a human or not, and finally a hierarchical tracking determines proper location of the pedestrians in the scene.

Figure 4.8: Diagram showing the most relevant phases for the second solution implemented for visual pedestrian detection and tracking (HT).

4.4.2.1 Template ensemble

This technique models human appearance by a set of templates updated online taking as the input, positive results from previously defined SVM classifier. In each frame I^t , a detector outputs a set of N_d detections $D^t = \{d_i^t\}_{i=1}^{N_d}$, in which d_i^t is a bounding box in frame I^t . At time t , the state of the tracker T_i is defined as $\{F_{T_i}^t, b_{T_i}^t, kf_{T_i}^t\}$, where the template ensemble $F_{T_i}^t = \{f_i^j\}_{j=1}^{N_i^t}$ is the set of N_i^t templates collected across time, $b_{T_i}^t = [xc_i^t, yc_i^t, w_i^t, h_i^t]$ defines the current estimate of the target's bounding box centered at (xc_i^t, yc_i^t) and of size (w_i^t, h_i^t) , and kf_i is the Kalman filter [84] used to model the tracker's dynamics. The Kalman filter's state is defined as the tracker's 2D center point position and the 2D velocity $(x, y, \dot{x}, \dot{y})$

4.4.2.2 Backprojection map

As illustrated in fig. 4.9, the distribution of pixel values in the back-projection image may not be evenly distributed on the person's body, a central patch (inner window) inside the detection window (outer window) is used as the foreground area, while the ring region surrounding the detection window is taken as background. When we apply mean-shift tracking [23] on the back-projection image (BP), the tracking window shifts to the densest area, causing a bias in position with respect to the object's center. Every time a detection is associated to a target, a template is added to the corresponding template ensemble. A template is defined as $f := \{channel[2], H, v_{shift}\}$ where $channel[2]$ represents two color channels selected to represent the template, H is the 2D histogram for the two selected channels (4D channel containing both inner and outer window) and v_{shift} is a 2D shifting vector that accounts for such bias so that, during mean-shift tracking, the feature template is shifted back to the center of the body. The two color channels are selected among the 9 from RGB [123], HSV [69] and Lab [45] color spaces to yield separation of pixels within the foreground and background. The best two channels are selected based on the variance ratio between the background and foreground's feature weight distributions.

Figure 4.9: Image courtesy of [141]. Color-feature selection and back-projection map for hierarchical tracking.

To maintain the template ensemble, a score is computed for each template based on its back-projection image given the updated target's position from the tracker. The tracker maintains most N_{max} templates by discarding all the templates with negative scores, and the templates with lowest scores. The score is computed as follows:

$$Score(f) = \underset{inner}{meanBP}(f) - \underset{r \subset R}{max}\{ \underset{r}{meanBP}(f) \} \quad (4.25)$$

Where $\underset{inner}{meanBP}(f)$ is the mean pixel value inside the inner window $BP(f)$; the second term $\underset{r \subset R}{max}\{ \underset{r}{meanBP}(f) \}$ takes the mean pixel value in the patch r whose size is the same of the inner window, and whose mean value is the highest within the ring area R .

4.4.2.3 Tracking

Tracking is performed by alternating between mean-shift tracking (relying on target's appearance) and Kalman filtering (relying on target's motion dynamics). Fig. 4.10 summarizes the main steps needed to perform tracking. At each frame, each template of tracker T_i is used to compute a back-projection image BP. The set of computed BP images is then averaged to provide a voting map. A mean-shift tracker is applied to the voting map to estimate the new target's position. This position is then used as the measurement input to the Kalman filter kf_i , which will predict the position for T_i to start the mean-shift tracker at the next frame. Trackers adopt different update strategies based on the template ensemble characteristics.

Figure 4.10: Diagram showing the most relevant phases for the hierarchical tracking implemented in the second solution for visual pedestrian detection and tracking.

We assume that a tracker's confidence may be quantified as the number of templates it possesses. Trackers are divided into two groups:

1. **Experts** are trackers having more than just K templates. These trackers have high confidence and may use their template ensemble to perform tracking.
2. **Novices** are trackers having fewer K templates. These trackers have low confidence and during tracking rely more on the newly assigned detections than on their template ensemble.

We will also consider a group of candidate trackers, which are trackers waiting for enough matched detections before being accepted as a novice. Once a novice is initialized, it tries to accumulate templates that may robustly track the target. After K templates have been accumulated, a novice is promoted to expert. Conversely, an expert that loses templates is demoted to novice; this usually occurs when the target undergoes occlusions or appearance changes. A novice will retain its last template if all the templates' scores are negative. In our implementation (N_{max}, K) are set to $(10, 5)$.

4.4.2.3.1 Mean-shift tracker. Initial prediction given by SVM output is used in order to calculate corresponding backprojection maps, in order to provide a more accurate prediction that feeds the corresponding kf_i . Let $H_{fr}(i)$ be the histogram from selected color components for pixels on the pedestrian (inner window), and $H_{bg}(i)$ be a histogram for pixels from the background sample (outer window). We form an empirical discrete probability density $p(i)$ for the object, and density $q(i)$ for the background, by normalizing each histogram by the number of elements in it:

$$p(i) = H_{ped}(i)/n_{ped} \quad (4.26)$$

$$q(i) = H_{bg}(i)/n_{bg} \quad (4.27)$$

With n_{ped} and n_{bg} being the number of object and background samples, respectively. The log likelihood of a feature value i is given by:

$$L(i) = \log \frac{\max\{p(i), \delta\}}{\max\{q(i), \delta\}} \quad (4.28)$$

Where δ is a small value (we set it to 0.0001) that prevents dividing by zero or taking the log of zero. The nonlinear log likelihood ratio maps potentially multimodal pedestrian/background distributions into positive values for colors distinctive to the object, and negative for colors associated with the background. Colors that are shared by both object and background tend towards zero. A new image composed of these log likelihood values becomes the Backprojection map image used for tracking.

4.4.2.3.2 Kalman filter. The detection association process has three steps. First, detections are matched to the experts. Then, remaining detections are matched to the novices. Finally, detections that are not associated to experts or novices are matched with candidates on the waiting list. If any detections remain unassigned after these steps, then these are added to the waiting list. Thus, to initiate a novice a minimum number of matching observations must be found.

Given a set of detections, its assignation (to existing trackers kf_i or to a new candidate) is solved finding the maximum correspondence in a bipartite graph. For that aim Hungarian Algorithm [88] is used with the following cost matrix:

$$T \sim (s(g_0 - g_c), s(g_1 - g_c), \dots, s(g_{P-1} - g_c)) \quad (4.29)$$

$$s(x) = \begin{cases} dist_{ij}, & dist_{ij} < R_i \\ \infty, & otherwise \end{cases} \quad (4.30)$$

Where $dist_{ij}$ is the distance between current estimated position of the pedestrian, and detection center d_j^t . $R_i = \alpha_i \omega_i$ depends on the width of the target window (ω_i) and the growing search area for new detections (α_i).

While an expert corrects its Kalman filter at each step, a novice only updates the posterior state of its Kalman filter without shrinking the posterior covariance. The implemented method switches between two modes of measurement noise covariance in Kalman filter by thresholding the number of templates that a tracker has. In this way, a novice has a growing search area and, therefore, it has more chances to be associated with a detection even though it may lose its track for a short time. When a new detection $(x_{det}, y_{det}, w_{det}, h_{det})$ is associated to a novice T_i , the posterior state of kf_{T_i} , $(x, y, \dot{x}, \dot{y})$, is replaced by $(x_{det}, y_{det}, \dot{x}, \dot{y})$. In this way, the novice jumps to the position of the newly matched detection. This allows a novice to recover its tracking efficiently after a short-term occlusion or change of appearance.

Beginning and end of one tracker kf_i are calculated based on the matching ratio τ_i and the average matching ratio τ between all trackers established at current time t . τ_i is defined in the following way:

$$tau_i = \frac{\Delta N_i^{matched}}{\Delta t} \quad (4.31)$$

Where $\Delta N_i^{matched}$ is the number of detections matched with T_i in a temporal sliding window of length Δt .

4.5 Testing scenarios

For both visual tracking solutions previously presented, four testing sets have been designed in different scenarios whose features will be presented in this section:

1. **ETSIT Hall.** **Boosting** has been tested in this scenario simulating a Digital Signage environment [41]. For that aim the camera was located on the ceil of a tv (at a height of

2m and an angle of -25°) with different commercials, and tracking system was combined with a content recommendation system outside of the scope of this thesis.

2. **Retail store.** **HT** has been tested in this scenario in order to provide low level sensing solutions able to improve the efficiency of retail establishments. In this case a zenithal plain is obtained locating the camera at 3.5m.
3. **Cognitive surveillance.** **HT** has been tested in this scenario in order to extend behavioural information for expert clinicians [47]. In this case a zenithal plain is obtained locating the camera at 3m.
4. **Lab.** **HT** has been tested in this scenario in combination with other sensing devices such as Bluetooth or WiFi building a Wireless Sensor Network (WSN). In this case a zenithal plain is obtained locating the camera at 3m.

The cameras used to acquire video signal in the scenarios previously enumerated are shown in fig. 4.11.

Figure 4.11: Cameras used to capture the data in the proposed scenarios. Left image corresponds to Logitech Webcam C930e and right one to Mobotic Hemispheric C25 6MP.

Logitech Webcam C930e is a low cost and easily portable video camera with no optics settings, but a fast USB rate and 90° field of view. It provides two recording modes:

1. High Definition (HD), resolution of 1920×1080 and 40 fps.
2. Standard Definition (SD), resolution of 640×480 and 25 fps (in our case we will use 15fps).

Standard definition has been chosen in order to prevent efficiency of the algorithm.

Mobotic Hemispheric C25 6MP is camera with a *fisheye* lens whose features are detailed below:

1. Sensor 1/1.8" CMOS Progressive-Scan.
2. Maximum resolution of 6MP (in our case we will keep using 480×480 to maintain efficiency).
3. Hemispheric lense of 1.6mm y f/20.0.
4. Frame resolution from 8fps to 30fps (in our case we will keep using 15fps).
5. Field of view of $180^\circ \times 180^\circ$.

4.5.1 Training

This section is intended to describe the process implemented in order to train Machine Learning algorithms presented previously. Figure 4.12 shows the scheme designed for that aim.

Figure 4.12: Diagram showing the scheme followed to train Machine Learning algorithms.

Initially background subtraction is applied, and the windows contained in the blobs built from the ROIs obtained are extracted and manually classified depending on whether they belong to a pedestrian or not. These images are combined with regular images obtained from the Internet (negative images) without any pedestrian on the scene and they are adjusted to the proposed blob size in order to extract the feature descriptors (HOG, LBP). Those feature descriptors are used to train the two Machine Learning solutions presented previously, and obtained classifiers will be tested in the proposed scenarios. The most relevant information for the different training procedures performed can be noted on tab. 4.1.

	0		1.2		2.a		2.b		2.1	2.2
	Angle	Resol.	Blob	Win	Blobs	Wins	Im.	Wins	FD	Class.
1	High	640x480	30x90	16x32	250	1500	200	4000	HOG	Boost
2	Zenith	640x480	32x64	32x64	500	500	100	1000	HOG	SVM
3	Zenith	640x480	32x64	32x64	500	500	100	1000	HOG,LBP	SVM

Table 4.1: Parameters specification for the training procedures performed in the different scenarios (1: ETSIT Corridor, 2 Retail store and Cognitive Surveillance, 3: Lab). The following columns specify the angle of acquisition and the resolution of the acquired video. The size and number of extracted blobs are exposed as well, but only for positive samples (pedestrians), for negative samples the windows are extracted from images containing different information (indoor/outdoor, objects/landscapes, ...). The last two columns specify the feature descriptors and the classification algorithm used.

4.5.2 ETSIT Hall

Five sequences of 450 frames recorded in the main corridor of the Escuela Tecnica Superior de Ingenieros de Telecomunicacion (ETSIT) that belongs to Universidad Politecnica de Madrid (UPM) have been selected in order to test the first solution. From now on they will be named as: Opening, Midday, Lunch, Afternoon and Closing. Several of the visual results of the different phases of the implemented solution previously described can be explored on fig. 4.14. Since the tests performed are related to Digital Signage purposes it was defined as an area of $3 \times 3 \text{ m}^2$ where it is supposed that the pedestrians are being impacted by the publicity shown on the screen. For that aim we have evaluated the angle where the camera is located and we have delimited the area in the image. It can be noted on fig. 4.13.

Figure 4.13: A sequence of three images is shown where a pedestrian is getting closer to the camera located on the screen. It can be observed how proposed visual detection and tracking algorithm estimates effectively when the user is impacted by the publicity on the screen.

Figure 4.14: Six figures showing the visual results for the most relevant phases of the first solution implemented for visual pedestrian and tracking (from left to right, up to down): Original, Background subtraction, positive results of Adaboost with HOG features, final windows for detection after K-means, impact estimation and Particle filter results.

4.5.3 Retail store

Two sequences of 250 frames recorded during the daytime at a retail establishment have been used in order to test the accuracy of the second solution presented. The first of them was recorded at a time when there was a low frequency (LF) of pedestrians in the shop, and the second was recorded when there was a high frequency (HF) of consumers there. From this point onwards, they will be named LF and HF. And one sample for each of them can be noted on fig. 4.15.

Figure 4.15: Visual tracking results of two sample images extracted from both sequences of the retail store are shown.

4.5.4 Cognitive surveillance

Two sequences of 600 frames recorded during the daytime in a cognitive patients association have been used in order to test the accuracy of the second solution presented. The first of them presents specific human pedestrians including clinical experts and patients. The second of them presents mixed human pedestrians (caregivers, clinic experts, association stuff and patients). From now on they will be named specific and mixed and one sample for each of them can be noted on fig. 4.16

Figure 4.16: Visual tracking results of two sample images extracted from both sequences of the cognitive patients association are shown.

4.5.5 Lab

This subsection is describing all the tests carried out in the Lab scenario. This is the scenario where most complete tests were performed since it allows for all the deployments necessary, and there are no restrictions with lab colleagues related to Data Privacy. Tests performed include:

1. Calibration
2. Feature selection
3. Multisensing solution
4. Real-time solution

Two video sequences of 200 frames each have been selected in order to test the visual detection and tracking solution proposed (HT). Both of them include pedestrians walking in the scene at a regular speed. The first one includes 3 pedestrians walking whose cross their trajectories at internal and centered regions of the scene (3Ped). The second one includes 4 pedestrians whose cross their trajectories at the external and centered regions of the scene (4Ped). The different regions of the scene can be noted on fig. 4.17.

Figure 4.17: Sample image from the lab where the last experiments where performed including the delimitation of the areas considered for the second solution of the pedestrian visual detection and tracking.

4.5.5.1 Calibration

In order to fuse the visual detection and tracking algorithm proposed in HT with an adaptive fingerprint based tracking solution, a calibration step was required. For that purpose first the distortion introduced by the ‘fisheye’ lens is corrected, and then a method to convert pixel values to cm is provided.

4.5.5.1.1 Distortion correction. To correct the distortion introduced by the hemispheric camera used in this scenario radial and tangential factors were taken into account. For the radial factor introduced by the ‘fisheye’ lense, the following formula was used:

$$\begin{cases} x_d = x(1 + k_1r^2 + k_2r^4 + \dots + k_i r^{i^2}) \\ y_d = yf(1 + k_1r^2 + k_2r^4 + \dots + k_i r^{i^2}) \end{cases} \quad (4.32)$$

Where:

1. (x, y) are the coordinates of the pixel on the distorted image.
2. (x_d, y_d) are the coordinates of the pixel on the corrected image.
3. $f(r, k_i)$ is the function that corrects geometrical distortion.

4. r represents radial distortion.

5. $k = [k_1, k_2, \dots, k_i]$ is the vector of distortion coefficients.

In our case we have chosen the rational model where $k = 6$ since it was empirically found to provide the best results. Tangential distortion occurs because the image capturing lenses are not perfectly parallel to the imaging plane. It has been corrected via the following formulas:

$$\begin{cases} x_d = x + [2p_1xy + p_2(r^2 + 2x^2)] \\ y_d = y + [p_1(r^2 + 2y^2) + 2p_2xy] \end{cases} \quad (4.33)$$

Where p_1 and p_2 are the tangential factors. Fig. 4.18 shows the chessboard used for radial and tangential correction, original image captured from the camera, and obtained image after correction.

Figure 4.18: Chessboard used and visual results for the calibration in the lab scenario.

4.5.5.1.2 From pixel to cm. Finally, the conversion from pixels to real coordinates in a common system for all the sensors considered is performed by using the Perspective-n-Point formula. As a result, the real 3D coordinates are computed as follows:

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (4.34)$$

Where s is a scale parameter and u, v are the 2D projections (from the undistorted image). On the right side, the first matrix contains the intrinsic camera parameters: focal lengths f_x, f_y and optical center c_x, c_y matching the 2D pixels with the 3D real world coordinates $[X, Y, Z, 1]^t$. Furthermore, the second matrix, comprises the rotation r_{ij} and translation t_i values that change the coordinate reference system.

4.5.5.2 Feature selection

Tests developed in the retail store with the zenith view and ‘fisheye’ lense showed that the HOG descriptor was able to provide accurate results for the regions of the image where the whole body of the pedestrian was present, but for centered regions of the image results were not quite satisfactory. In Solution 1 the design is mainly focused on shape properties, improvements in Solution 2 included color information and a more complex and efficient statistical model in order to contemplate temporal features. However, following [68] we decided to test a combination of both feature descriptors HOG and LBP in the different regions of the scene using [54] in order to reduce the dimensionality of the feature vector and not overload the SVM classifier with redundant features.

4.5.5.3 Multisensing solution

Solution 2 has been combined with adaptative fingerprinting solution for indoor tracking which is presented in [11]. For that aim the deployment shown in 4.19 was performed in Lab scenario:

Figure 4.19: Multi-sensing deployment performed in the lab scenario including 4 Raspberry Pies and one zenith camera.

Four Raspberry Pi were located in the corners of the ceiling and one zenith camera in the center. In order to combine the tracking estimation from both sensing solutions measurements of the fingerprinting were taken from a reference plane whose height is 1.40 m and centroids of the bounding box estimated by Solution 2 previously presented were used as input parameters for the trajectories in a time-manner. Fusion equation implemented is presented below:

$$p_{f_i} = w_{c_1} \vec{p}_{c_i} + (1 - w_{c_1}) [w_{b_i} \vec{p}_{b_i} + w_{w_i} \vec{p}_{w_i}] \quad \forall \quad i = 1 \dots t \quad (4.35)$$

$$w_{c_1} + (1 - w_{c_1}) = 1; \quad w_{b_i} + w_{w_i} = 1; \quad (4.36)$$

The vectors $vecp_{c_i}$, $vecp_{b_i}$, $vecp_{w_i}$ correspond to the position estimation (X, Y , height is approximated to 1.4m) provided by the solutions from every sensing device (camera, Bluetooth and WiFi) during a given time i . Regarding time dimensionality, trajectories obtained from fingerprinting are composed by 1 measurement per second, and Computer Vision solution by 15 estimations per second. A prefixed temporal sliding window of 1 sec is used in order to interpolate fingerprinting estimations, and coordinate temporal shifts between the different signals obtained by the system. Accuracy provided by Solution 2 is also higher than the one provided by the fingerprinting solution, therefore the weights of trajectories provided by the camera solution are much higher than for the other two sensing solutions. But it should be also taken into account that trackers can loose positions at a certain time due to occlusions (false negatives), and as long as the pedestrians move further from the camera, the accuracy is lower for the estimations of Solution 2 (because of the distortion introduced by the ‘fisheye’ lense). In those two cases estimations provided by the fingerprinting solution improves results of the final trajectories composed from multisensing fusion.

4.5.6 Baseline solutions

In order to validate HT three baseline solutions for visual tracking have been tested:

1. **State of the art top tracking algorithms.** In order to validate the solution proposed we initially tested two of the most widely used tracking techniques nowadays:

- (a) **Tracking Learning Detection (TLD)** [56]
- (b) **Kernelized Correlation Filter (KCF)** [46]

Both of them require an initialization step where a bounding box enclosing target objects to be tracked (in our case pedestrians) are specified. For that aim we selected some frames from the results of the detection step from the last scenarios previously presented (retail store and lab). Several samples are presented on fig. 4.20, fig. 4.21 and fig. 4.22. We selected the number of frames according to the complexity of the scenario and the proper sequence (e.g., the most complex scenario is the retail store, and the most complex sequence is the one with high frequency of pedestrians and therefore we used four frames to test tracking accuracy of that solution).

Figure 4.20: Sample results of detection step selected for testing TLD and KCF at retail scenario.

Figure 4.21: Sample results of detection step selected for testing TLD and KCF at cognitive surveillance scenario.

Figure 4.22: Sample results of detection step selected for testing TLD and KCF at Lab scenario.

Results obtained with KCF showed a notably better accuracy and efficiency than the ones provided by TLD and for that reason the last ones will not be included in the results displayed in the next section.

2. **Kalman and Hungarian tracking (Kalman).** The last baseline solution tested is intended to validate detection based on HOG features and SVM classification and hierarchical solution exposed for multitasking. Instead of using a classifier based on specific features to detect ROI in the image, in this case a contour extraction detection specific for overhead cameras and optic system used for acquisition has been tested. Initially contours of the image are extracted and the ones whose size is less than 10% of the largest contour's area are removed. In the next step contours that belong to the same person are joined based on the distance to their mass centers using the next formula:

$$dist \leq \alpha * \max(\max(width(b1), height(b1)), \max(width(b2), height(b2))) \quad (4.37)$$

Where α is a weight parameter in the solution tested empirically fixed to 0.7, $b1$ is the bounding box of the first contour and $b2$ is the bounding box of the second solution. ROIs obtained are used as input for a similar tracking solution as the one previously proposed combining Kalman [84] and Munkres Hungarian algorithm [88], but in this case without using a hierarchical representation of the trackers such as the one proposed in HT.

4.6 Results

This section is intended to evaluate the tests carried out with the solutions previously presented on the scenarios already introduced.

4.6.1 Performance indicators

Precision and recall are the performance indicators chosen in order to evaluate visual detection and tracking of the human pedestrian previously exposed in the following proposed scenarios. It is calculated by the next equations:

$$Precision = \frac{tp}{tp + fp} \quad (4.38)$$

$$Recall = \frac{tp}{tp + fn} \quad (4.39)$$

Where:

1. tp (True positives):

- (a) Detection: Windows detected as pedestrian by the corresponding classifiers that belong to regions of the corresponding frame where a human is located.
- (b) Tracking: Bounding boxes estimated by the corresponding solution where a pedestrian is located in the sequence at a specific time t .

2. fp (False positives):

- (a) Detection: Windows detected as pedestrian by the corresponding classifiers that belong to regions of the corresponding frame where a human is not located.
- (b) Tracking: Bounding boxes estimated by the corresponding solution where a pedestrian is not located in the sequence at a specific time t .

3. fn (False negatives):

- (a) Detection: Windows not detected as human pedestrian by the corresponding classifiers where a pedestrian is located on the scene.
- (b) Tracking: Bounding boxes not estimated by the corresponding solution where a pedestrian is located in the sequence at a specific time t .

4.6.1.1 Probabilistic interpretation

It is possible to interpret precision and recall not as ratios but as probabilities:

- 1. **Precision** is the probability that a (randomly selected) retrieved pedestrian is relevant.
- 2. **Recall** is the probability that a (randomly selected) relevant pedestrian is retrieved in a search.

It should also be noted that the probability that a retrieved pedestrian is relevant depends on their visual appearance and the concrete conditions of the scenario. In the context of a surveillance system whose aim is to provide sensing estimations that will be further analysed by a data science solution, recall is the most important performance indicator since trajectory analysis algorithms will be able to correct false positives more easily than false negatives. Therefore, this performance indicator will be taken as the one to predominate in the comments for the results discussed in the next section.

4.6.2 ETSIT Hall

Initial tests were mostly focused on the performance of Boosting, influence of pedestrian flows, light source and evaluation of the publicity impact area. Results of the tracking solution for sequences previously enumerated are shown in table 4.2 and table 4.3

Sequence	True positives	False positives	False negatives
Opening	72.32 %	17.79 %	9.88 %
Midday	74.57 %	14.82 %	10.60 %
Launch	83.32 %	6.48 %	10.19 %
Afternoon	81.89 %	7.67 %	10.43 %
Closing	80.73 %	7.74 %	11.52 %

Table 4.2: Results for visual tracking obtained by the Boosting solution in the sequences from the hall at ETSIT.

Sequence	Precision	Recall
Opening	80.25 %	87.98 %
Midday	83.42 %	87.55 %
Launch	92.78 %	89.10 %
Afternoon	91.43 %	88.70 %
Closing	91.25 %	87.51 %

Table 4.3: Precision and Recall for visual tracking obtained by Boosting solution in the sequences from the hall at ETSIT.

From the results exposed it can be inferred that specular reflections introduced by sunlight during opening and midday periods alongside the massive flow (occlusions) of people getting entering school on the same public transport increase the number of false positives. In an analogous fashion, shadows introduced by artificial light sources increase the number of false negatives for closing period. These facts are reflected as well in Precision and Recall results where the highest difference can be noticed in Opening and Midday periods due to the highest values in false positives rates. Highest rates are obtained during lunch time where both light sources coexist and mitigate their effects, and crowds are not present in the scene .

For the impact evaluation we have not evaluated Precision and Recall since there were no false positives. Results are shown on table 4.4.

Sequence	True positives	False negatives
Opening	62.32 %	17.79 %
Midday	70.34 %	29.65 %
Launch	74.32 %	25.67 %
Afternoon	72.29 %	27.7 %
Closing	71.31 %	27.68 %

Table 4.4: Results for impact evaluation from the hall at ETSIT.

In this table in can be observed that scenario illumination and pedestrian flow influence analogously the results, but in this case rates are a slightly lower since the task is more complex.

4.6.3 Retail store

The tests performed in retail store are intended to analyse the feasibility of the algorithm in a real scenario. In this case a strong artificial illumination is present on the scene throughout the day, and the main feature tested is the pedestrian flow. Results can be noted on table 4.5 and table 4.6

Sequence	True positives	False positives	False negatives
Low frequency	82.69 %	17.30 %	0 %
High frequency	79.30 %	9.86 %	10.83 %

Table 4.5: Results for HT on the sequences at a retail store.

Sequence	Precision	Recall
Low frequency	82.69 %	100 %
High frequency	88.93 %	87.98 %

Table 4.6: Precision and recall for HT on the sequences at a retail store.

The tests performed demonstrated the expected behaviour of the algorithm proposed. With a low frequency of pedestrians there are not pedestrian not being tracked on the scene, but they appear when crowds are present in the setting due to occlusion. Lower accuracy of the results than in the previous scenario can be explained by the introduction of a zenith camera with a *fisheye* lens located at a high height due to the design of the retail establishment. Solutions to these problems will be explored in the next scenario where a deeper study is performed.

4.6.3.1 Baseline comparison

Comparison of the proposed algorithm with baseline solutions tested can be explored in tab 4.7 and tab 4.8.

	KCF			Kalman			HT		
	TP	FP	FN	TP	FP	FN	TP	FP	FN
LF	80.71	9.64	9.64	84.87	4.52	10.61	82.69	17.30	0
HF	58.49	12.37	29.14	60.23	22.59	17.18	79.30	9.86	10.83

Table 4.7: Comparison of results obtained by baseline solutions with the results obtained by HT for retail scenario. All of them expressed in %.

For the low frequency sequence there are not significant differences between the algorithms. Kalman obtains the best rate and HT is the only one without any False Negative, but is penalized by the largest number of false positives. This can be explained by the specific contour extraction performed by Kalman that enhances the results from the detection step. KCF is not affected by that fact since the initialization step is already given, and in that initial phase is where most of the false positives are constrained. The high definition sequence exposes the biggest difference between the ratios obtained. In this sequence HT is the one with the best results with a remarkable difference regarding the other two algorithms. In this case the highest number of false positives is achieved by Kalman due to the problems of contour extraction when the number of occlusions is increased. In the same fashion the number of false negatives is decreased by HT which is able to correct errors in the detection step. For both solutions the highest rates are obtained with both Kalman filter based solutions which proves its suitability for tackling this problem.

The highest rates are obtained by HT except for Precision of the low frequency sequence due to the false positives obtained in the initial frames. Most noticeable differences are in the high frequency sequence. On the one hand, regarding recall if KCF and HT are compared.

	KCF		Kalman		HT	
	Precision	Recall	Precision	Recall	Precision	Recall
LF	89.33	89.33	94.94	88.89	82.69	100
HF	82.54	66.75	72.72	77.81	87.98	88.93

Table 4.8: Comparison of precision and recall obtained by baseline solutions with results obtained by HT for retail scenario. All of them expressed in %.

Performance of KCF is much lower due to the lack of a continuous detection step updating tracking input, and a hierarchical Kalman Filter able to correct that input. On the other hand, regarding precision if Kalman and HT are compared, Kalman performance is much lower due to the problems of a contour based detection for a sequence that includes a noticeable number of occlusions.

4.6.4 Cognitive surveillance

In this case we present a complex scenario mixing artificial and sun lighting, where the camera is pointing to the main corridor of the association and two doors (one indoor and another one outdoor) are also present in the scene. Pedestrian flow is lower in this scenario, and therefore the number of occlusions is also smaller in this scenario. Results can be noted on table 4.9 and table 4.10

Sequence	True positives	False positives	False negatives
Mixed	82.18 %	12.47 %	5.34 %
Specific	87.71 %	12.29 %	0 %

Table 4.9: Results for HT on the sequences at the association for cognitive patients.

Sequence	Precision	Recall
Mixed	86.82 %	93.90 %
Specific	87.71 %	100 %

Table 4.10: Precision and recall for HT on the sequences at the association for cognitive patients.

The tests performed demonstrated a slight increase in the performance of the algorithm proposed. In this case we have proposed a more complex scenario where the pedestrian flow is more simple. We can assert with the results exposed that occlusions are the feature with the highest influence on the results. It can be observed as well that results of this scenario are lower for both performance indicators precision and recall due to the small crowds observed in the mixed sequence .

4.6.4.1 Baseline comparison

Comparison of proposed algorithm with baseline solutions tested can be noted on tab 4.7 and tab 4.8.

We can observe that for the lower pedestrian flow the results get closer than in the previous scenario. The most remarkable difference is true positives for a mixed sequence using the Kalman

	KCF			Kalman			HT		
	TP	FP	FN	TP	FP	FN	TP	FP	FN
Mixed	83.26	16.74	0	69.78	14.10	16.11	82.18	12.47	5.34
Specific	63.72	19.65	16.61	81.78	9.98	4.23	87.71	12.29	0

Table 4.11: Comparison of results obtained by baseline solutions with results obtained by HT at the association for cognitive patients. All of them expressed in %.

	KCF		Kalman		HT	
	Precision	Recall	Precision	Recall	Precision	Recall
Mixed	83.26	100	83.19	88.89	86.82	93.90
Specific	76.43	79.32	89.58	93.22	87.71	100

Table 4.12: Comparison of precision and recall obtained by baseline solutions with results obtained by HT at the association for cognitive patients. All of them expressed in %.

solution due to a high rate of false positives and false negatives related to specular reflections. HT offers the most balanced rates also for this scenario.

4.6.5 Lab

This subsection exposes the results obtained for the different tests performed at the lab scenario.

4.6.5.1 Calibration

Results of the two procedures developed to provide real world coordinates of the trajectories estimated in this scenario are shown in this subsection.

4.6.5.1.1 Distortion correction. Correction of the distortion by radial and tangential factors has been evaluated through Root Mean Square (RMS). RMS value represents corresponding re-projection error that measures the difference between real distance, and the distance measured in pixels from the image captured with ‘fisheye’ lens camera and the corrected one. It can be defined as the geometric error between the point projected and the manually measured one, and it is calculated by applying the next equation:

$$\sum_i = \sqrt{d(x_i, x'_i)^2 + d(x_i, x''_i)^2} \quad (4.40)$$

Different tests have been performed including variations of the rational model, different rotations and distances of the chessboards and different evaluations of its corners. The most relevant are shown in table 4.13

Frames	RMS
20	1.207
30	1.534
40	1.281

Table 4.13: Most relevant RMS results for distortion correction of fisheye lens.

It can be noted that the solution with 20 frames is the one providing the best results with an error slightly higher than 1 pixel which is a reasonable approach.

4.6.5.1.2 From Pixel to cm. Initially the next 10 points were selected in order to build the system previously presented, the results are shown on table 4.14

Id	Ground truth	Distance to the origin	Translation obtained	Error(cm)
0	(0,0)	0	(0.94,2.12)	2.32
1	(-80,-132)	154.35	(-82.32,-135.89)	4.53
2	(160,-12)	160.44	(163.5,-8.36)	5.05
3	(120,188)	223.03	(122.44,192.16)	4.82
4	(-40,308)	310.58	(-40,306)	2
5	(-80,-292)	302.76	(-81.91,-293.69)	2.55
6	(160,-172)	234.91	(160.88,-170.57)	1.68
7	(160,308)	347.07	(160.14,308.08)	1.88
8	(0,188)	188	(-0.09,189.88)	1.88
9	(-120,12)	120.59	(-122.57,-14.16)	3.36

Table 4.14: 10 Points selected in order to perform translation from pixel to cm and corresponding error.

From the previous table it can be noted that the error increases as long as the points are further from the origin (camera location). We selected nine random points in order to confirm that assumption and the results are shown on table 4.15

Pixels (Undistorted image)	Position translated (cm)	Ground truth	Error (cm)
(442,585)	(-33,-199)	(-35,212)	13.15
(504,133)	(-44,238)	(-50,268)	30.59
(528,95)	(-26,279)	(-38,302)	25.94
(539,89)	(-10,355)	(-42,398)	53.6
(661,208)	(117,194)	(132,234)	42.72
(667,233)	(123,171)	(121,192)	21.09
(668,450)	(167,-32)	(160,-42)	12.2
(702,525)	(211,-55)	(202,-152)	9.48
(432,640)	(-39,-250)	(-35,-232)	22.36

Table 4.15: 9 Random points measured in order to evaluate translation from pixel to cm and corresponding error.

Results confirmed previous hypotheses, and therefore the multi-sensing solution implemented will make use of the more accurate calibration given by wireless sensing devices in order to provide measurements in a real coordinate system when the pedestrians move further from the location of the camera.

4.6.5.2 Pedestrian detection

In this section the different tests performed in order to properly explore different solutions to improve detection rate of pedestrians are shown.

4.6.5.2.1 General. The first results are obtained with the HOG descriptor and without performing any distinction between the areas of the room. They can be explored in table 4.16

Sequence	True positives	False positives	False negatives
3Ped	83.47 %	13.33 %	3.18 %
4Ped	87.22 %	8.92 %	3.85 %

Table 4.16: Results for the pedestrian detection of visual tracking on 3Ped and 4Ped at the lab.

The results obtained during this step are the highest provided in this chapter. In this scenario the height of the camera is smaller, the room and number of pedestrians is smaller, and lighting conditions are more controlled than in the retail store. Since the sequences are simpler, and accessibility to this scenario is much easier, more tests could be performed and a deeper analysis of possible techniques to improve results is provided in the next subsections.

4.6.5.2.2 Feature selection. This subsection will analyse the results obtained in the different regions of the scene, in order to provide an improvement for the implemented solution, combining the HOG descriptor with LBP. Initial results for 3Ped and 4Ped only using HOG but dividing the regions can be explored on table 4.17, table 4.18, table 4.19 and table 4.20

Area	True positives	False positives	False negatives
Internal	64.93 %	35.06 %	0 %
Center	86.91 %	12.15 %	0.93 %
External	90.06 %	3.73 %	6.21 %

Table 4.17: Results for the pedestrian detection of HT on 3Ped in the lab with HOG features divided into areas.

Area	Precision	Recall
Internal	64.93 %	100 %
Center	87.73 %	98.93 %
External	96.02 %	93.54 %

Table 4.18: Precision and recall for the pedestrian detection of HT on 3Ped in the lab with HOG features divided into areas.

Area	True positives	False positives	False negatives
Internal	87.01 %	12.98 %	0 %
Center	75.28 %	23.59 %	1.12 %
External	90.51 %	3.98 %	5.50 %

Table 4.19: Results for the pedestrian detection of HT on 4Ped in the lab with HOG features divided into areas.

Area	Precision	Recall
Internal	87.01 %	100 %
Center	76.13 %	98.52 %
External	95.79 %	94.26 %

Table 4.20: Precision and recall for the pedestrian detection of HT on 4Ped in the lab with HOG features divided into areas.

Rates show that the internal zone is the one where the results are the worst. Only in 4Ped the true positives rate is smaller and false positive rate bigger for center region, due to the fact that pedestrians perform several crossing in this region and occlusions are more present there. Results of the internal zone are justified by the fact that in this region only the heads of the pedestrians are visible and not the whole body which is the shape targeted by the HOG descriptor. Recall results are quite satisfactory for both sequences, but precision show some rates that could be improved. Specially when regarding the internal area of Sequence 1 and center area of 4Ped.

Therefore, a combination of HOG and LBP with a dimensional reduction provided by PCA has been tested in this scenario obtaining the results exposed on table 4.21, table 4.22, table 4.23 and table 4.24

Area	True positives	False positives	False negatives
Internal	65 %	35 %	0 %
Center	86.60 %	10.71 %	2.68 %
External	88.34 %	3.07 %	8.59 %

Table 4.21: Results for the pedestrian detection of HT on 3Ped in the lab with HOG and LBP features divided into areas.

Area	Precision	Recall
Internal	65 %	100 %
Center	88.99 %	97 %
External	96.64 %	91.13 %

Table 4.22: Precision and recall for the pedestrian detection of HT on 3Ped in the lab with HOG and LBP features divided into areas.

From the results obtained it can be inferred that the precision of the detection has been slightly improved for all the regions. Recall has slightly decreased in center and external areas. But the difference is not significant. Obtained results do not provide any clear hint regarding necessity of the combination of both feature detectors.

Area	True positives	False positives	False negatives
Internal	49.21 %	9.52 %	41.26 %
Center	77.32 %	10.31 %	12.37 %
External	74.68 %	2.44 %	23.78 %

Table 4.23: Results for the pedestrian detection of HT on 4Ped in the lab with HOG and LBP features divided into areas.

Area	Precision	Recall
Internal	83.78 %	54.38 %
Center	88.23 %	86.2 %
External	96.83 %	75.85 %

Table 4.24: Precision and recall for the pedestrian detection of HT on 4Ped in the lab with HOG and LBP features divided in areas.

The new solution has increased complexity, and decreased the true positives rate for 4Ped which includes more pedestrians on the scene. The rate has just slightly increased in the center region, but comparing previous tests on 4Ped the true positives rate had slightly decreased. Therefore HOG is the only feature descriptor that will be used from now on.

4.6.5.3 Tracking

This subsection will evaluate results obtained from the lab scenario regarding 3Ped and 4Ped.

4.6.5.3.1 General. General results of HT only using the HOG descriptor are provided in this section. Tab. 4.25 and tab. 4.26 show the general accuracy and precision results for this scenario:

Sequence	True positives	False positives	False negatives
3Ped	73.40 %	19.94 %	6.64 %
4Ped	87.78 %	12.01 %	0.20 %

Table 4.25: Results for HT in the sequences at the lab.

Sequence	Precision	Recall
3Ped	78.63 %	91.69 %
4Ped	87.95 %	99.76 %

Table 4.26: Precision and Recall for HT in the sequences at the lab.

Results of 4Ped are satisfactory, but in 3Ped precision and recall rates are lower due to the fact that in this sequence the number of pedestrians is low, and the pedestrians cross their trajectories in the internal region, where the HOG descriptor achieves its lowest performance.

4.6.5.3.2 Areas. Previous results will be deeply analysed in this subsection. For that aim tab. 4.27, tab 4.28, tab. 4.29 and tab 4.30 have been built showing the results obtained for the different regions:

Area	True positives	False positives	False negatives
Internal	52.80 %	47.19 %	0 %
Center	57 %	20.56 %	22.42 %
External	95.15 %	4.84 %	0 %

Table 4.27: Results for HT on 3Ped in the lab with HOG features divided into areas.

Area	Precision	Recall
Internal	52.80 %	100 %
Center	73.49 %	71.76 %
External	95.15 %	100 %

Table 4.28: Precision and recall for HT on 3Ped in the lab with HOG features divided into areas.

Area	True positives	False positives	False negatives
Internal	96.67 %	1.66 %	1.66 %
Center	100 %	0 %	0 %
External	85.31 %	14.68 %	0 %

Table 4.29: Results for HT on 4Ped in the lab with HOG features divided into areas.

Area	Precision	Recall
Internal	98.30 %	98.30 %
Center	100 %	100 %
External	85.31 %	100 %

Table 4.30: Precision and recall for HT on 4Ped in the lab with HOG features divided into areas.

Recall results are very close to 100% except for the center region of 3Ped. Regarding precision internal region of 3Ped presents the worst results, due to the small number of pedestrians and problems related to the initialization. In addition, it can be noted that for 4Ped the hierarchical Kalman Filter has been able to solve the discontinuities in the detection introduced by the false positives in the internal region. This fact could not be achieved for 3Ped since in this sequence the true positives ratio was lower, and pedestrians were crossing their trajectories in this region where the tracking algorithm has its lowest performance. A multi-sensing solution will challenge pending problems.

4.6.5.4 Baseline comparison

Comparison of proposed algorithm with baseline solutions tested can be noted on tab 4.31 and tab 4.32.

	KCF			Kalman			HT		
	TP	FP	FN	TP	FP	FN	TP	FP	FN
3Ped	59.37	20.31	20.31	74.01	12.77	13.21	73.40	19.94	6.64
4Ped	75.09	12.45	12.45	87.82	2.78	9.39	87.78	12.01	0.20

Table 4.31: Comparison of results obtained by baseline solutions with results obtained by HT for Lab scenario. All of them expressed in %.

It should be appreciated that rates for the first sequence are lower because the pedestrians cross their trajectories in this region for that sequence, where the ROI is smaller and there is less information concerning the body of the pedestrian. KCF shows a remarkably lower performance when comparing its results with the ones obtained by Kalman and HT. These two algorithms show a similar performance regarding true positives rate, but the first shows a smaller number of false positives, while the second one does the same for false negatives.

	KCF		Kalman		HT	
	Precision	Recall	Precision	Recall	Precision	Recall
3Ped	74.51	74.51	85.27	84.91	78.63	91.69
4Ped	85.78	85.78	96.92	90.33	87.95	99.76

Table 4.32: Comparison of precision and recall obtained by baseline solutions with results obtained by HT for Lab scenario. All of them expressed in %.

Observing obtained results it can be inferred that Kalman is the most suitable solution for precision, and Solution 2 is the most suitable for recall. A trajectory analysis algorithm on top of them would be suitable to solve those errors, and removing those outliers introduced in the result would be a simpler task, than solving discontinuities in the trajectories obtained. Therefore recall is a more suitable performance indicator for this problem.

4.6.5.5 Multisensing solution

This last subsection is intended to describe results obtained combining different sensing solutions. The procedure followed in previous subsections cannot be followed due to the different nature of the sensing devices. Our work has been based on a trajectory analysis instead of a location for a given time t . Figure 4.23 shows how trajectory estimated by Solution 2 is corrected by the Multisensing solution implemented. This figure demonstrates how the trajectory described by the pedestrian is very similar to the one extracted by HT Solution using only camera acquisition, however combining different sensing devices provides uniformity in the results obtained. It can be observed that extreme variations in the trajectories related to the false positives obtained using just an overhead camera in the center of the lab are corrected solving precision errors previously mentioned, and that trajectory accuracy is slightly improved for the other regions. Displayed figure also reveals that more information regarding the beginning of the trajectory in the lower part, and loop realized in the upper part are achieved through the new solution implemented.

Figure 4.23: Results of trajectories for the lab scenario are shown: red (ground truth), camera (blue) and fusion (green).

4.7 Conclusions

This chapter has presented all the tasks related to a complete Computer Vision and Machine Learning application development:

1. Scenario definition
2. Algorithm analysis
3. Algorithm development
4. Scenario calibration and deployment

5. Samples Acquisition

6. Defining training and testing sets

7. Results

The methodology followed has iterated several times on these steps in order to achieve satisfactory results. Two different approaches for camera perspective have been tested, and two solutions have been developed. The final proposal includes an overhead camera that allows an homogeneous surveillance of a wider area and an algorithm that achieves results close to 100% for the most relevant performance indicator (recall) in the targeted scenarios of this thesis (retail and cognitive disorder surveillance). The second performance indicator calculated (precision) in all the scenarios tested in this solution is over 75%. Finally we have presented a sample where a Multisensing solution combining results obtained with a fingerprinting WSN solves the weakness of the presented solution.

With the proposed solution the ROIs where the human pedestrians are located in the scene have been accomplished, and therefore the system is able to perform a further facial analysis.

Chapter 5

3DWF, FACIAL LANDMARK DETECTION AND HEAD POSE

This chapter is intended to expose the different methods implemented to deploy a proper multi-camera RGBD system, perform 3D face reconstruction, detect facial landmarks and estimate head pose by using Computer Vision and Deep Learning techniques [105]. For that aim, different steps are required in order to gather specific imaging data, perform the corresponding pre-processing steps, augment the 3D data gathered to obtain meaningful 2D features and finally tune and test the neural network architecture chosen.

Two datasets have been used in this chapter. The first of them captures static information of the subjects by using Structured Sensor 3D scanner [4]. The second of them gathers dynamic information (RGBD video) captured by a system composed of 3 Asus Xtion depth cameras [1]. Additionally high resolutions point clouds are obtained by deploying Faro Freestyle 3D scanner [2], and subjects information such as age or gender are annotated as contribution for Research Community in this field. Sensing devices can be noted on fig. 5.1

Figure 5.1: Sensing devices used to capture 3D information.

5.1 3D Wide Faces Dataset 3DWF

This dataset have been recorded in order to acquire multi-camera RGB and depth information from 92 subjects that modify their head pose following the markers sequence proposed.

5.1.1 Characterization

For this dataset age and gender were registered from all the subjects. Statistics of those features can be noted on fig. 5.2

Figure 5.2: Graphics showing the most important features of the subjects included in 3DWF Dataset: age and gender.

We can observe that most of the population is located between 20 and 40 years old due to the fact that the dataset has been recorded in a university. But the dataset covers a wide range of ages. It is also noticeable that gender is a little unbalanced, however looking to the specific graphs of age ranges for every gender it can be observed that the age of females is more balanced than the age of males.

5.1.2 Multi-camera RGBD setup

The setup proposed can be noted in fig. 5.3. Three RGB-D cameras can be observed in the figure 5.3, the one in the middle will be named as frontal camera, and the other ones as side cameras. The number in the box represents the sequence of markers that the subjects were asked to follow (starting in box 1 and ending in box 10). Where W stands for width, H for height and D for depth, and origin is located at the frontal camera for W and D , and at the floor for H .

Figure 5.3: Description of the scenario with the different devices and markers involved for the capture of 3DWF Dataset.

5.1.2.1 Camera initialization

The cameras Asus Xtio PRO Live require USB supply to capture RGB-Depth information. Connecting three cameras of this kind to one machine was the main problem faced in this step. To tackle this problem three independent USB buses are required, and synchronization among them has been implemented to provide a uniform acquisition. The maximum resolution for both RGB and depth frames have been chosen (640x480 pixels), and the maximum frame rate has been selected as well (30fps). This kind of devices allow the registration between RGB and depth sensors, and therefore this procedure has been included in the algorithm proposed. The sequence performed to capture the data has the following steps:

1. Obtain the list of the devices connected.
2. Activate every device.
3. Create the data structures required to capture sensing information.

5.1.2.2 Capture

The procedure followed for every camera involves the following phases:

1. Sensing device captures the RGB image and store it in JPEG format [131].
2. Infrared laser emit a pattern on the scene aided by the diffuser.
3. CMOS sensor capture the points and analyze them. To find the distance from the sensor to every point, it uses triangulation method exposed in figure 5.4. The pattern is known by the device and it is projected to a reference plane whose distance is Z_f . Therefore, if

one object is located closer or further than the reference plane, the displacement will be located on X axis (disparity). Depth information gathered is stored in YAML format [33]

Figure 5.4: Triangulation technique employed to find the distance to the depth sensor.

5.1.3 Devices optimization

Initial tests were mainly based on three parameters:

1. Distance from model to the frontal camera
2. Light source
3. Cameras orientation

Next paragraphs will expose the tests performed in order to optimize the scenario. It should be noted that Camera 1 is the frontal one, and Camera 2 and Camera 3 are the side ones. We should also expose that the differences on grayscale intensities of Camera 2 and Camera 3 are due to the different background of both sides of the scenario.

5.1.3.1 Distance from model to the frontal camera

First parameter evaluated has been the distance from the model to the frontal camera. Manufacturer of the device recommends a distance in the range 80 – 150 cms. Therefore tests have been performed in this range. Results based on the number of points included in the obtained cloud of points from the frontal camera are expressed in table 5.1 and visual results are shown in fig. 5.5

Distance (cm)	PCL Points
80	7754
100	5871
120	4342

Table 5.1: Evaluation of the distance from the model to the frontal camera.

Figure 5.5: Clouds of points obtained with frontal camera for different distances from the model to the camera. Left cloud is gathered with a distance of 80 cm, 100 cm for middle cloud and 120 cm for right cloud

From table 5.1 it can be inferred that the highest resolution is obtained with distance of 80 cm, analogously the best visual appearance shown in fig. 5.5 is obtained with that value, and therefore it is the value chosen.

5.1.3.2 Light source

Once the optimum distance from the model to the camera has been calculated, next step is to determine the parameters related to the LED light source employed whose main features are expressed in table 5.2.

Feature	Value
Power	36W
Color temperature	5800K $\pm$ 300K
Luminous flux	4200 lm

Table 5.2: Main features of the light source employed in 3DWF setup.

Different tests have been performed based on three parameters:

1. Distance. Tests performed for the distance were mainly based on the visual appearance on the cloud of points obtained. For distances smaller than 200 cm the appearance was too bright, and we found that optimum distance should be fixed on 250 cm.
2. Luminous flux. For the luminous flux we based our evaluation on the resolution of the cloud of points obtained for every camera. Results obtained can be noted on table 5.3. It can be inferred that as long as the luminous flux increases, the resolution of the cloud of points decreases. Therefore 250 Lumens (lm) (the minimum provided by the manufacturer) is chosen to define the 3DWF dataset scenario.
3. Orientation. To optimize this parameter the grayscale mean and the Standard deviation (STD) have been evaluated. For that purpose the different angles between the light sources have been explored with a luminous flux of 250 lm. But in this case we should consider also visual appearance, for that aim we have tested light sources pointing to three targets: models, frontal cameras and opposite side cameras. Tests performed pointing to the models

(lm)	Points of the cloud		
	PCL Camera 1	PCL Camera 2	PCL Camera 3
5000	5859	7693	7460
2625	6029	7896	7986
1435	6084	8221	7562
250	6621	8841	8935

Table 5.3: Evaluation of the influence of the luminous flux.

are not presented because their result showed the worst visual results, even though different diffusion filters have been tested on the light source. Other values are shown on the results in table 5.4. In this case the median and the STD have been replaced by the median and the Median Absolute Desviation (MAD), since the median and the mean own a notable difference. Also angles of the cameras are included in the table since they have a big influence on the result, although this parameter will be analyzed in the following subsection.

Focus	Angles		Camera 1		Camera 2		Camera 3	
	Light	Cams	Median	MAD	Median	MAD	Median	MAD
Front	30 °	40 °	189	56	87	58	124	73
Side	40 °	30 °	138	51	89	51	128	62
Front	30 °	60 °	164	50	100	68	122	81

Table 5.4: Evaluation of the orientation of the lighting devices.

Optimum results are obtained for 40° since the difference between the median and the MAD is smaller for that angle.

5.1.3.3 Cameras orientation

In order to test the best orientation of the cameras different angles among them have been tested. All optimum parameters exposed previously have been deployed on the scenario to test this parameter. The RGB-D devices chosen capture the scene affected by all the parameters previously exposed, and therefore we believe it might be the last parameter to be tested, and the most critical since it determines the field of view. First tests were carried out in order to determine the region of interest to be covered for the subjects involved in the experiment. Visual results for the surface covered of the cameras can be noted on table 5.6

Figure 5.6: Surface covered with the different angles of the cameras. Left image corresponds to 30 °, middle image to 40 °, and right image to 60 °

From the images exposed it can be noted that the minimum angle required to cover completely the face of the subjects is 30° and as long as the angles among the cameras are increased, surface covered increases as well. Finally, grayscale values obtained with the RGB sensor are evaluated in order to capture the faces with similar color intensities. Results are expressed on table 5.5

Angle	Camera 1		Camera 2		Camera 3	
	Mean	STD	Mean	STD	Mean	STD
30 °	129.35	62.69	94.22	66.38	118.35	72.43
40 °	172.00	71.09	98.29	71.37	122.51	75.80
60 °	171.50	71.90	108.674	79.94	124.57	85.75

Table 5.5: Evaluation of the orientation of the cameras.

With this table we can observe that as long as the angle among the cameras is increased, the difference between the mean and the STD is also increased, therefore we can conclude that the optimum angle among the cameras is 30°. All reconstruction procedure, and devices location is summarized on fig. 5.7

Figure 5.7: Graphical exposition of the procedure followed to reconstruct the 3D models upon the depth and RGB information captured by the 3 RGB-D devices.

5.1.4 3D Reconstruction algorithm

In order to test the visual accuracy of the set up proposed for the experiment an algorithm based on the combination of depth and color intensities captured has been implemented whose steps can be noted on the block diagram exposed in figure 5.8

Figure 5.8: Block diagram to describe the different steps required for the multi-camera 3D reconstruction algorithm.

5.1.4.1 Registration

Initially a cloud of points has been built for every frame stored in every camera (Cl_{Left} , $Cl_{Frontal}$ and Cl_{Right}). To build those clouds spatial information is gathered from the depth frames captures. Z coordinate corresponds to the numerical value (multiplied by a scale factor) stored in the depth frames. X and Y coordinates correspond to the position in the matrix stored by the depth frame. Finally these coordinates will determine the color intensity extracted from the

RGB frame stored as well. Therefore at this point we will have three clouds, but stored from a different viewpoint every one of them. Visual results can be noted in fig. 5.9

Figure 5.9: Initial clouds of points obtained from left, frontal and right cameras.

Next step determines rotation matrix and translation vector $\{R_{Left}, t_{Left}\}, \{R_{Right}, t_{Right}\}$. Both matrices are intended to locate the data gathered in the same coordination system. For that purpose every cloud from side cameras will be registered with the cloud from the frontal camera by using the following transformation matrix:

$$T = \begin{bmatrix} r_{1,1} & r_{1,2} & r_{1,3} & t_1 \\ r_{2,1} & r_{2,2} & r_{2,3} & t_2 \\ r_{3,1} & r_{3,2} & r_{3,3} & t_3 \\ a_{4,1} & a_{4,2} & a_{4,3} & a_{4,4} \end{bmatrix}$$

Where $r_{i,j} \forall \{i, j\} \in \{1 \dots 3\}$ compose the rotation matrix R and $t_i \forall i \in \{1 \dots 3\}$ compose the translation vector t . Finally to have an affine transformation the last row of the matrix is replaced by the last row of the identity matrix:

$$T_{affine} = \begin{bmatrix} r_{1,1} & r_{1,2} & r_{1,3} & t_1 \\ r_{2,1} & r_{2,2} & r_{2,3} & t_2 \\ r_{3,1} & r_{3,2} & r_{3,3} & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

To obtain the optimum values of the transformation matrix for the system developed we could choose between an automatic or a manual procedure. Initial tests for an automatic solution based on keypoint detection and feature descriptors did not provide satisfactory results, and therefore manual point selection was chosen. In our set-up cameras are fixed, and therefore we only need to select points once to obtain a proper registration. Markers were not used in the face of the model for this procedure since it has some recognizable parts that can easily be matched. Ten points have been selected from every cloud (two by two matching). To obtain the transformation matrix we have built an homogeneous transformation, using frontal point cloud as reference ($Cl_{Frontal}$):

$$f(q_i) = R p_i + t \quad (5.1)$$

Obtaining a origin matrix with a point in every row $p_i = (x_i, y_i, z_i, 1)$ where $i \in \{1 \dots 10\}$ from Cl_{Right} and Cl_{Left} and a target matrix with a point in every row $q_i = (x_i, y_i, z_i, 1)$ where $i \in \{1 \dots 10\}$ from $Cl_{Frontal}$.

And to overcome accuracy errors in manual annotation and obtain the optimum values for $R_{left,right}$ and $T_{left,right}$ we have employed Random Sample Consensus (RANSAC) [34]. Then the initial clouds (Cl_{Left} , Cl_{Right}) are transformed towards the reference frontal cloud ($Cl_{Left'}$, $Cl_{Right'}$) and resulting clouds are added two by two to obtain the complete cloud Cl_{Total} . A sample of Cl_{Total} is shown on fig. 5.10

Figure 5.10: Sample of a complete cloud of points from different perspectives.

5.1.4.2 Refinement

The second step for reconstruction is based on Iterative Closest Point (ICP) [14] algorithm. ICP is used to minimize difference between sets of geometrical points such as segments, triangles or parametric curves. In our work, we use the point-to-point approach. Metric distance between the origin cloud (Cl_{Left} and Cl_{Right}) and target cloud ($Cl_{Frontal}$) is minimized by the following equation:

$$i = \arg \min_i \|p_i - q_i\|^2, \quad (5.2)$$

Where p_i is the point belonging to the origin cloud (Cl_{Left} and Cl_{Right}) and q_i is a point belonging to the target cloud ($Cl_{Frontal}$). Regarding rotation and translation matrix, the algorithm iterates over the minimum square distances by:

$$R, t = \sum_{i=1}^N \arg \min_{R, T} \|(Rp_i + t) - q_i\|^2, \quad (5.3)$$

Where N is the number of iterations fixed to 30 for our solution and we have also fixed the percentage of worst candidate removal to 90 %. With this refinement $Cl_{Total'}$ is obtained.

5.1.4.3 ROI

Once the whole cloud is built and refined, we use the Dlib face detector [51] on the RGB image from the frontal camera in order to determine the region of interest (ROI). In a similar way, we apply the face landmark detection based on VanillaCNN exposed in section 5.3 obtaining the locations of facial landmarks. To project the facial landmarks obtained from the neural network,

we use Perspective Projection Model [82]. By applying the following equations:

$$X_k = -\frac{Z_k}{f}(x_k - x_c + \delta_x), Y_k = -\frac{Z_k}{f}(y_k - y_c + \delta_y), \quad (5.4)$$

Where X_k , Y_k and Z_k are the projected coordinates in the cloud, x_c and y_c are the coordinates of the center of the 2D image, x_k and y_k are the input coordinates from the 2D image and δ_x and δ_y are the parameters to correct the distortion of the lens provided by the manufacturer. Obtaining 3D projection for the bounding box delimiters to project the cropped cloud obtained after refinement $Cl_{Total'}$ into a cloud with mostly facial properties Cl_{F_0} .

In order to test the accuracy of Cl_{F_0} we have considered the clouds gathered with Faro Freestyle 3D Laser Scanner (Cl_{HD}) as ground truth, and we have measured the average minimum distance from $\forall pt_i \in Cl_{F_0} \in Marker1$ to $\forall pt'_i \in Cl_{HD}$ for every subject, obtaining as result distances in the range [16 – 23] mm. In addition, we should consider:

1. Since the distance from the subject to the camera is below 1 m, the error of the depth sensor should be in the range [5 – 15] mm according to the results exposed in [38].
2. The faces of the subjects are not rigid (although both captures have been performed on a neutral pose).

Therefore, the range measured as distance from Cl_{F_0} to Cl_{HD} proves the accuracy of the 3D reconstruction performed.

5.1.4.4 Noise filtering

In this section, an algorithm to filter Cl_{F_0} is proposed to obtain reliable face clouds. The following features are proposed:

1. **Color.** Initially we need to delimit two areas:
 - (a) 2D ROI' to extract. We have employed facial landmarks detected by VanillaCNN through data augmentation procedure presented in subsections 5.2.2 and 5.3 respectively. In our setup we have detected five points: left eye (*le*), right eye (*re*), nose (*n*), left mouth (*lm*) and right mouth (*rm*). A new ROI (*ROI'*) is defined based on a bounding box with these detections:

$$\{(x_{le}, y_{le}), (x_{re}, y_{re}), (x_{lm}, y_{lm}), (x_{rm}, y_{rm})\}$$

ROI'_{RGB} intensities are transformed to a more uniform color space: CIELAB [45]. Components values of the two intensities samples used for thresholding are calculated in the following manner following a normal distribution:

$$th_{i_{Lab}} = (\hat{L}_{ROI'} \pm w\sigma_{L_{ROI'}}, \hat{a}_{ROI'} \pm w\sigma_{a_{ROI'}}, \hat{b}_{ROI'} \pm w\sigma_{b_{ROI'}}) \quad (5.5)$$

Where $\hat{L}_{ROI'}$ and $\sigma_{L_{ROI'}}$ are the mean and standard deviation of L component from CIELAB color space for the new ROI defined. Analogously for $\hat{a}_{ROI'}$, $\sigma_{a_{ROI'}}$, $\hat{b}_{ROI'}$ and $\sigma_{b_{ROI'}}$. And w is fixed to 0.75 in our implementation.

- (b) 3D Contour Cl_{F_0} . In this case we have defined two margins for width and height from Cl_{F_0} to filter farther points to the cloud centroid by applying CIEDE2000 $\forall pt_i \in Cl_{F_0}$:

$$if \Delta E_{00}^* \{(\hat{L}_{ROI'}, \hat{a}_{ROI'}, \hat{b}_{ROI'}), (L_{pt_i}, a_{pt_i}, b_{pt_i})\} < \Delta E_{00}^*(th_{1_{Lab}}, th_{2_{Lab}}) \Rightarrow pt_i \in Cl_{F_{FC}} \quad (5.6)$$

Where ΔE_{00}^* is the metric used in CIEDE2000 and $Cl_{F_{FC}}$ is the point cloud obtained after color filtering.

2. **Depth.** Mainly focused on noise introduced by depth sensors and outliers from color filtering. For that aim we have built a confidence interval based on normal distribution of $Cl_{F_{FC}}$ $[\hat{Z}_{Cl_{F_{FC}}} - w_Z \sigma, \hat{Z}_{Cl_{F_{FC}}} + w_Z \sigma]$. Where w_Z is fixed to 2.25 in our implementation.

5.1.4.5 Uniform distribution

To provide a reliable point cloud dataset, it is important that clouds have similar resolutions and that every part of the cloud is constant regarding point-space density. For that reason, Cl_{F_F} is divided into four parts based on its width and height. A resolution of 2K points is proposed as target resolution. Therefore, every cloud part should have $2K/4$ points. An example of the split performed for Marker 1 can be observed in 5.11. An implementation of voxel grid downsampling [135] based on a dynamic radius search is used. The voxel grid filter down-samples the data by taking a spatial average of the points in the cloud through employing rectangular areas that are known as voxels. The set of points that lie within the bounds of a voxel are assigned to that voxel and will be combined into one output point. With this final step Cl_F is composed, and sample values for one subject are displayed in fig. 5.12. In an analogous way one sample for Marker 1 without texture mapping is shown in fig. 5.13.

Figure 5.11: 3D face parts split for uniform sampling

Figure 5.12: 2D images extracted from the final face clouds proposed by this work. Clouds exposed corresponds to one subject at the different poses (1-10) established by the markers of the scenario.

Figure 5.13: 2D images extracted from the final face clouds for Marker 1 without texture mapping proposed by this work.

5.2 UVA Dataset

Dataset provided by University of Amsterdam (UVA) has been captured by rotating structured sensor device in order to reconstruct the 3D models of 300 subjects. 3D reconstruction of the data gathered is out of the scope of this work, in our case we will focus on the deep learning method to extract valuable features from meshes already processed.

5.2.1 Characterization

This massive data capture did not allow gathering of additional features of the subjects such as age, gender or height. Therefore the subjects have been classified based on their appearance. Classification is enumerated below:

1. Gender: Male, Female.
2. Age: Kids, young persons, adults and elder.

The distribution of the proposed classes is shown on fig. 5.14. It is noticeable that adult and young classes are the ones with a highest population due to their accessibility. The lowest population is for elder people, but we believe 20 subjects are still relevant for this work. Gender rates offer a balanced population.

Figure 5.14: Graphics showing the most important features of the subjects included in UVA dataset: age and gender.

Finally subjects are divided in three datasets following deep learning paradigm for the training and testing steps required in this kind of development:

1. Training set: 245 subjects (81.67%).
2. Validation set: 40 subjects (13.34%).
3. Testing set: 15 subjects (5%).

5.2.2 Data annotation and augmentation

This section will expose the methodology followed to annotate ground truth values on the UVA dataset, and procedures followed to perform the geometric transformations that allowed extraction of 2D images from 3D meshes changing the viewpoint. The whole annotation and augmentation procedure is summarized on figure 5.15

Figure 5.15: Flow diagram followed to pre-process the UVA Dataset.

5.2.2.1 Facial landmark annotation

Initially 50 facial landmarks were annotated for 300 subjects stored in the UVA Dataset. Those points that could not be clearly identified, were not annotated. An example of an annotated subject is presented on figure 5.16

Figure 5.16: Sample of the annotation of the facial landmarks for the UVA Dataset.

5.2.2.2 Data augmentation

Once the ground truth facial landmarks are available, the data augmentation was performed on meshes in order to gather images annotated with those landmarks. For that aim we have used raycasting [93] to perform projection of the 3D mesh to a 2D image. By implementing this technique we have moved the view plane in front of the pinhole to remove the inversion, graphical explanation can be noted on fig. 5.17. If an object point is at distance z_0 from the viewpoint, and has y coordinate y_0 , then its projection y_p onto the viewplane will be determined by the ratios of sides of similar triangles: $(0, 0), (0, z_p), (y_p, z_p)$, and $(0, 0), (0, z_0), (y_0, z_0)$. So we have:

$$\frac{y_p}{z_p} = \frac{y_0}{z_0} \quad (5.7)$$

Figure 5.17: Raycasting geometry model with a plane and a pinhole extracted from [93].

Therefore we have modified viewpoints of the virtual camera based on the following parameters and values:

1. **Pitch.** $[-30^\circ, 30^\circ]$ Interval: 5°
2. **Yaw.** $[-30^\circ, 30^\circ]$ Interval: 5°
3. **Roll .** $[-30^\circ, 30^\circ]$ Interval: 5°
4. **Distance.** $[110\text{cm}, 160\text{cm}]$ Interval: 10cm

5.2.2.3 Bounding box calculation

Next step in order to have visual data in the optimum shape to apply neural networks is to automatically determine the ROI for the augmented data. This step has been based on the landmarks already annotated. Procedure is detailed below:

1. Longitudinal line. Longitudinal centered line ($Long_{Center}$) is calculated by applying linear least-squares regression [133] on coordinates of the following facial landmarks: jaw bottom, mouth bottom, mouth top, under nose, nose tip, center and hair line. Therefore with this procedure we obtain the following line equation:

$$y = m_{Long}x + b_{Long_{Center}} \quad (5.8)$$

2. Transversal line. Transversal centered line ($Trans_{Center}$) is perpendicular to longitudinal line, therefore its slope can be calculated in the following way:

$$m_{Trans} = \frac{-1}{m_{Long}} \quad (5.9)$$

And since $Trans_{Center}$ passes by center facial landmark we can obtain independent value of the line equation by solving following equation:

$$b_{Trans_{Center}} = y_{Center} - m_{Trans} x_{Center} \quad (5.10)$$

3. Side Margins. $Marg_{Left}$ and $Marg_{Right}$ have been calculated by employing m_{Trans} and binary mask from figure 5.15. For left margin this step decreases the value of X coordinate of points belonging to $Trans_{Center}$ until intensity values of 5 consecutive points are 0 in the binary mask. The same procedure is applied for right margin, but increasing the value of X coordinate.

4. Straight Margins. $Marg_{Bottom}$ is calculated based on jaw bottom facial landmark annotation, and $Marg_{Top}$ is calculated in the same manner than $Marg_{Left}$ and $Marg_{Right}$, but employing m_{Long} and decreasing Y values.

All elements previously mentioned for bounding box calculation are graphically exposed on fig. 5.18

Figure 5.18: Sample bounding box calculation for UVA Dataset, and graphical exposition of geometric elements used for that purpose.

5.3 Facial landmark detection

This section explains deep learning solutions adopted for facial landmark detection.

5.3.1 VanillaCNN

Solution exposed in [137] has been explored in order to improve state of the art results for facial landmark detection.

5.3.1.1 Architecture

The main peculiarity of this network is the tweaking model oriented to two main processes :

1. It performs and specific clustering in the intermediate layers by a representation which discriminate between differently aligned faces. And with that information it can train pose specific landmark regressors.
2. Remaining weights from the first dense layer output are fine-tuned by selecting only the group of images classified in the same cluster with the features from the intermediate layers.

The network consists of four convolutional layers (denoted $CL_1, \dots, CL_4$) with intermittent max pooling layers (stride=2). These are followed by a fully connected (dense) layer, FC_5 , which is then fully connected to an output with $2xm$ values for the m landmark coordinates: $P = (p_1, \dots, p_m) = (x_1, y_1, \dots, x_m, y_m)$. Architecture can be noted on figure 5.19

Figure 5.19: Architecture of VanillaCNN neural network [137].

Absolute hyperbolic tangent is used as an activation function and Adam is used for training optimization [62]

5.3.1.2 Loss and error

L2 normalized by the inter-ocular distance ($Dist_{inter}$) is implemented as the network loss:

$$\varphi(P_i, \hat{P}_i) = \frac{\|P_i - \hat{P}_i\|_2^2}{\|\hat{p}_{i,1} - \hat{p}_{i,2}\|_2^2} \quad (5.11)$$

Where P_i is the $2xm$ vector of predicted coordinates for a training image, $\hat{P}_i$ their ground truth locations, and $\hat{p}_{i,1}, \hat{p}_{i,2}$ the reference eye positions.

Localization error is measured as a fraction of the inter-ocular distance (analogously to loss), a measure invariant to the actual size of the images. We declare a point correctly detected if the pixel error is below $0.1 \times Dist_{inter}$.

5.3.1.3 Results

For this subsection initial tests have been performed by joining augmented training set and validation set from UVA as training Dataset, and testing set as validation set. To build those tests a maximum range of augmented data and a maximum number of subjects per view have been established. Those subjects are randomly selected in every view.

Finally some tests have been performed on the Annotated Facial Landmarks in the wild (AFLW [64]) dataset by using fine-tuning with pre-trained weights obtained just training with the UVA dataset.

5.3.1.3.1 Max range: $|10^\circ|$. To test the suitability of the architecture of the network a first test has been performed. Specific features of this training are named below:

1. Number of views: 875
2. Training subjects per view: 50
3. Validation subjects per view: 12

4. Learning rate: 10^{-4}

Some hyperparameters and best results are specified in table 5.6.

Test	Batch size		Validation Param.		Test Result	
	Training	Validation	Iter	Interval	Iteration	Error %
10Max	300	150	3570	30K	1095K	7.08

Table 5.6: Parameters and results using only UVA Dataset with viewpoint of maximum 10° for pitch, yaw and roll.

Accuracy is above 90% and therefore the architecture is validated. More details can be obtained from training curve of figure 5.20

Figure 5.20: Learning curve for Test1 using only UVA Dataset with viewpoint of maximum 10° for pitch, yaw and roll. Red curve represents training loss, green one validation loss and blue one testing accuracy.

5.3.1.3.2 Max range: $|30^\circ|$. In order to use learned weights to train network with standard Faces in the Wild datasets range has increased to 30° . Training details are enumerated below:

1. Number of views: 15379
2. Training subjects per view: 35
3. Validation subjects per view: 12
4. Learning rate: 10^{-4}

Some hyperparameters and best results are specified in table 5.7.

Rates have decreased but still they are over 85 %. More details can be obtained from training curve of figure 5.21.

Test	Batch size		Validation Param.		Test Result	
	Training	Validation	Iter	Interval	Iteration	Error %
30Max	300	150	3570	30K	1475K	13.57

Table 5.7: Parameters and results using only UVA Dataset with viewpoint of maximum 30° for pitch, yaw and roll.

Figure 5.21: Learning curve for Test1 using only UVA Dataset with viewpoint of maximum 30° for pitch, yaw and roll. Red curve represents training loss, green one validation loss and blue one testing accuracy.

5.3.1.3.3 AFLW. For this section training and validation sets for the pre-trained networks are the same ones than the used in the previous subsection. For fine tuning network AFLW training and testing (used as validation set) sets are used. Last 1000 images of testing set were removed, and used to obtain testing results expressed in the tables and figures.

1. Training images: 9000
2. Validation images: 3000
3. Testing images: 1000
4. Learning rate: 10^{-4}

Some hyperparameters and best results are specified in table 5.8.

Test	Batch size		Validation Param.		Test Result	
	Training	Validation	Iter	Interval	Iteration	Error %
AFLW	30	10	2.5K	4K	40K	5.8

Table 5.8: Parameters and results using UVA Dataset with viewpoint of maximum 30° for pitch, yaw and roll for pre-training and AFLW for fine-tuning.

Error rate is quite satisfactory and overcomes results exposed in [137]. Results can be explored more precisely in figure 5.22.

Figure 5.22: Learning curve using UVA Dataset with viewpoint of maximum 30° for pitch, yaw and roll for pre-training and AFLW for fine-tuning. Red curve represents training loss, green one validation loss and blue one testing accuracy.

5.3.2 Recombinator Neural Network RCN

Second selected network [50] performs learning by using landmark independent feature maps. In this case instead of performing an specific learning, a more purely statistical approach is performed. The output of each branch is upsampled, then concatenated with the next level branch with one degree of finer resolution. Therefore the main novelty is that branches pass more information to each other during training letting the network learn how to combine them non-linearly to maximize the log likelihood of the landmarks. It is only at the end of the R th branch that the feature maps are converted into a per-landmarks scoring representation by implementing a softmax.

5.3.2.1 Architecture

RCN performs learning by using landmark independent feature maps. Per-landmark scoring representation are only learned at the end of R th branch on which a softmax is then applied. This type of architecture needs more data (from both coarse and fine layers) to be transferred from one layer to another during training process. Architecture proposed in exposed in figure 5.23. P,C represents a pooling layer followed by a convolutional layer, U represents an upsampling layer, and K represents concatenation of two sets of feature maps along the feature map dimension. All convolutions are 3x3 and all poolings are 2x2. All upsamplings are by a factor of 2. All convolutional layers are followed by ReLU non-linearity except the one right before the softmax.

This architecture is trained globally using gradient backpropagation to minimize the sum of negated conditional log probabilities of all N training (input-image,landmark-locations) pairs, for all M landmarks (x_m, y_m) , with an additional regularization term for the weights calculated through the next equation:

$$L(W) = \frac{1}{N} \sum_{n=1}^N \sum_{k=1}^K -\log P(Y_k = y_k^{(n)} | X = x^{(n)}) + \lambda \|W\|^2 \quad (5.12)$$

Where W are the network parameters to minimize within regularization term to minimize and λ their weight. Loss error is calculated analogously to VainillaCNN.

Landmark independent feature maps learned in the initial stages of this network are reliable for a high accuracy for the different natures of data proposed, but it may not provide noticeable increasing on the accuracy for a fine-tuning approach.

Figure 5.23: Architecture of RCN neural network [50].

In summary, we have selected two architectures that alternate convolution and max-pooling layers, but whose nature is completely different. VanillaCNN presents 4 convolution layers and 2 dense layers. Dense layers are interlaid by a discrimination among the clusters previously learned from midnetwork features in a specific pose manner. RCN presents a bidirectional architecture with different branches including 3-4 convolution layers whose results are concatenated in the end of each branch to the inputs of the next one. VanillaCNN presents a descending size of filtering sizes along the network and RCN keeps it fixed.

5.3.2.2 Results

Histogram shown on figure 5.24 expose the error for the different combinations of networks and datasets tested. It can be noted that the lowest error rate obtained is training RCN just with the UVA dataset. Massive normalized 2D projections of this dataset learned in a bidirectional approach achieved the lowest error rate.

Figure 5.24: Results obtained for both pipelines of face landmark detection with different combinations of datasets.

In addition, we should also mention that network layers based on midnetwork features proposed by VanillaCNN achieve the worst results with the same training and testing data since this is a specific solution for another nature of data. But weights and bias pre-learned from UVA dataset increases the performance of the algorithm, and helps to achieve the best results for AFLW after its finetuning. In this case massive data initialize properly midnetwork features so that the network can go beyond in global minimum target for common datasets of landmark detection such as AFLW. This procedure can be noted with the visualization of the filters ex-

pressed on fig. 5.25 and fig. 5.26 for last convolution and max-pooling layers. Exploring both of them it can be noted that initial training of UVA provides blunter features at this stage of the learning due to the triangulation procedure to gather mesh input data. It can be inferred as well, that the fine-tuning process provides sharper features that increases the performance of the network.

Figure 5.25: Representation of the last layers of convolution filters from VainillaCNN. Left image corresponds to the training using UVA dataset, and right image corresponds to the fine-tuning using AFLW.

Figure 5.26: Representation of the last layers of max-pooling filters from VainillaCNN. Left image corresponds to the training using UVA dataset, and right image corresponds to the fine-tuning using AFLW.

5.4 Head pose classification

This section is intended to describe the methods implemented to validate the alignment of the head pose values of the data gathered in 3DWF dataset with markers located in the scene. For that aim we have used visual information of subjects when they look at marker 1 (relaxed pose looking to the front) as reference for the other markers. Therefore, this marker has been deleted from the results that will be exposed. Steps carried out for that purpose are graphically defined in figure 5.27. Initial steps are analogous to the ones proposed in subsection 5.1.4.3. In this case we have reversely used projection equations expressed in (5.4) together with 2D Euclidean distance to gather closest points included in $Cl_{Total'}$ to 2D facial landmarks detected by Vanilla CNN. Therefore, a new set composed by 3D facial landmarks is calculated:

$$\begin{aligned} & (X_{le}, Y_{le}, Z_{le}), (X_{re}, Y_{re}, Z_{re}), (X_n, Y_n, Z_n), \\ & (X_{lm}, Y_{lm}, Z_{lm}), (X_{rm}, Y_{rm}, Z_{rm}) \end{aligned}$$

Figure 5.27: Block diagram of the different steps involved in head pose estimation.

5.4.1 Rigid motion

The initial transformation has been performed by implementing Least-Squares Rigid Motion Using SVD [117] from the set of 3D facial landmarks to obtain corresponding rotation matrix. Let $P = p_1, p_2, \dots, p_n$ where p_i are 3D coordinates of the facial landmarks for marker $1 \in R^3$ and $Q = q_1, q_2, \dots, q_n$ where q_i are 3D coordinates of the facial landmarks for markers 2-10 $\in R^3$ be our reference and target sets of data respectively. We are able to find a rigid transformation that optimally aligns the two sets in the least squares sense, i.e., assuming unity vector for the translation matrix (subjects are static in the experiment proposed):

$$R = \underset{R \in SO(3)}{\operatorname{argmin}} \sum_{i=1}^n w_i \|Rp_i - q_i\|^2 \quad (5.13)$$

Restating the problem such that the translation would be zero:

$$\begin{aligned} x_i &= p_i - \bar{p} \\ y_i &= q_i - \bar{q} \end{aligned} \quad (5.14)$$

Simplifying the expression exposed in equation (5.13):

$$\begin{aligned} \|Rp_i - q_i\|^2 &= x_i^T x_i - 2y_i^T R x_i + y_i^T y_i = \\ &= \underset{R \in SO(3)}{\operatorname{argmin}} (-2 \sum_{i=1}^n w_i y_i^T R x_i) = \\ &= \underset{R \in SO(3)}{\operatorname{argmax}} \sum_{i=1}^n w_i y_i^T R x_i = \\ &= \operatorname{tr}(WY^T R X) \end{aligned} \quad (5.15)$$

Where $W = \operatorname{diag}(w_1, \dots, w_n)$ is an $n \times n$ diagonal matrix with the weight w_i on diagonal entry i , Y is the $d \times n$ matrix with y_i as its columns and X is the $d \times n$ matrix with x_i as its columns. tr is the trace of a square matrix (sum of the elements on the diagonal) and owes commutative property with respect to product. Therefore, we are looking for a rotation R that maximizes $\operatorname{tr}(RXWY^T)$. Now we have denoted $d \times d$ covariance matrix $S = XWY^T$. If we take Single Value Decomposition (SVD) of S such that $S = U \Sigma V^T$:

$$\operatorname{tr}(RXWY^T) = \operatorname{tr}(RS) = \operatorname{tr}(RU \Sigma V^T) = \operatorname{tr}(\Sigma V^T RU) \quad (5.16)$$

Note that V , R and U are all orthogonal matrices, so $V^T RU$ is also an orthogonal matrix and we can assume identity. Therefore:

$$R = VU^T \quad (5.17)$$

5.4.2 Euler Angles

R can be expressed as a sequence of three rotations, one about each principle axis. Since matrix multiplication does not commute, the order of the axes which one rotates about will affect the result. For this analysis, we have assumed first about the x-axis, then the y-axis, and finally the z-axis. Such a sequence of rotations can be represented as the matrix product:

$$R = R_Z(\phi)R_Y(\theta)R_X(\psi)$$

Given a rotation matrix R , we can compute the Euler angles, ϕ , θ , ψ by equating each element in R with the corresponding element in the matrix product $R_Z(\phi)R_Y(\theta)R_X(\psi)$. This results in nine equations that have been used to find the Euler angles.

5.4.3 Classification

In order to validate dataset proposed we have assumed Euler angles calculated previously as head pose angles (pitch (ϕ), yaw (ψ) and roll (θ)). We have plot the data obtained from the markers 2-10 and the 92 subjects captured on the 3DWF dataset. Three 2D projections of this data can we noted on fig. 5.28. It can be inferred that the distribution of the data is quite uniform, and therefore alignment of the point clouds reconstructed, and projection of the 2D landmarks can be validated through a classification method.

Figure 5.28: Graphical plot of Euler Angles obtained for the different markers of 3DWF.

5.4.3.1 Validation methods

Given a set of samples $(x_{1Pitch}, x_{1Yaw}, x_{1Roll}), (x_{2Pitch}, x_{2Yaw}, x_{2Roll}), \dots, (x_{nPitch}, x_{nYaw}, x_{nRoll})$, where $x_{iPitch}, x_{iYaw}, x_{iRoll} \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ and their class labels $y_1, y_2, \dots, y_m$ where $y_k \in [2, 9] \in \mathbb{Z}$, in order to provide the quantitative results of a method to perform the head pose classification two Machine Learning approaches based on supervised learning have been adopted:

1. Linear Discriminant Analysis LDA[43]
2. Gaussian Naive Bayes GNB[80]

5.4.3.1.1 LDA. Multi-class LDA is based on the analysis of two scatter matrices: within-class scatter matrix and between-class scatter matrix. The within-class scatter matrix is defined as:

$$S_w = \sum_{i=1}^n (x_i - \hat{x}_{y_k})(x_i - \hat{x}_{y_k})^T \quad (5.18)$$

Where $\hat{x}_{y_k}$ is the sample mean (Euler Angles) of the k -th class, and x_i is composed by $(x_{i_{Pitch}}, x_{i_{Yaw}}, x_{i_{Roll}})$. The between-class scatter matrix is defined as:

$$S_b = \sum_{k=1}^m n_k (\hat{x}_{y_k} - \hat{x})(\hat{x}_{y_k} - \hat{x})^T \quad (5.19)$$

Where m is the number of classes, $\hat{x}$ is the overall sample mean, and n_k is the number of samples in the k -th class. Then, multi-class LDA can be formulated as an optimization problem to find a set of linear combinations (with coefficients w) that maximizes the ratio of the between-class scattering to the within-class scattering, as

$$\hat{w} = \underset{w}{\operatorname{argmax}} \frac{w^T S_b w}{w^T S_w w} \quad (5.20)$$

The solution is given by the following generalized eigenvalue problem:

$$S_b w = \lambda S_w w \quad (5.21)$$

Generally, at most $m-1$ generalized eigenvectors are useful to discriminate between m classes.

5.4.3.1.2 GNB. Bayes Theorem describes the probability of an event, based on prior knowledge of conditions be related of conditions to the event. Mathematically you can write the Bayes theorem as following for the stated problem:

$$P(y_k|x_i) = \frac{P(x_i|y_k)P(y_k)}{\sum_{l=1}^m P(y_l|x_i)P(y_l)} \quad (5.22)$$

Where $P(y_k|x)$ is the posterior probability for hypothesis y_k given the data x_i . The conditional probability can be decomposed as:

$$p(y_k|x_i) = \frac{p(y_k)p(x_i|y_k)}{p(x_i)} \quad (5.23)$$

The problem can be reformulated by adopting Naive Bayes, assuming that each instance of Euler Angles x_i is conditionally independent of every other feature x_j for $j \neq i$, given the marker y_k :

$$p(x_i|y_k) = p(x_i|x_{i+1}, \dots, x_n, y_k) \quad (5.24)$$

Thus, the joint model can be expressed as

$$\begin{aligned} p(y_k|x_1, \dots, x_n) &= p(y_k)p(x_1 | y_k) p(x_2 | y_k) p(x_3 | y_k) \dots \\ &= p(y_k) \prod_{i=1}^n p(x_i|y_k) \end{aligned} \quad (5.25)$$

Once the naive Bayes probabilistic model has been explored, a decision rule is required to build a classifier. The corresponding classifier, a Bayes classifier, is the function that assigns a class label $\hat{y} = y_k$ for some k as follows:

$$\hat{y} = \underset{k \in \{1, \dots, m\}}{\operatorname{argmax}} p(y_k) \prod_{i=1}^n p(x_i|y_k) \quad (5.26)$$

If the nature of the data can be analysed in a continuous manner (like in our proposal) a Gaussian distribution can be assumed. In our case, assumed probability density function of the normal (Gaussian) distribution is given by:

$$f(x) = \frac{1}{\sqrt{2\hat{x}\sigma^2}} \exp\left(\frac{-(x - \hat{x})^2}{2\sigma^2}\right) \quad (5.27)$$

Where $\hat{x}$ represents the mean of Euler Angles, and σ^2 their variance. By calculating the probability density for x_i for class y_k as $p(x = x_i|y_k)$, we can plug x_i to the Gaussian distribution equation with parameters $\hat{x}_k$ and σ_k^2 in the following way:

$$p(x = x_i|y_k) = \frac{1}{\sqrt{2\hat{x}\sigma^2}} \exp\left(\frac{-(x_i - \hat{x}_k)^2}{2\sigma_k^2}\right) \quad (5.28)$$

5.4.4 Results

The aim of this classification methods is to the validate data capture, and the projection of facial landmarks to point clouds gathered. Results and main features of these methods can be noticed on table 5.9.

Method	Training samples	Testing samples	Training accuracy	Testing accuracy
LDA	80%	20%	82 %	83 %
GNB	80%	20%	84%	84%

Table 5.9: Results for head pose classification.

Confusion matrix for GNB classification, where rows correspond to ground truth markers (2-10) and columns to predicted markers in the same range, is shown in figure 5.29. Results show promising values for a classification technique such as GNB. It can be noticed that those markers where the subjects are looking to one side of the scene (such as 2 or 5) are the most complex to predict, and those markers where the subjects are looking straight and modifying their pitch (such as 10) the simplest.

Figure 5.29: Confusion Matrix for Head Pose Classification.

5.5 Conclusions

We have presented an optimized multi-camera RGB-D system for facial properties to capture accurate and reliable data. In this scope we have performed a data collection including 92 people fulfilling the need of a 3D facial dataset able to exploit capabilities of deep learning paradigm in 3D scope. In addition, we provide a complete pipeline to process data collected and pose a challenge for Computer Vision and Machine Learning Research Community by annotating human characteristics such as age or gender. RGB-D streams collected allows other related tasks such as face tracking or 3D reconstruction with a wide source of visual information that improves performance of common acquisition systems for extreme head poses.

In this scope we found facial landmark detection one of the main tasks where our work should contribute to research lines that project 3D information into a more feasible and less costly domain such as 2D. For that reason we have proposed an innovative data augmentation method, tested and discussed its accuracy on two state of the art deep learning solutions. We have trained and evaluated synthetic and visual imaging data on two complementary architectures, finding a combined solution that improves results for a very common deep net architecture like Vanilla CNN.

Finally, the alignment of the path proposed to the subjects by ten markers is validated by implementing a geometric approach for head-pose through previously estimated features. The refinement of the learning techniques implemented for this task is one of the lines of research proposed for future work.

Chapter 6

CONCLUSIONS AND FUTURE WORK

In this dissertation we have presented several ICT, Computer Vision & Machine Learning solutions to model human behaviour, specifying two application domains: retail and health monitoring. We have proved that current technology is advanced enough to assist professionals from both areas in their decision-making. However, there exist HW limitations that are out of the scope of this work, and there are opened research lines that may enhance proposed behaviour modeling, they will be posed in this chapter.

Several tasks have been performed to raise the conclusions exposed in this chapter:

1. We have formed expert teams by enforcing collaborative environments.
2. We have adapted the techniques employed to design agile architectures for our problem.
3. We have implemented Computer Vision and Machine Learning methods in two programming languages (C++ and Python) by employing different frameworks/libraries (OpenCV, Point Cloud Library, Caffe, Theano and TensorFlow) relying on different kinds of visual imaging devices (HD camera, overhead camera, RGB-D sensor and 3D scanner).
4. We have employed modeling methodologies focused on human behaviour understanding.

From a high level of abstraction the architectures presented respond to the necessities of the experts consulted in both fields, at a reasonable cost and deployment. The main concerns at the beginning were related to convince the experts on both fields about the reliability of the technology currently being developed. The technical team had to enforce the idea that this approach is not aimed to replace any expert on the field, its objective is to provide them specific tools whose output is intelligent information with a high degree of accuracy. This structured information can lead to a larger knowledge of human behaviour that can benefit different application areas such as retail or health monitoring. In terms of this collaboration, another challenge faced was the completely different terminologies employed by the technical team, and the expert teams from the application domains involved.

Once the most relevant features for every domain had been specified, there exists several kinds of sensing technologies that should be contemplated in order to obtain reliable data inferences for the ambit where this dissertation is located. The combination of heterogeneous sensing devices is a topic with countless applications in our days. Its development does not only concern

signal processing, also organization, structure and communications of the technologies developed have a great significance. Various concepts such as latency, communications or data processing are considered and discussed in Chapter 3 to validate the architectures proposed. A detailed deployment and definition of the requirements of the devices involved in the signal acquisition process are also contemplated. Every sensing device provide a different nature of data intended to estimate behavioural properties of the humans in the scenes defined. The exposition is directed towards a proper balance between some performance indicators such as efficacy or feasibility. Future research lines in this context may provide quantitative results to find an optimum trade off between the latency and the HW resources required for the implementation of the data flow in terms of efficiency for the proposed architecture. Acceptance of the devices at the scenarios defined is another critic aspect to be evaluated by patients/caregivers and visitors of the retail stores.

From a low level of abstraction, two Computer Vision development cycles are completed (deployment of the devices, data acquisition, implementation of the algorithms, results and discussion). All of them required the installation of frameworks related, compilation of libraries and the corresponding linking procedures to use the methods contained. Automatic localization of humans in the proposed scenarios provides the platform to perform a deeper analysis of human behaviour. In our approach we evaluate the algorithms implemented in two real scenarios, and we perform an extended analysis of the different aspects with a significant influence on these kind of systems in a controlled environment. Video signal captured is processed by using feature descriptors and Machine Learning classifiers whose performance is discussed in Chapter 4, proposing a complete solution that supports an accurate trajectory estimation. For all Computer Vision based systems available there is a big concern: the light source. In our tests we could observe that depending on the time of the day when the tests were performed and the presence or not artificial light the results obtained can own a large variance. Specific properties derived from human pedestrian tracking such as occlusion or scale invariance has a great impact on the results as well. Deployment on real scenarios adds a great value to this dissertation, but there are some constraints derived from this fact. The infrastructure of the scenarios is not the most suitable in order to perform this kind of installations, and there are some privacy constraints that shall be taken into account as well. Spaces are limited in terms of dimensions, and network and electricity are not always available on the most suitable place for signal processing.

In the selection of the visual imaging devices intended to perform a proper acquisition we have contemplated different kind of technologies. For human pedestrian tracking methods we initially tested a simple HD camera, followed by a device mostly used for visual surveillance systems (overhead camera). The approach had to be completely modified in order to obtain promising results when the optics of the sensing device and the perspective were completely altered. This type of device provides a wider field of view (in spite of image distortion that can be corrected by the calibration method provided) that allows for monitoring larger areas such as the ones related to the use cases proposed. Distortion introduced by the eyefish lens provoked different results for the regions of the scenario. In addition, tedious training procedures have been completed by annotating a large amount of images, and cropping the most relevant regions. Moreover, synchronization of different natures of devices has been another complex task achieved. Measurements for localization gathered by a WSN are not in the same scale/domain than the ones provided by a Computer Vision based algorithm. Alignment and calibration of both systems was achieved through several trials and concept re-formulation. Results gathered for human pedestrian tracking opens a new gate for the optimization of this kind of human tracking systems by the integration of different kind of signals, and its embedding in devices more suitable for their deployment such as Field-Programable Gate Arrays (FPGA) or Raspberry Pies.

The second cycle for visual imaging technologies is more complex and advanced. Recent

advances in Computer Vision and Machine Learning have directed the Research Community towards building large collections of annotated data in order to improve the performance of different application domains. In an analogous manner, 3D imaging has arrived via the introduction of low-cost devices and the development of efficient 3D reconstruction algorithms. In Chapter 5, we focus on 3D facial imaging by proposing a multi-camera RGB-D setup that opens up many possibilities to extract human properties from facial analysis. The implementation of this kind of setup demands very high HW resources, and consequently a complex synchronization among them. For the proposed setup, all the devices are connected through USB interfaces, and running simultaneously. Those facts caused a lot of problems related to the buses where the data flow. Definition of the scenario with elements such as light sources, tripods and subjects with the corresponding geometric features such as distances or angles between them required a large amount of tests and modifications to obtain the optimal solution. In addition, the large amount of data collected involves scheduling a lot of appointments and a campaign to attract as many subjects as possible. Inside this procedure the collection should be carried out very carefully concerning different aspects such as position of the cameras, position of the subjects, luminous flux of the light sources . . . Because otherwise some captures are not normalized or aligned with the previous ones, and they shall be repeated to obtain valuable data.

In our case we have focused on three tasks: 3D reconstruction, facial landmarks estimation and head pose classification. For the first we present a complete pipeline that exploits geometrical features of the multi-camera system deployed, and makes a significant contribution in terms of massive data collection, and cloud normalization for the next generation of deep learning algorithms using 3D information as source of data. Registration of the visual data gathered by the multi-camera setup required another annotation procedure where some results did not represent properly visual appearance of the person captured, and some subjects had to be repeated more than once. The setup optimized and the acquisition process completed has provided a vast amount of information, but also a lot of noise embedded in the captured signal. To provide clouds with only facial information several tests and developments have been carried out by employing several 3D techniques related to features such as spatial properties, color, distance to the neighbors or density at every region of the cloud.

It is known that deep learning methods are data hungry. Hence, there exists a data-demanding problem of deep networks, and tedious data annotation procedures related. For facial landmarks estimation, we employ 3D meshes of 300 faces of subjects that allows for a proper augmentation method whose parameters are Euler angles (pitch, yaw and roll). With this method, we only need to annotate every model once, but then we can generate thousands of augmented images for the different head poses. This was the procedure with a highest demand of HW resources. We needed to deploy the program developed to perform raycasting augmentation on several computers, and still it took more than one month to generate millions of images with different values for pitch, yaw and roll in order to cover a wide range of viewpoints. Dealing with that big amount of data is a process very sensitive and every test required a considerable time to obtain initial results. Initial tests were mainly concerning the range of pitch, yaw and roll that started at the interval $[-50^\circ, 50^\circ]$ and ended at the interval $[-30^\circ, 30^\circ]$. The most relevant datasets publicly available for this task have been obtained, and the corresponding methods to parse them have also been implemented, deriving in some tests where the information was not properly processed and results were not consistent with the techniques implemented. In addition, neural networks are very complex algorithms whose optimization involves a lot of parameters whose dependencies are not intuitive. Training procedures associated to deep learning are really expensive in terms of HW sources, and there are lot of dependencies for frameworks such as Theano or Caffe that need to be solved. Moreover, installation of GPUs is a very specific procedure that involves many components of a computer, and versioning is a quite common problem to perform a proper

installation. We have evaluated an innovative data augmentation method and evaluated state of the art deep net architectures, that have yielded very promising results

Head pose estimation is a topic widely explored with applications such as autonomous driving, focus of attention modeling or emotion analysis. For that reason we have combined the results of the first and second tasks by two common classification approaches in the field of Machine Learning that validates the data collected. Head pose estimation is highly correlated with facial landmark detection (especially in 3D domain) and we have proven that with a good performance in facial landmarks, head pose can easily be approached. Here we have checked the path followed for every subject according to the markers located on the scene, and we have annotated the frames relevant for the extreme poses included in our work. Labeling the data is a very laborious process that can easily lead to results that are not meaningful at all. Even though labeling and parsing has been properly performed, human properties such as age, gender or ethnicity determine different pose to look at the same marker from the same position.

Finally Appendix A and Appendix B define a knowledge base where insights of all ICT and Signal Processing methods are stored to ease decision-making for the experts involved in the use cases analysed. For data storage it is needed to delimit properly the values of all the fields included in the model, and if the model is for a specific application like the one presented in the Appendixes, very specific terminology needs to be collected and studied. Cardinalities among the entities of the model are another element that shall be carefully reviewed, and that demands a high knowledge on the field of application.

The implementation of user-friendly interfaces for the collection of the data that should be manually introduced in the system, and the systems able to capture the interaction between the user and the interface implemented is another future line of work introduced by this dissertation.

In addition, this dissertation has locally offered the possibility to be included in a technological research group enforcing the capabilities to be proactive in a research hub environment. Working in direct collaboration with a group of researchers enthusiastic about technology development is very profitable and opens the possibility to interact with very advanced topics. This research environment has offered the possibility to be enrolled in research projects, attending to scientific meetings/conferences, and to globally interact with highly skilled professionals with relevant backgrounds. Those collaborations embedded in a research methodology result in a very enriching human experience for the candidate.

Bibliography

- [1] Asus xtion. https://www.asus.com/3D-Sensor/Xtion_PRO//. Accessed: 2018-05-10.
- [2] Faro freestyle 3dx. <https://www.faro.com/es-es/productos/construccion-bim-cim/faro-scanner-freestyle3d-x/>. Accessed: 2018-05-10.
- [3] Science buddies. steps of the scientific method. http://www.sciencebuddies.org/mentoring/project_scientific_method.shtml/. Accessed: 2019-01-09.
- [4] Structured sensor. <https://structure.io//>. Accessed: 2018-05-10.
- [5] 3dMD LLC. 3dmd. <http://www.3dmd.com>, 2004. [Online; accessed 10-December-2018].
- [6] J. Ajay, C. Song, A. Wang, J. Langan, Z. Li, and W. Xu. A pervasive and sensor-free deep learning system for parkinsonian gait analysis. In *2018 IEEE EMBS International Conference on Biomedical Health Informatics (BHI)*, pages 108–111, March 2018.
- [7] F. Alvarez, M. Popa, V. Solachidis, G. Hernández-Peñaloza, A. Belmonte-Hernández, S. Asteriadis, N. Vretos, M. Quintana, T. Theodoridis, D. Dotti, and P. Daras. Behavior analysis through multimodal sensing for care of parkinson’s and alzheimer’s patients. *IEEE Multi-Media*, 25(1):14–25, Jan 2018.
- [8] S. Avidan. Ensemble tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(2):261–271, Feb 2007.
- [9] Andrew D. Bagdanov, Alberto Del Bimbo, and Iacopo Masi. The florence 2d/3d hybrid face dataset. In *Proceedings of the 2011 Joint ACM Workshop on Human Gesture and Behavior Understanding, J-HGBU ’11*, pages 79–80, New York, NY, USA, 2011. ACM.
- [10] S. Banerjee, N. I. Ghali, A. Roy, and A. E. Hassanein. A bio-inspired perspective towards retail recommender system: Investigating optimization in retail inventory. In *2012 12th International Conference on Intelligent Systems Design and Applications (ISDA)*, pages 161–165, Nov 2012.
- [11] A. Belmonte-Hernández, G. Hernández-Peñaloza, F. Álvarez, and G. Conti. Adaptive fingerprinting in multi-sensor fusion for accurate indoor tracking. *IEEE Sensors Journal*, 17(15):4983–4998, Aug 2017.
- [12] B. Benfold and I. Reid. Stable multi-target tracking in real-time surveillance video. In *CVPR 2011*, pages 3457–3464, June 2011.
- [13] J. Berclaz, F. Fleuret, E. Turetken, and P. Fua. Multiple object tracking using k-shortest paths optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(9):1806–1819, Sept 2011.

- [14] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, Feb 1992.
- [15] A. Bhattacharyya. On a measure of divergence between two statistical populations defined by their probability distributions. *Bulletin of the Calcutta Mathematical Society*, 35:99–109, 1943.
- [16] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '99*, pages 187–194, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co.
- [17] Guido Borghi, Marco Venturelli, Roberto Vezzani, and Rita Cucchiara. Poseidon: Face-from-depth for driver pose estimation. *CoRR*, abs/1611.10195, 2016.
- [18] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, Oct 2001.
- [19] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. Van Gool. Online multiperson tracking-by-detection from a single, uncalibrated camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(9):1820–1833, Sept 2011.
- [20] Xudong Cao, Yichen Wei, Fang Wen, and Jian Sun. Face alignment by explicit shape regression. *Int. J. Comput. Vision*, 107(2):177–190, April 2014.
- [21] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, 2017.
- [22] LeadingAge CAST. Telehealth and remote patient monitoring for longterm and postacute care: A primer and provider selection guide. Technical report, LeadingAge Center for Aging Services Technologies, Washington, DC, 2015.
- [23] R. T. Collins, Yanxi Liu, and M. Leordeanu. Online selection of discriminative tracking features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1631–1643, Oct 2005.
- [24] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Mach. Learn.*, 20(3):273–297, September 1995.
- [25] CSN. Adaboost. <http://blog.csdn.net/autocyzy/article/details/51305999>, 2016. [Online; accessed 29-September-2017].
- [26] R. Cucchiara, C. Grana, M. Piccardi, A. Prati, and S. Sirotti. Improving shadow suppression in moving object detection with hsv color information. In *ITSC 2001. 2001 IEEE Intelligent Transportation Systems. Proceedings (Cat. No.01TH8585)*, pages 334–339, 2001.
- [27] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*, CVPR '05, pages 886–893, Washington, DC, USA, 2005. IEEE Computer Society.
- [28] O. Demigha, W. Hidouci, and T. Ahmed. On energy efficiency in collaborative target tracking in wireless sensor network: A review. *IEEE Communications Surveys Tutorials*, 15(3):1210–1222, Third 2013.

- [29] M. Dyshel, D. Arkadir, H. Bergman, and D. Weinshall. Quantifying levodopa-induced dyskinesia using depth camera. In *2015 IEEE International Conference on Computer Vision Workshop (ICCVW)*, pages 511–518, Dec 2015.
- [30] Allard M Husky M Catheline G et al.. Mobile technologies in the early detection of cognitive decline. *PLoS ONE*, 9(12):1–10, Dec 2014.
- [31] G. Fanelli, J. Gall, and L. Van Gool. Real time head pose estimation with random regression forests. In *CVPR 2011*, pages 617–624, June 2011.
- [32] Z. Feng, P. Huber, J. Kittler, P. Hancock, X. Wu, Q. Zhao, P. Koppen, and M. Raetsch. Evaluation of dense 3d reconstruction from 2d face images in the wild. In *2018 13th IEEE International Conference on Automatic Face Gesture Recognition (FG 2018)*, pages 780–786, May 2018.
- [33] Martin Fenner. One-click science marketing. *Nature Materials*, 11(4):261–263, mar 2012.
- [34] Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, June 1981.
- [35] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Additive logistic regression: a statistical view of boosting. *Annals of Statistics*, 28:2000, 1998.
- [36] Nir Friedman, Dan Geiger, and Moises Goldszmidt. Bayesian network classifiers. *Mach. Learn.*, 29(2-3):131–163, November 1997.
- [37] X. Gao, Y. Su, X. Li, and D. Tao. A review of active appearance models. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 40(2):145–158, March 2010.
- [38] H. Gonzalez-Jorge, B. Riveiro, E. Vazquez-Fernandez, J. Martínez-Sánchez, and P. Arias. Metrological evaluation of microsoft kinect and asus xtion sensors. *Measurement*, 46(6):1800 – 1806, 2013.
- [39] Ralph Gross, Iain Matthews, Jeffrey Cohn, Takeo Kanade, and Simon Baker. Multi-pie. *Image Vision Comput.*, 28(5):807–813, May 2010.
- [40] Saeed Hamine, Emily Gerth-Guyette, Dunia Faulx, B. Beverly Green, and Sarah Amy Ginsburg. Impact of mhealth chronic disease management on treatment adherence and patient outcomes: A systematic review. *J Med Internet Res*, 17(2):e52, Feb 2015.
- [41] J. V. Harrison and A. Andrusiewicz. Enhancing digital advertising using dynamically configurable multimedia. In *Multimedia and Expo, 2003. ICME '03. Proceedings. 2003 International Conference on*, volume 1, pages I–717–20 vol.1, July 2003.
- [42] J. A. Hartigan and M. A. Wong. A K-means clustering algorithm. *Applied Statistics*, 28:100–108, 1979.
- [43] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001.
- [44] Kristiina Häyrynen, Kaija Saranto, and Pirkko Nykänen. Definition, structure, content, use and impacts of electronic health records: A review of the research literature. *International journal of medical informatics*, 77 5:291–304, 2008.

- [45] Songhua He and Qiao Chen. *The Color Appearance Attributes Analysis of CIELAB Color Space*, pages 353–359. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.
- [46] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista. High-speed tracking with kernelized correlation filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(3):583–596, March 2015.
- [47] G. Hernández-Peñaloza, A. Belmonte-Hernández, M. Quintana, and F. Álvarez. A multi-sensor fusion scheme to increase life autonomy of elderly people with cognitive problems. *IEEE Access*, PP(99):1–1, 2017.
- [48] N. Herodotou, K. N. Plataniotis, and A. N. Venetsanopoulos. A color segmentation scheme for object-based video coding. In *1998 IEEE Symposium on Advances in Digital Filtering and Signal Processing. Symposium Proceedings (Cat. No.98EX185)*, pages 25–29, Jun 1998.
- [49] Jeffrey Hightower and Gaetano Borriello. Location systems for ubiquitous computing. *Computer*, 34(8):57–66, August 2001.
- [50] Sina Honari, Jason Yosinski, Pascal Vincent, and Christopher J. Pal. Recombinator networks: Learning coarse-to-fine feature aggregation. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 5743–5752, 2016.
- [51] Rein-Lien Hsu, M. Abdel-Mottaleb, and A. K. Jain. Face detection in color images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):696–706, May 2002.
- [52] P.G. Jacobs. *Remote assessment of health in older and frail adults living independently through mobility assessment*, pages 61–84. Science Publishers, 2013.
- [53] L. Jia and L. Tong. Day ahead dynamic pricing for demand response in dynamic environments. In *52nd IEEE Conference on Decision and Control*, pages 5608–5613, Dec 2013.
- [54] I. T. Jolliffe. *Principal Component Analysis and Factor Analysis*, pages 115–128. Springer New York, New York, NY, 1986.
- [55] P. E. JUPP and K. V. MARDIA. A general correlation coefficient for directional data and related regression problems. *Biometrika*, 67(1):163–173, 1980.
- [56] Z. Kalal, K. Mikolajczyk, and J. Matas. Tracking-learning-detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(7):1409–1422, July 2012.
- [57] Tunç Güven Kaya and Engin Firat. Multi-view face detection with one classifier for video analytics systems. In Cosimo Distanto, Sebastiano Battiato, and Andrea Cavallaro, editors, *Video Analytics for Audience Measurement*, pages 97–108, Cham, 2014. Springer International Publishing.
- [58] J. M. Keller, M. R. Gray, and J. A. Givens. A fuzzy k-nearest neighbor algorithm. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-15(4):580–585, July 1985.
- [59] Vladimir Khryashchev, Andrey Priorov, and Alexander Ganin. Online audience measurement system based on machine learning techniques. In Cosimo Distanto, Sebastiano Battiato, and Andrea Cavallaro, editors, *Video Analytics for Audience Measurement*, pages 111–122, Cham, 2014. Springer International Publishing.

- [60] D. Kim, J. Choi, J. T. Leksut, and G. Medioni. Accurate 3d face modeling and recognition from rgb-d stream in the presence of large pose changes. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 3011–3015, Sept 2016.
- [61] H. Kim, J. Lee, S. Lee, X. Cui, and H. Kim. Sensor fusion-based human tracking using particle filter and data mapping analysis in in/outdoor environment. In *2013 10th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, pages 741–744, Oct 2013.
- [62] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.
- [63] J. Kröckel and F. Bodendorf. Customer tracking and tracing data as a basis for service innovations at the point of sale. In *2012 Annual SRII Global Conference*, pages 691–696, July 2012.
- [64] M. Köstinger, P. Wohlhart, P. M. Roth, and H. Bischof. Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 2144–2151, Nov 2011.
- [65] Mars Lan, Lauren Samy, Nabil Alshurafa, Myung-Kyung Suh, Hassan Ghasemzadeh, Aurelia Macabasco-O’Connell, and Majid Sarrafzadeh. Wanda: An end-to-end remote health monitoring and analytics system for heart failure patients. In *Proceedings of the Conference on Wireless Health, WH ’12*, pages 9:1–9:8, New York, NY, USA, 2012. ACM.
- [66] Ha A. Le and Ioannis A. Kakadiaris. Uhd31: A dataset for better understanding face recognition across pose and illumination variation. In *The IEEE International Conference on Computer Vision (ICCV) Workshops*, Oct 2017.
- [67] David D. Lewis. Naive (bayes) at forty: The independence assumption in information retrieval. In Claire Nédellec and Céline Rouveirol, editors, *Machine Learning: ECML-98*, pages 4–15, Berlin, Heidelberg, 1998. Springer Berlin Heidelberg.
- [68] B. Li, C. Yang, Q. Zhang, and G. Xu. Condensation-based multi-person detection and tracking with hog and lbp. In *2014 IEEE International Conference on Information and Automation (ICIA)*, pages 267–272, July 2014.
- [69] Dongqing Li, editor. *HSV Color Space*, pages 793–793. Springer US, Boston, MA, 2008.
- [70] H. B. Li, W. Wang, H. W. Ding, and J. Dong. Mining paths and transactions data to improve allocating commodity shelves in supermarket. In *Proceedings of 2012 IEEE International Conference on Service Operations and Logistics, and Informatics*, pages 102–106, July 2012.
- [71] M. H. Li, T. A. Mestre, S. H. Fox, and B. Taati. Automated vision-based analysis of levodopa-induced dyskinesia with deep learning. In *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 3377–3380, July 2017.
- [72] Y. Li, H. Ai, T. Yamashita, S. Lao, and M. Kawade. Tracking in low frame rate video: A cascade particle filter with discriminative observers of different life spans. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(10):1728–1740, Oct 2008.

- [73] Yangyan Li, Rui Bu, Mingchao Sun, and Baoquan Chen. Pointcnn, 2018. cite arxiv:1801.07791Comment: Small updates.
- [74] Shengcai Liao, Xiangxin Zhu, Zhen Lei, Lun Zhang, and Stan Z. Li. Learning multi-scale block local binary patterns for face recognition. In Seong-Whan Lee and Stan Z. Li, editors, *Advances in Biometrics*, pages 828–837, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.
- [75] Daniele Liciotti, Marco Contigiani, Emanuele Frontoni, Adriano Mancini, Primo Zingaretti, and Valerio Placidi. Shopper analytics: A customer activity recognition system using a distributed rgb-d camera network. In *VAAM@ICPR*, 2014.
- [76] X. Liu, W. Liang, Y. Wang, S. Li, and M. Pei. 3d head pose estimation with convolutional neural network trained on synthetic images. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 1289–1293, Sept 2016.
- [77] Ping Luo. Hierarchical face parsing via deep learning. In *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, CVPR ’12, pages 2480–2487, Washington, DC, USA, 2012. IEEE Computer Society.
- [78] C. Ma, J. Huang, X. Yang, and M. Yang. Hierarchical convolutional features for visual tracking. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 3074–3082, Dec 2015.
- [79] Satu-Marja Mäkelä, Sari Järvinen, Tommi Keränen, Mikko Lindholm, and Elena Vildjiounaite. Shopper behaviour analysis based on 3d situation awareness information. In Cosimo Distanto, Sebastiano Battiato, and Andrea Cavallaro, editors, *Video Analytics for Audience Measurement*, pages 134–145, Cham, 2014. Springer International Publishing.
- [80] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*, pages 253–278. Cambridge University Press, New York, NY, USA, 2008.
- [81] A. Markov. Extension of the Limit Theorems of Probability Theory to a Sum of Variables Connected in a Chain. In R. Howard, editor, *Dynamic Probabilistic Systems (Volume I: Markov Models)*, chapter Appendix B, pages 552–577. John Wiley & Sons, Inc., New York City, 1971.
- [82] Steve Marschner and Peter Shirley. *Fundamentals of Computer Graphics, Fourth Edition*. A. K. Peters, Ltd., Natick, MA, USA, 4th edition, 2016.
- [83] Geoffrey J McLachlan. *Discriminant Analysis and Statistical Pattern Recognition*, pages 793–793. Wiley, Newark, NJ, 2005.
- [84] R. Mehra. On the identification of variances and adaptive kalman filtering. *IEEE Transactions on Automatic Control*, 15(2):175–184, April 1970.
- [85] A. Y. Meigal, K. S. Prokhorov, N. A. Bazhenov, L. I. Gerasimova-Meigal, and D. G. Korzun. Towards a personal at-home lab for motion video tracking in patients with parkinson’s disease. In *2017 21st Conference of Open Innovations Association (FRUCT)*, pages 231–237, Nov 2017.
- [86] J. Melià-Seguí and R. Pous. Human-object interaction reasoning using rfid-enabled smart shelf. In *2014 International Conference on the Internet of Things (IOT)*, pages 37–42, Oct 2014.

- [87] A. Milan, K. Schindler, and S. Roth. Multi-target tracking by discrete-continuous energy minimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(10):2054–2068, Oct 2016.
- [88] James Munkres. Algorithms for the assignment and transportation problems, 1957.
- [89] Leann Myers and Maria J. Sirois. *Spearman Correlation Coefficients, Differences between*. American Cancer Society, 2014.
- [90] Leann Myers and Maria J. Sirois. *Spearman Correlation Coefficients, Differences between*. American Cancer Society, 2014.
- [91] H. Nam and B. Han. Learning multi-domain convolutional neural networks for visual tracking. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4293–4302, June 2016.
- [92] N. Noury, T. Herve, V. Rialle, G. Virone, E. Mercier, G. Morey, A. Moro, and T. Porcheron. Monitoring behavior in home using a smart fall sensor and position sensors. In *1st Annual International IEEE-EMBS Special Topic Conference on Microtechnologies in Medicine and Biology. Proceedings (Cat. No.00EX451)*, pages 607–610, 2000.
- [93] University of Clemson. Raycasting. <https://people.cs.clemson.edu/~dhouse/courses/405/notes/raycast.pdf>, 2016. [Online; accessed 20-November-2018].
- [94] Timo Ojala, Matti Pietikäinen, and Topi Mäenpää. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(7):971–987, July 2002.
- [95] Kenji Okuma, Ali Taleghani, Nando de Freitas, James J. Little, and David G. Lowe. *A Boosted Particle Filter: Multitarget Detection and Tracking*, pages 28–39. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.
- [96] OpenCV. Introduction to Support Vector Machines. http://docs.opencv.org/2.4/doc/tutorials/ml/introduction_to_svm/introduction_to_svm.html, 2011-2014. [Online; accessed 27-September-2017].
- [97] M. Ozuysal, M. Calonder, V. Lepetit, and P. Fua. Fast keypoint recognition using random ferns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(3):448–461, March 2010.
- [98] Massimiliano Patacchiola and Angelo Cangelosi. Head pose estimation in the wild using convolutional neural networks and adaptive gradient methods. *Pattern Recognition*, 71:132 – 143, 2017.
- [99] M. Pavel, A. Adami, M. Morris, J. Lundell, T. L. Hayes, H. Jimison, and J. A. Kaye. Mobility assessment using event-related responses. In *1st Transdisciplinary Conference on Distributed Diagnosis and Home Healthcare, 2006. D2H2.*, pages 71–74, April 2006.
- [100] R. Pierdicca, D. Liciotti, M. Contigiani, E. Frontoni, A. Mancini, and P. Zingaretti. Low cost embedded system for increasing retail environment intelligence. In *2015 IEEE International Conference on Multimedia Expo Workshops (ICMEW)*, pages 1–6, June 2015.
- [101] S. Pradhan, A. Balashankar, N. Ganguly, and B. Mitra. (stable) virtual landmarks: Spatial dropbox to enhance retail experience. In *2014 Sixth International Conference on Communication Systems and Networks (COMSNETS)*, pages 1–8, Jan 2014.

- [102] A. Procházka, M. Schätz, O. Ľupa, M. Yadollahi, O. Vysata, and M. Walls. The ms kinect image and depth sensors use for gait features detection. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 2271–2274, Oct 2014.
- [103] Charles Ruizhongtai Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 77–85, 2017.
- [104] M. Quintana, J.M. Menendez, F. Alvarez, and J.P. Lopez. Improving retail efficiency through sensing technologies: A survey. *Pattern Recognition Letters*, 81(Supplement C):3 – 10, 2016.
- [105] Marcos Quintana, Sezer Karaoglu, Federico Alvarez, Jose Manuel Menendez, and Theo Gevers. Three-d wide faces (3dwf): Facial landmark detection and 3d reconstruction over a new rgb-d multi-camera dataset. *Sensors*, 19(5), 2019.
- [106] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.
- [107] Robert Ravník, Franc Solina, and Vesna Zabkar. Modelling in-store consumer behaviour using machine learning and digital signage audience measurement data. In Cosimo Distanto, Sebastiano Battiato, and Andrea Cavallaro, editors, *Video Analytics for Audience Measurement*, pages 123–133, Cham, 2014. Springer International Publishing.
- [108] Christos Sagonas, Epameinondas Antonakos, Georgios Tzimiropoulos, Stefanos Zafeiriou, and Maja Pantic. 300 faces in-the-wild challenge. *Image Vision Comput.*, 47(C):3–18, March 2016.
- [109] N. Sano, K. Yada, and T. Suzuki. Category evaluation method for business intelligence using a hierarchical bayes model. In *2014 IEEE 13th International Conference on Cognitive Informatics and Cognitive Computing*, pages 400–407, Aug 2014.
- [110] Anita Sant’Anna. *A Symbolic Approach to Human Motion Analysis Using Inertial Sensors : Framework and Gait Analysis Study*, pages 561–606. Science Publishers, 2013.
- [111] Jason M. Saragih, Simon Lucey, and Jeffrey F. Cohn. Deformable model fitting by regularized landmark mean-shift. *Int. J. Comput. Vision*, 91(2):200–215, January 2011.
- [112] Eric See-To Savvas Papagiannidis and Michael Bourlakis. Virtual test-driving: The impact of simulated products on purchase intention. *Journal of Retailing and Consumer Services*, 21 5:877–887, 2014.
- [113] S. Sedai, M. Bennamoun, D. Q. Huynh, A. El-Sallam, S. Foo, J. Alderson, and C. Lind. 3d human pose tracking using gaussian process regression and particle filter applied to gait analysis of parkinson’s disease patients. In *2013 IEEE 8th Conference on Industrial Electronics and Applications (ICIEA)*, pages 1636–1642, June 2013.
- [114] H. Ben Shitrit, J. Berclaz, F. Fleuret, and P. Fua. Tracking multiple people under global appearance constraints. In *2011 International Conference on Computer Vision*, pages 137–144, Nov 2011.

- [115] A. V. Shpakov, A. V. Voronov, E. V. Fomina, N. Yu. Lysova, M. V. Chernova, and I. B. Kozlovskaya. Comparative efficiency of different regimens of locomotor training in prolonged space flights as estimated from the data on biomechanical and electromyographic parameters of walking. *Human Physiology*, 39(2):162–170, Mar 2013.
- [116] A. W. M. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah. Visual tracking: An experimental survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7):1442–1468, July 2014.
- [117] Olga Sorkine-Hornung and Michael Rabinovich. Least-squares rigid motion using svd. https://ig1.ethz.ch/projects/ARAP/svd_rot.pdf, 2017. [Online; accessed 27-November-2018].
- [118] Chris Stauffer and W. E. L. Grimson. Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 2, pages 246–252 Vol. 2, Los Alamitos, CA, USA, August 1999. IEEE.
- [119] Thomas G. Stockham, Jr. High-speed convolution and correlation. In *Proceedings of the April 26-28, 1966, Spring Joint Computer Conference*, AFIPS '66 (Spring), pages 229–233, New York, NY, USA, 1966. ACM.
- [120] Yi Sun, Xiaogang Wang, and Xiaoou Tang. Deep convolutional network cascade for facial point detection. In *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '13, pages 3476–3483, Washington, DC, USA, 2013. IEEE Computer Society.
- [121] A. Sylvain, A. Doniec, R. Mandiau, and S. Lecoeuche. Purchase intention based model for a behavioural simulation of sale space. In *2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)*, volume 3, pages 318–324, Aug 2014.
- [122] A. Sánchez-Rico, P. Garel, I. Notarangelo, M. Quintana, G. Hernández-Peñaloza, S. Asteriadis, M. Popa, N. Vretos, V. Solachidis, and M. Burhos. Ict services for life improvement for the elderly. *Studies in health technology and informatics*, 242(1):600–605, 2017.
- [123] Sabine Süsstrunk, Robert Buckley, and Steve Swen. Standard rgb color spaces. In *In The Seventh Color Imaging Conference: Color Science, Systems, and Applications*, pages 127–134, 1999.
- [124] Y. Tian, P. Luo, X. Wang, and X. Tang. Deep learning strong parts for pedestrian detection. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1904–1912, Dec 2015.
- [125] Y. Tian, P. Luo, X. Wang, and X. Tang. Pedestrian detection aided by deep learning semantic tasks. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5079–5087, June 2015.
- [126] Alexandros T. Tzallas, Markos G. Tsipouras, Georgios Rigas, Dimitrios G. Tsalikakis, Evaggelos C. Karvounis, Maria Chondrogiorgi, Fotis Psomadellis, Jorge Cancela, Matteo Pastorino, MarAa Teresa Arredondo Waldmeyer, Spiros Konitsiotis, and Dimitrios I. Fotiadis. Perform: A system for monitoring, assessment and management of patients with parkinson disease. *Sensors*, 14(11):21329–21357, 2014.

- [127] Ashok Vegesna, Melody L Tran, Michele Angelaccio, and Steve Arcona. Remote patient monitoring via non-invasive digital technologies: A systematic review. In *Telemedicine journal and e-health : the official journal of the American Telemedicine Association*, 2017.
- [128] F. Viola, A. D’Elia, D. Korzun, I. Galov, A. Kashevnik, and S. Balandin. The m3 architecture for smart spaces: Overview of semantic information broker implementations. In *2016 19th Conference of Open Innovations Association (FRUCT)*, pages 264–272, Nov 2016.
- [129] Paul Viola and Michael J. Jones. Robust real-time face detection. *Int. J. Comput. Vision*, 57(2):137–154, May 2004.
- [130] Enrique P. Becerra Vishag Badrinarayana and Sreedhar Madhavaram. Influence of congruity in store-attribute dimensions and self-image on purchase intentions in online stores of multichannel retailers. *Journal of Retailing and Consumer Services*, 21 6:1013–1020, 2014.
- [131] Gregory K. Wallace. The jpeg still picture compression standard. *Communications of the ACM*, pages 30–44, 1991.
- [132] Nannan Wang, Xinbo Gao, Dacheng Tao, and Xuelong Li. Facial feature point detection: A comprehensive survey. *CoRR*, abs/1410.1037, 2014.
- [133] Geoffrey S. Watson. Linear least squares regression. *The Annals of Mathematical Statistics*, 38(6):1679–1699, 1967.
- [134] S. E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh. Convolutional pose machines. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4724–4732, June 2016.
- [135] T. Whelan, M. Kaess, M.F. Fallon, H. Johannsson, J.J. Leonard, and J. McDonald. Kintinuuous: Spatially extended kinectfusion. In *RSS Workshop on RGB-D: Advanced Reasoning with Depth Cameras*, Sydney, Australia, Jul 2012.
- [136] T. Wu, F. Zhou, and Q. Liao. Real-time 3d face reconstruction from one single image by displacement mapping. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 2204–2208, Sept 2017.
- [137] Y. Wu, T. Hassner, K. Kim, G. Medioni, and P. Natarajan. Facial landmark detection with tweaked convolutional neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2017.
- [138] C. N. Xenakidis, A. M. Hadjiantonis, and G. M. Milis. A mobile assistive application for people with cognitive decline. In *2014 International Conference on Interactive Technologies and Games*, pages 28–35, Oct 2014.
- [139] Y. Xi and X. Fangqin. Data modeling and analysis based on the automated storage and retrieval system. In *2012 Second International Conference on Business Computing and Global Informatization*, pages 633–636, Oct 2012.
- [140] Jianming Zhang, Shugao Ma, and Stan Sclaroff. Meem: Robust tracking via multiple experts using entropy minimization. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 188–203, Cham, 2014. Springer International Publishing.

- [141] Jianming Zhang, Liliana Lo Presti, and Stan Sclaroff. Online multi-person tracking by tracker hierarchy. In *Proceedings of the 2012 IEEE Ninth International Conference on Advanced Video and Signal-Based Surveillance*, AVSS '12, pages 379–385, Washington, DC, USA, 2012. IEEE Computer Society.
- [142] H. Zhong, L. Xie, and Q. Xia. Coupon incentive-based demand response: Theory and case study. *IEEE Transactions on Power Systems*, 28(2):1266–1276, May 2013.
- [143] X. Zhu and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2879–2886, June 2012.
- [144] Zoran Zivkovic. Improved adaptive gaussian mixture model for background subtraction. In *Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04) Volume 2 - Volume 02*, ICPR '04, pages 28–31, Washington, DC, USA, 2004. IEEE Computer Society.
- [145] Zoran Zivkovic and Ferdinand van der Heijden. Recursive unsupervised learning of finite mixture models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(5):651–656, 2004.
- [146] Zoran Zivkovic and Ferdinand van der Heijden. Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern Recogn. Lett.*, 27(7):773–780, May 2006.

Appendix A

DATA MODEL FOR RETAIL

This appendix is intended to define the data model that stores all relevant information to optimize retail efficiency expressed in previous chapters. For that aim a Entity/Relation diagram is exposed on figure A.1. In the diagram it can be noted that there is one high-level entity (Retail Establishment) that stores general data of the store. The store is composed by three main entities (Zone, Staff and Point Of Sale). Visitors features are extracted with the visual information from POS and the different zones of the establishment are monitored by hotspots containing different sensing devices. Processing of those signals produces three main outcomes that are represented with weak entities (Heatmap, Satisfaction, Interaction and Queue). Heatmaps can be built upon the trajectory estimation solution proposed in subsection 4.4.2.3 and the satisfaction estimation is highly correlated to the facial landmarks detected by the pipeline proposed in subsection 5.3. Interaction and queues are not fed by any algorithm proposed, but their requirements have been considered for the design of the model and the architectures presented in Chapter 3 . In an analogous manner, alerts generated by the system are also represented by a weak entity. Attributes contained in the entities are not included in the diagram for visibility reasons. In the next subsections they will be specified including Primary Keys (PK)and Foreign Keys (FK).

Figure A.1: Knowledge base modeling to optimize efficiency of a retail establishment.

A.1 Retail Establishment

Fields and keys for Retail Establishment are expressed on table A.1.

Name	Type	Value	Cardinality	Description
Id	Integer	$1 \dots N_{Establish}$	$1 \dots 1$	PK
Default flow	Float[3][N_{Flow}]	$[1 \dots Max_{Flow}]$	$1 \dots N_{Flow}$	Standard visitor flow (Batch)
Default satisfaction	Float	$0 \dots 10.0$	$1 \dots 1$	Standard customer satisfaction (Batch)
Visitors per day	Float	$1 \dots Max_{Vis}$	$1 \dots 1$	Average visitors (Batch)
Recurrency	Float	$1 \dots 100$	$1 \dots 1$	% Of recurrent visitors (Batch)
Sales per day	Float	$1 \dots Max_{Sales}$	$1 \dots 1$	Average value of sales (Batch)
State	Enumerated	{Opening, Closing, Opened, Closed}	$1 \dots 1$	Current establishment state (Real-Time)

Table A.1: Fields included in root table for Retail Establishment.

A.2 Heatmap

Fields and keys for Heatmap are expressed on table A.2.

Name	Type	Value	Cardinality	Description
Establishment	Integer	$1 \dots N_{Establish}$	$1 \dots 1$	PK, FK (Retail Establishment)
Init	Timestamp	YYYY-MM-DD HH:MM:SS	$1 \dots 1$	PK (Real-Time)
End	Timestamp	YYYY-MM-DD HH:MM:SS	$1 \dots 1$	PK (Real-Time)
Storage	String	""	$1 \dots 1$	File location (Real-Time)

Table A.2: Fields included in table Heatmap for Retail Establishment.

A.3 Alert

Fields and keys for Alert are expressed on table A.3.

Name	Type	Value	Cardinality	Description
Establishment	Integer	$1 \dots N_{Establish}$	$1 \dots 1$	PK, FK (Retail Establishment)
Init	Timestamp	YYYY-MM-DD HH:MM:SS	$1 \dots 1$	PK (Real-Time)
End	Timestamp	YYYY-MM-DD HH:MM:SS	$1 \dots 1$	PK (Real-Time)
Responsible	Integer	$1 \dots Max_{Staff}$	$1 \dots 1$	PK, FK (Staff)
Events	String[N_{Even}]	[""]	$1 \dots N_{Even}$	Notifications and remedies (Real-Time)

Table A.3: Fields included in table Alert for Retail Establishment.

A.4 Zone

Fields and keys for Zone are expressed on table A.4.

Name	Type	Value	Cardinality	Description
Id	Integer	$1 \dots Max_{Zones}$	$1 \dots 1$	PK (Batch)
Establishment	Integer	$1 \dots N_{Establ}$	$1 \dots 1$	PK, FK (Retail Establishment)
Ambit	String	""	$1 \dots 1$	Product classification (Batch)
Traffic global	Float	$1 \dots Max_{traff}$	$1 \dots 1$	Global visitors (Batch)
Traffic hour	Float	$1 \dots Max_{traffH}$	$1 \dots 1$	Visitors per hour (Real-Time)
Dimensionality	Float[N_{Dim}]	$[1 \dots Max_{Dim}]$	$1 \dots N_{Dim}$	Coordinates of Polygon in m^3 (Batch)

Table A.4: Fields included in table Zone for Retail Establishment.

A.5 Hotspot

Fields and keys for Hotspot are expressed on table A.5.

Name	Type	Value	Cardinality	Description
Id	Integer	$1 \dots Max_{Hot}$	$1 \dots 1$	PK (Batch)
Establishment	Integer	$1 \dots N_{Establ}$	$1 \dots 1$	PK, FK (Retail Establishment)
Zone	Integer	$1 \dots Max_{Zones}$	$1 \dots 1$	PK, FK (Zone)
Area	Float[N_{Dim}]	$[1 \dots Max_{Dim}]$	$1 \dots N_{Dim}$	Coordinates of Poligon in m^3 (Batch)
Device	String[N_{Dev}]	[$''''$]	$1 \dots N_{Dev}$	Devices per hotspot (Real-Time)
Traffic hour	Float	$1 \dots Max_{Traf}$	$1 \dots 1$	MB per hour (Real-Time)

Table A.5: Fields included in table Hotspots for Retail Establishment.

A.6 Interaction

Fields and keys for Interaction are expressed on table A.6.

Name	Type	Value	Cardinality	Description
Establishment	Integer	$1 \dots N_{Establish}$	$1 \dots 1$	PK, FK (Retail Establishment)
Zone	Integer	$1 \dots Max_{Zones}$	$1 \dots 1$	PK, FK (Zone)
Init	Timestamp	YYYY-MM-DD HH:MM:SS	$1 \dots 1$	PK (Batch)
End	Timestamp	YYYY-MM-DD HH:MM:SS	$1 \dots 1$	PK (Batch)
Location	Float[3]	$[1 \dots Max_{Dim}]$	$1 \dots 1$	PK (Batch)
Sensing Type	Enumerated	{Visual 2D, Visual 3D WSN, Fusion}	$1 \dots 1$	Sensing data source (Batch)
Semantic Type	Enum.[4]	{Look, Touch Take, Neglect}	$0 \dots 4$	Intelligent estimation (Batch)

Table A.6: Fields included in table Interaction for Retail Establishment.

A.7 Staff

Fields and keys for Staff are expressed on table A.7.

Name	Type	Value	Cardinality	Description
Id	Integer	$1 \dots Max_{Staff}$	$1 \dots 1$	PK (Batch)
Establishment	Integer	$1 \dots N_{Establish}$	$1 \dots 1$	PK, FK (Retail Establishment)
Name	String	""	$1 \dots 1$	Person details (Batch)
Departments	String[N_{Dep}]	[""]	$1 \dots N_{Dep}$	Target tasks (Batch)
Saturation	Float	$1 \dots 100.0$	$1 \dots 1$	Work load (Real-Time)
Shift	String[N_{Shift}]	[""]	$1 \dots N_{Shift}$	Worker timing (Batch)

Table A.7: Fields included in table Staff for Retail Establishment.

A.8 POS

Fields and keys for POS are expressed on table A.8.

Name	Type	Value	Cardinality	Description
Id	Integer	$1 \dots Max_{POS}$	$1 \dots 1$	PK (Batch)
Establishment	Integer	$1 \dots N_{Establ}$	$1 \dots 1$	PK, FK (Retail Establishment)
Zone	Integer	$1 \dots Max_{Zone}$	$1 \dots 1$	PK, FK (Zone)
Billing	Float[N_{Bil}]	$[0 \dots Max_{Bil}]$	$1 \dots N_{Bil}$	Quantification of sales per day (Real-Time)
Sale/Neglect	Float	$0 \dots 100.0$	$1 \dots 1$	Success percentage (Batch)
Saturation	Float	$1 \dots 100.0$	$1 \dots 1$	POS load (Real-Time)
State	Enumerated	{Opened, Closed}	$1 \dots 1$	Current POS state (Real-Time)

Table A.8: Fields included in table POS for Retail Establishment.

A.9 Queue

Fields and keys for Queue are expressed on table A.9.

Name	Type	Value	Cardinality	Description
POS	Integer	$1 \dots Max_{POS}$	$1 \dots 1$	PK, FK (POS)
Init	Timestamp	YYYY-MM-DD HH:MM:SS	$1 \dots 1$	PK (Real-Time)
End	Timestamp	YYYY-MM-DD HH:MM:SS	$1 \dots 1$	PK (Real-Time)
Establishment	Integer	$1 \dots N_{Establish}$	$1 \dots 1$	PK,FK (Retail Establishment)
Zone	Integer	$1 \dots Max_{Zone}$	$1 \dots 1$	PK,FK (Zone)
Persons	Integer	$0 \dots Max_{Pers}$	$1 \dots 1$	Current persons in queue (Real-Time)
Average persons	Float	$0 \dots Max_{Pers}$	$1 \dots 1$	Average persons in queue (Batch)
Average time	Integer	$1 \dots Max_{Time}$	$1 \dots 1$	Average waiting time in s (Real-Time)
Neglect time	Integer	$1 \dots m * Max_{Time}$	$1 \dots 1$	Average neglecting time in s (Real-Time)

Table A.9: Fields included in table Queue for Retail Establishment.

A.10 Visitor

Fields and keys for Visitor are expressed on table A.10.

Name	Type	Value	Cardinality	Description
Id	Integer	$1 \dots Max_{Vis}$	$1 \dots 1$	PK (Batch)
Zone	Integer[N_{Zones}]	$[1 \dots Max_{Zone}]$	$1 \dots N_{Zone}$	PK, FK (Zone)
POS	Integer[N_{Pos}]	$[1 \dots Max_{POS}]$	$1 \dots N_{POS}$	PK, FK (POS)
Establishment	Integer	$1 \dots N_{Establ}$	$1 \dots 1$	PK, FK (Retail Establishment)
Bills	Integer[N_{Bills}]	$[1 \dots Max_{Bills}]$	$1 \dots N_{Bills}$	History of sales (Real-Time)
Preferences	String[N_{Pref}]	[$""$]	$1 \dots N_{Pref}$	Ordered favorite ambit (Batch)
Gender	Enumerate	{Male, Female }	$0 \dots 1$	Intelligent characterization (Batch)
Age	Enumerate	{Kid, Young, Adult, Elder }	$0 \dots 1$	Intelligent characterization (Batch)
Ethnicity	Enumerate	{Caucasian, Asiatic, African, Hindu, Indian }	$0 \dots 1$	Intelligent characterization (Batch)

Table A.10: Fields included in table Visitor for Retail Establishment.

A.11 Satisfaction

Fields and keys for Satisfaction are expressed on table A.11.

Name	Type	Value	Cardinality	Description
Visitor	Integer	$1 \dots Max_{Vis}$	$1 \dots 1$	PK, FK (Visitor)
Time	Timestamp	YYYY-MM-DD HH:MM:SS	$1 \dots 1$	PK (Real-Time)
POS	Integer	$[1 \dots Max_{POS}]$	$1 \dots N_{POS}$	FK (POS)
Establishment	Integer	$1 \dots N_{Establish}$	$1 \dots 1$	FK (Retail Establishment)
Degree	Float	$0 \dots 10.0$	$1 \dots 1$	Degree of satisfaction (Batch)
Confidence	Float	$0 \dots 100.0$	$1 \dots 1$	Certainty of estimation (Batch)

Table A.11: Fields included in table Satisfaction for Retail Establishment.

Appendix B

DATA MODEL FOR COGNITIVE HEALTH MONITORING

This appendix is intended to define the data model that stores all relevant information to perform health monitoring of patients with cognitive diseases. Another Entity/Relation diagram has been built to fulfill related information for health professionals, it can be noted on figure B.1. In the diagram it can be noted that root entity is composed by two entities (EHR and Behaviour). The first of them stores general data of the corresponding patient with cognitive diseases, and the second of them performs patient monitoring through signal processing. On one hand clinic data is mostly annotated by health professionals/caregivers (Medication, Scale and Professional) and just weak entities (Symptom and Measurements) are extracted automatically by analyzing sensing devices signal. On the other hand, Motion, Emotion and Routine weak entities add information to Behaviour entity based on sensing data captured with the different devices and architectures presented previously. Trajectories and motion indicators are implicitly estimated by subsection 4.4.2.3, emotion estimation is highly correlated to the facial landmarks detected by the pipeline proposed in subsection 5.3, and some routines are highly correlated to both estimations. The same methodology of previous appendix has been followed to specify fields and keys of every entity in the next sections.

Figure B.1: Knowledge base modeling for cognitive health monitoring.

B.1 Patient

Fields and keys to store general information of one patient are expressed on table B.1.

Name	Type	Value	Cardinality	Description
Id	Integer	$1 \dots N_{Pat}$	$1 \dots 1$	PK (Batch)
Main diagnosis	String	""	$1 \dots 1$	Cognitive disease diagnosed for patient (Batch)
Stage	Integer	$1 \dots 5$	$1 \dots 1$	Current state of disease (Batch)
Diagnosis date	Date	YYYY-MM-DD	$1 \dots 1$	Starting date of disease (Batch)
Ongoing	Boolean	True False	$1 \dots 1$	Disease currently has affections (Real-Time)

Table B.1: Fields included in table patient for health monitoring.

B.2 Alert

Fields and keys for Alert are expressed on table B.2.

Name	Type	Value	Cardinality	Description
Patient	Integer	$1 \dots N_{Pat}$	$1 \dots 1$	PK, FK (Patient)
Init	Timestamp	YYYY-MM-DD HH:MM:SS	$1 \dots 1$	PK (Real-Time)
End	Timestamp	YYYY-MM-DD HH:MM:SS	$1 \dots 1$	PK (Real-Time)
Event	String[N_{Eve}]	[""]	$1 \dots N_{Eve}$	Notifications and remedies (Real-Time)

Table B.2: Fields included in table Alert for health monitoring.

B.3 EHR

Fields and keys for Clinic are expressed on table B.3.

Name	Type	Value	Cardinality	Description
Patient	Integer	$1 \dots N_{Pat}$	$1 \dots 1$	PK, FK (Patient)
Centre	Integer	$1 \dots Max_{Centres}$	$1 \dots 1$	PK (Batch)
Comorbidities	String[N_{Com}]	[""]	$0 \dots N_{Com}$	Additional diseases or disorders (Batch)
Surgeries	String[N_{Surg}]	[""]	$0 \dots N_{Surg}$	Operations performed (Batch)
Implants	Enum.[N_{Imp}]	{Vision, Hearing, Skeleton muscle, Cardiac, Neurologist, Digestive, Urinary, Breathing}	$0 \dots N_{Imp}$	Body insertions performed (Batch)
Addictions	String[N_{Add}]	[""]	$0 \dots N_{Add}$	Dependencies related to the disease (Batch)
Vaccines	String[N_{Vacc}]	[""]	$0 \dots N_{Vacc}$	Antitoxins injected (Batch)
Allergies	String[N_{All}]	[""]	$0 \dots N_{All}$	Aversions treated (Batch)
Care visits	Times.[N_{Vis}]	YYYY-MM-DD HH:MM:SS	$0 \dots N_{Vis}$	Visits to centres (Batch)

Table B.3: Fields included in table EHR.

B.4 Medication

Fields and keys for Medication are expressed on table B.4.

Name	Type	Value	Cardinality	Description
Id	Integer	$1 \dots Max_{Med}$	$1 \dots 1$	PK
Patient	Integer	$1 \dots N_{Pat}$	$1 \dots 1$	PK, FK (Patient)
Name	String	""	$1 \dots 1$	Name of the medication (Batch)
Disease	String[N_{Dis}]	""	$1 \dots N_{Dis}$	Diseases related (Batch)
Times/day	Integer	$1 \dots 24$	$1 \dots 1$	Frequency per day (Batch)
Number of pills	Integer	$1 \dots 10$	$1 \dots 1$	Pills per dose (Batch)
Starting date	Date	YYYY-MM-DD	$1 \dots 1$	Day to start the treatment (Batch)
Ending date	Date	YYYY-MM-DD	$1 \dots 1$	Day to end the treatment (Batch)
Notification	Boolean	True False	$1 \dots 1$	Alert required (Real-Time)

Table B.4: Fields included in table Medication for health monitoring.

B.5 Scale

Fields and keys for Scale are expressed on table B.5.

Name	Type	Value	Cardinality	Description
Id	Integer	$1 \dots MaxScale$	$1 \dots 1$	PK
Patient	Integer	$1 \dots N_{Pat}$	$1 \dots 1$	PK, FK (Patient)
Type	Enumerated	{Hoehn and Yard, Barthel, Lawton and Brody, SPMSQ }	$1 \dots 1$	Name of the scale (Batch)
Date	Date	YYYY-MM-DD	$1 \dots 1$	Day of the evaluation (Batch)
Value	Float	$[Min_{Scale} \dots Max_{Scale}]$	$1 \dots N$	Result of specified scale (Batch)
Details	String[N]	[“”]	$1 \dots N$	Professional annotations (Batch)

Table B.5: Fields included in table Scale for health monitoring.

B.6 Measurement

Fields and keys for Measurement are expressed on table B.6.

Name	Type	Value	Cardinality	Description
Patient	Integer	$1 \dots N_{Pat}$	$1 \dots 1$	PK, FK (Patient)
Time	Timestamp	YYYY-MM-DD HH:MM:SS	$1 \dots 1$	PK (Real-Time)
Type	Enumerated	{Heart rate, Body T°, Blood pressure}	$1 \dots 1$	Measurement target (Real-Time)
Intensity	Float	$Min_{Int} \dots Max_{Int}$	$1 \dots 1$	Value of the measurement (Real-Time)
Details	String[N_{Det}]	[“”]	$1 \dots N_{Det}$	Professional/Caregiver annotations (Real-Time)

Table B.6: Fields included in table Measurement for health monitoring.

B.7 Symptom

Fields and keys for Symptom are expressed on table B.7.

Name	Type	Value	Cardinality	Description
Patient	Integer	$1 \dots N_{Pat}$	$1 \dots 1$	PK, FK (Patient)
Init	Timestamp	YYYY-MM-DD HH:MM:SS	$1 \dots 1$	PK (Batch)
End	Timestamp	YYYY-MM-DD HH:MM:SS	$1 \dots 1$	PK (Batch)
Type	Enumerated	{Festination, Freezing, Loss of balance, Fall down, Dementia}	$1 \dots 1$	PK (Batch)
Intensity	Float	$Min_{Int} \dots Max_{Int}$	$1 \dots 1$	Quantification of the symptom (Batch)
Details	String[N_{Det}]	[$''$]	$1 \dots N_{Det}$	Professional/Caregiver annotations (Batch)

Table B.7: Fields included in table Symptom for health monitoring.

B.8 Professional

Fields and keys for Professional are expressed on table B.8.

Name	Type	Value	Cardinality	Description
Id	Integer	$1 \dots Max_{Prof}$	$1 \dots 1$	PK (Batch)
Patient	Integer	$1 \dots N_{Pat}$	$1 \dots 1$	PK, FK (Patient)
Init	Timestamp	YYYY-MM-DD HH:MM:SS	$1 \dots 1$	Beginning time of professional treatment (Batch)
End	Timestamp	YYYY-MM-DD HH:MM:SS	$0 \dots 1$	Ending time of professional treatment (Batch)
Type	Enumerated	{Neurologist, Psychologist, Psychiatrist, Physiotherapist, Social worker , GP}	$1 \dots 1$	Expertise of the professional (Batch)

Table B.8: Fields included in table Professional for health monitoring.

B.9 Behaviour

Fields and keys for Behaviour are expressed on table B.9.

Name	Type	Value	Cardinality	Description
Id	Integer	$1 \dots Max_{Behav}$	$1 \dots 1$	PK
Patient	Integer	$1 \dots N_{Pat}$	$1 \dots 1$	PK, FK (Patient)
Default trajectories	Float[3][N_{TrajD}]	$[1 \dots Max_{TrajD}]$	$1 \dots N$	Standard patient trajectories (Batch)
Default emotions	String[N_{Emot}]	$[1 \dots Max_{Emot}]$	$1 \dots N$	Standard patient emotions (Batch)
Default routines	String[N_{Rout}]	$[1 \dots Max_{Rout}]$	$1 \dots N$	Standard patient routines (Batch)
Init	Timestamp	YYYY-MM-DD HH:MM:SS	$1 \dots 1$	Beginning time of surveillance (Batch)
End	Timestamp	YYYY-MM-DD HH:MM:SS	$0 \dots 1$	Ending time of surveillance (Batch)
Type	Enumerated	{Visual 2D, Visual 3D WSN, Fusion}	$1 \dots 1$	Sensing surveillance (Batch)

Table B.9: Fields included in table Behaviour for health monitoring.

B.10 Motion

Fields and keys for Motion are expressed on table B.10.

Name	Type	Value	Cardinality	Description
Patient	Integer	$1 \dots N_{Pat}$	$1 \dots 1$	PK, FK (Patient)
Init	Timestamp	YYYY-MM-DD HH:MM:SS	$1 \dots 1$	PK
End	Timestamp	YYYY-MM-DD HH:MM:SS	$1 \dots 1$	PK
Trajectories	Float[3][N_{Traj}]	$[1 \dots Max_{Traj}]$	$0 \dots N$	Estimated trajectories of the patients (Real-Time)
Daily motion	Float	$0 \dots 10.0$	$0 \dots 1$	Quantification of daily movement (Batch)
Sleeping motion	Float	$0 \dots 10.0$	$0 \dots 1$	Quantification of sleeping movement (Batch)
Physiotherapy	Float[N_{Phy}], Enum.[N_{Phy}]	$[0 \dots 10.0]$, [{Postural, Rigidity, Head, Right arm, Left arm, Trunk, Left leg, Right leg }]	$0 \dots N_{Phy}$	Physiotherapy body part movement (Batch)

Table B.10: Fields included in table Motion for health monitoring.

B.11 Emotion

Fields and keys for Emotion are expressed on table B.11.

Name	Type	Value	Cardinality	Description
Patient	Integer	$1 \dots N_{Pat}$	$1 \dots 1$	PK, FK (Patient)
Init	Timestamp	YYYY-MM-DD HH:MM:SS	$1 \dots 1$	PK (Batch)
End	Timestamp	YYYY-MM-DD HH:MM:SS	$1 \dots 1$	PK (Batch)
Appearance	Enum. $[N_{Emot}]$	{Fear, Sadness, Anxiety, Stress, Disgust, Disappointment, Hurt, Jealousy, Hate, Anger, Irritation, Guilt, Shame}	$0 \dots N_{Emot}$	Estimated patient emotion (Batch)
Confidence	Float	$0 \dots 100.0$	$1 \dots 1$	Certainty of estimation (Batch)

Table B.11: Fields included in table Emotion for health monitoring.

B.12 Routine

Fields and keys for Routine are expressed on table B.12.

Name	Type	Value	Cardinality	Description
Patient	Integer	$1 \dots N_{Pat}$	$1 \dots 1$	PK, FK (Patient)
Init	Timestamp	YYYY-MM-DD HH:MM:SS	$1 \dots 1$	PK (Batch)
End	Timestamp	YYYY-MM-DD HH:MM:SS	$1 \dots 1$	PK (Batch)
Continence	Integer[N_{Bath}]	$[0 \dots Max_{Bath}]$	$1 \dots N_{Bath}$	Visits to the bathroom per day (Batch)
Outdoor time	Float[N_{Out}]	$[0 \dots Max_{Out}]$	$1 \dots N_{Out}$	Hours spent outside home per day (Real-Time)
Personal relations	String[N_{Rel}]	$[0 \dots Max_{Rel}]$	$0 \dots N_{Rel}$	Human affinities (Batch)
DIT	String[N_{DIT}]	$[0 \dots Max_{DIT}]$	$0 \dots N_{Rel}$	Paths followed in the app

Table B.12: Fields included in table Routine for health monitoring.