UNIVERSITAT DE BARCELONA

Evolutionary Bags of Space-Time Features for Human Analysis

Víctor Ponce López

Evolutionary Bags of Space-Time Features for Human Analysis

Víctor Ponce López

Dept. of Mathematics and Computer Science
University of Barcelona

This dissertation is submitted for the degree of
Doctor in Mathematics and Computer Science.

Faculty of Mathematics May 2016

DIRECTORS	***Dr. Sergio Escalera Guerrero*** Dept. of Mathematics and Computer Science, Universitat de Barcelona & Computer Vision Center, Universitat Autònoma de Barcelona. ***Dr. Xavier Baró Solé*** Internet Interdisciplinary Institute and Estudis d'Informàtica, Multimèdia i Telecomunicacions, Universitat Oberta de Catalunya.
CO-DIRECTOR	***Dr. Hugo Jair Escalante*** Dept. Computational Sciences, Instituto Nacional de Astrofísica, Óptica y Electrónica.
INTERNATIONAL EVALUATORS & THESIS COMMITTEE	***Dr. Stephane Ayache*** Laboratoire d'Informatique Fondamentale, Aix-Marseille Université. ***Dr. Kamal Nasrollahi*** Dept. of Architecture, Design, and Media Technology, Aalborg Universitet. ***Dr. David Masip Rodó*** Internet Interdisciplinary Institute and Estudis d'Informàtica, Multimèdia i Telecomunicacions, Universitat Oberta de Catalunya. ***Dr. Àgata Lapedriza Garcia*** IN3-EIMT, Universitat Oberta de Catalunya & Computer Vision Center, Universitat Autònoma de Barcelona & CSAIL, MIT.


UNIVERSITAT DE BARCELONA

This document was typeset by the author using LaTeX2e. The research described in this book was carried out at the Universitat Oberta de Catalunya, Universitat de Barcelona and the Computer Vision Center.

ISBN: 978-84-945373-0-1

Printed by Ediciones Gráficas Rey, S.L.

"Learning is the sense of nature."

Als meus pares ...

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other University. This dissertation is the result of my own work and includes nothing which is the outcome of work done in collaboration, except where specifically indicated in the text. This dissertation contains less than 65,000 words including appendices, bibliography, footnotes, tables and equations and has less than 150 figures.

Víctor Ponce López
May 2016

Acknowledgements

I would like to acknowledge the guidance, confidence, dedication, and support of my advisors Sergio and Xavier during this thesis, which have made it possible today. I am very happy to achieve the competences obtained during these years of hard working, through personal and professional experiences, some of them encouraged by my advisors in those moments of hesitation. There have been really enjoyable, adventurous, dangerous, and emotive moments. At the same time, I would like to thank Hugo for his work as co-advisor at the final stages of this thesis. His implication and support were given at tough times where necessary, both helping to keep developing this thesis in the right direction and checking some important details, a fact that always marks the difference. Thus, they contributed to my love for science.

I would like to thank all my colleagues from different universities that appeared in the different stages of this travel:

- At the final steps of my B.Sc. and M.Sc. degrees and the early stages of my PhD, I want to highlight special thanks to Miguel Àngel, Miguel, Toni, Albert, Xavi, and Adriana for the really good moments we shared: conversations, travels, parties, among other experiences. You changed my life, and you contributed to my motivation and dream of working together for common goals in a very good environment, which is enjoyable and very important. I would like to thank my other colleagues from the master in AI for the good moments, and the people from the dept. MAiA (UB) and the CVC (UAB) for their collaboration, feedbacks and events we shared. There are, in fact, too many people that I would like to mention here. All and each one of you have, somehow, contributed.

- I am very happy both to be present and to have been part of the consolidation of the groups HuPBA and SUNAI, and to welcome new members with new energies, novel ideas and proposals from whom I learned a lot in several events and seminars: Marc, Pablo, Ciprian, Meysam, Dani, Cristina, and Gerard. Also, I want to thank David, Àgata, and other colleagues from the IN3 and 22@, who are very nice people. You made possible and very welcoming, together with Xevi, me joining the UOC, my work

place, seminars given by well-known visitors, and to help managing administrative stuff for the assistance of some important conferences.

- Finally, I would like to thank Cécile and other nice people from the Qarma group at the Laboratoire d'Informatique Fondamentale and Centre des Mathématiques et de l'Informatique, Aix-Marseille Université, where I did my PhD stay. I met there nice people whom I am very thankful, like Antoine, who helped both my integration into the group, and for taking part of some different and enjoyable events in the city. And last, but not least, I would like to thank Isabelle, a very nice person with whom I worked and learned a lot during my stay, and we had a lot of brilliant brainstormings. All of you became very friendly, and it was very nice to learn from your knowledge and the way to work in my neighboring country. I am very thankful for all the knowledge I have learned over those few but intensive months.

Per altra banda, he de reconèixer i agraïr el recolzament i esforç d'altres persones properes i no tant properes.

En primer lloc, agraïr als meus bons amics, en particular en Jordi, Ricard i Oriol, la família que he triat i que són allà faci fred o calor per parlar, sortir, compartir moments bons i no tant bons, o pel fet tant simple però a la vegada tant important com és el de tenir moments de desconnexió en general. També a tots els companys, companyes, amics i amigues de l'escola Josep Tous als qui tant he arribat a apreciar i que segueixo tenint un apreci tant especial.

Tanmateix sento un agraïment especial a la meva escola de música Diesi, i en particular a la Gemma, qui em va ensenyar a aprendre, estimar, i transmetre la música i el seu llenguatge com a un dels arts i talents més grans que existeix, i que després de tants anys segueixo practicant per mantenir els sentiments i el benestar que la música produeix. De la mateixa manera, agraeixo als meus mestres de Hapkido i arts marcials, en particular l'André i el seu alumne Gabriel, la seva instrucció desinteressada que m'ha marcat de per vida tant en el meu ésser físic-mental com en el meu estil de vida. Així mateix també agraeixo als meus companys Albert, Carles, Ricard, Enrique, i entre molts d'altres que hi són o hi van ser, per compartir tot o part d'aquest camí, amb les bones energies i vibracions que es transmeten tant en el dojang com fora d'ell.

Agraeixo també les bones amistats fetes i que perduren de Gualba, el petit però alhora gran poble on cada cop que hi vaig tinc tants bons moments fora de la densa ciutat, com la Festa Major amb persones com l'Arnau, Jordi, Marc, o Òscar, entre molts d'altres. Sense dubte aquests moments allà no haguessin estat possible sense els meus tiets als qui n'estic molt agraït i aprecio molt.

En general, he de fer un reconeixement especial a la meva família Ponce López. Vosaltres i els meus cosins amb els qui he compartit i comparteixo bons moments, especialment cap al Nadal, heu estat una inspiració i motivació: he après molt de vosaltres des de ben petit per arribar a fer el que faig i ser qui sóc ara. I per les noves (bé, algunes ja no tant noves) generacions de cosins que apunten ben alt, com en Jaume, la Marta, la Laura, en Theo, i les que segueixen i hi seguiran venint al llarg del temps.

I per acabar, però no menys important, agrair a la meva família directa amb la qui he crescut. No hi ha prou paraules per descriure el que em transmeteu i el que he après de tots i cadascun de vosaltres. Agraeixo i admiro, entre moltes d'altres virtuds importants, la fermesa, bondat i humor del meu Pare; la destresa, humiltat i somriure de la meva Mare; i l'experiència, voluntat i solidaritat del meu Germà. Finalment, agrair a la Sònia per ser la meva companya d'aquest intens viatge des de gairebé l'inici del camí. Pel seu desinteressat i constant suport, el seu infinit amor, i per ser una gran bona persona, així com també ho són la seva Mare i el seu Avi. T'admiro i t'estimo.

Tots vosaltres sou uns grans exemples del meu model. Gràcies.

Abstract

The representation (or feature) learning has been an emerging concept in the last years, since it collects a set of techniques that are present in any theoretical or practical methodology referring to artificial intelligence. In computer vision, a very common representation has adopted the form of the well-known Bag of Visual Words. This representation appears implicitly in most approaches where images are described, and is also present in a huge number of areas and domains: image content retrieval, pedestrian detection, human-computer interaction, surveillance, e-health, and social computing, amongst others.

The early stages of this dissertation provide an approach for learning visual representations inside evolutionary algorithms, which consists of evolving weighting schemes to improve the BoVW representations for the task of recognizing categories of videos and images. Thus, we demonstrate the applicability of the most common weighting schemes, which are often used in text mining but are less frequently found in computer vision tasks. Beyond learning these visual representations, we provide an approach based on fusion strategies for learning spatiotemporal representations, from multimodal data obtained by depth sensors. Besides, we specially aim at the evolutionary and dynamic modelling, where the temporal factor is present in the nature of the data, such as video sequences of gestures and actions. Indeed, we explore the effects of probabilistic modelling for those approaches based on dynamic programming, so as to handle the temporal deformation and variance amongst video sequences of different categories. Finally, we integrate dynamic programming and generative models into an evolutionary computation framework, with the aim of learning Bags of SubGestures (BoSG) representations and hence to improve the generalization capability of standard gesture recognition approaches.

The results obtained in the experimentation demonstrate, first, that evolutionary algorithms are useful for improving the representation of BoVW approaches in several datasets for recognizing categories in still images and video sequences. On the other hand, our experimentation reveals that both, the use of dynamic programming and generative models to align video sequences, and the representations obtained from applying fusion strategies in multimodal data, entail an enhancement on the performance when recognizing some gesture categories. Furthermore, the combination of evolutionary algorithms with models based on

dynamic programming and generative approaches results, when aiming at the classification of video categories on large video datasets, in a considerable improvement over standard gesture and action recognition approaches.

Finally, we demonstrate the applications of these representations in several domains for human analysis: classification of images where humans may be present, action and gesture recognition for general applications, and in particular for conversational settings within the field of restorative justice.

Resum

L'aprenentatge de la representació (o de característiques) ha estat un concepte emergent en els darrers anys, ja que recopila un conjunt de tècniques que són presents en qualsevol metodologia teòrica o pràctica referent a la intel·ligència artificial. En la visió per computador, una representació molt comuna ha adoptat la forma de la ben coneguda Bossa de Paraules Visuals (BdPV). Aquesta representació apareix implícitament en la majoria d'aproximacions per descriure imatges, i és també present en un enorme nombre d'àrees i dominis: recuperació de contingut en imatges, detecció de vianants, interacció humà-ordinador, vigilància, e-salut, i la computació social, entre d'altres.

Les fases inicials d'aquesta dissertació proporcionen una aproximació per aprendre representacions visuals dins d'algorismes evolutius, que consisteix en evolucionar esquemes de pesat per millorar les representacions BdPV en la tasca de reconèixer les categories de vídeos i imatges. Per tant, demostrem l'aplicabilitat dels esquemes de pesat més comuns, que s'usen sovint en la mineria de textos però es troben amb menys freqüència en tasques de visió per computador. Més enllà d'aprendre representacions visuals, proporcionem una aproximació basada en estratègies de fusió per a l'aprenentatge de representacions espai-temporals, a partir de dades multi-modals obtingudes per sensors de profunditat. A més, el nostre objectiu és especialment el modelatge evolutiu i dinàmic, on el factor temporal és present en la naturalesa de les dades, com les seqüències de gestos i accions. De fet, explorem els efectes del modelatge probabilístic per aquelles aproximacions basades en programació dinàmica per a gestionar la deformació temporal i variància entre seqüències de vídeo de categories diferents. Finalment, integrem la programació dinàmica i els models generatius en un marc de computació evolutiva, amb l'objectiu d'aprendre representacions en Bosses de SubGestos i, per tant, millorar la capacitat de generalització de les aproximacions estàndards pel reconeixement de gestos.

Els resultats obtinguts en l'experimentació demostra, en primer lloc, que els algorismes evolutius són útils per millorar la representació d'aproximacions BdPV en diverses bases de dades pel reconeixement de categories en imatges fixes i seqüències de vídeo. Per altra banda, la nostra experimentació revela que, tant l'ús de la programació dinàmica i els models generatius per alinear seqüències de vídeos, com les representacions obtingudes d'aplicar

estratègies de fusió en dades multi-modals, comporten una millora en el rendiment a l'hora de reconèixer algunes categories de gestos. A més a més, la combinació d'algorismes evolutius amb models basats en programació dinàmica i aproximacions generatives resulten, a l'hora de classificar categories de vídeos de bases de dades grans, en una millora considerable sobre les aproximacions estàndards de reconeixement de gestos i accions.

Finalment, demostrem les aplicacions d'aquestes representacions en varis dominis per a l'anàlisi humà: classificació d'imatges on els humans poden ser-hi presents, el reconeixement d'accions i gestos per aplicacions en general, i en particular per entorns conversacionals dins del camp de la justícia restaurativa.

Resumen

El aprendizaje de la representación (o de características) ha sido un concepto emergente en los últimos años, ya que recopila un conjunto de técnicas que están presentes en cualquier metodología teórica o práctica referente a la inteligencia artificial. En la visión por computador, una representación muy comuna ha adoptado la forma de la bien conocida Bolsa de Palabras Visuales (BdPV). Esta representación aparece implícitamente en la mayoría de aproximaciones para describir imágenes, y está también presente en un enorme número de áreas y dominios: recuperación de contenido en imágenes, detección de peatones, interacción humano-ordenador, vigilancia, e-salud, y la computación social, entre otras.

Las fases iniciales de esta disertación proporcionan una aproximación para aprender representaciones visuales dentro de algoritmos evolutivos, que consisten en evolucionar esquemas de pesado para mejorar las representaciones BdPV en la tarea de reconocer las categorías de vídeos y imágenes. Por lo tanto, demostramos la aplicabilidad de los esquemas de pesado más comunes, que se utilizan a menudo en la minería de textos pero se encuentran con menos frecuencia en tareas de visión por computador. Más allá de aprender representaciones visuales, proporcionamos una aproximación basada en estrategias de fusión para el aprendizaje de representaciones espacio-temporales, a partir de datos multimodales obtenidos por sensores de profundidad. También, nuestro objetivo es especialmente el modelado evolutivo y dinámico, donde el factor temporal está presente en la naturaleza de los datos, como las secuencias de gestos y acciones. De hecho, exploramos los efectos del modelado probabilístico para aquellas aproximaciones basadas en programación dinámica para gestionar la deformación temporal y varianza entre secuencias de vídeo de categorías diferentes. Finalmente, integramos la programación dinámica y los modelos generativos en un marco de computación evolutiva, con el objetivo de aprender representaciones en Bolsas de SubGestos, y por lo tanto mejorar la capacidad de generalización de las aproximaciones estándares para el reconocimiento de gestos.

Los resultados obtenidos en la experimentación demuestra, en primer lugar, que los algoritmos evolutivos son útiles para mejorar la representación de aproximaciones BdPV en diversas bases de datos para el reconocimiento de categorías en imágenes fijas y secuencias de vídeo. Por otra parte, nuestra experimentación revela que, tanto el uso de

la programación dinámica y los modelos generativos para alinear secuencias de vídeos, como las representaciones obtenidas de aplicar estrategias de fusión en datos multimodales, conllevan una mejora en el rendimiento a la hora de reconocer algunas categorías de gestos. Además, la combinación de algoritmos evolutivos con modelos basados en programación dinámica y aproximaciones generativas resultan, a la hora de clasificar categorías de vídeos de bases de datos grandes, en una mejora considerable sobre las aproximaciones estándares de reconocimiento de gestos y acciones.

Finalmente, demostramos las aplicaciones de estas representaciones en varios dominios para el análisis humano: clasificación de imágenes donde los humanos pueden estar presentes, el reconocimiento de acciones y gestos para aplicaciones en general, y en particular para entornos conversacionales dentro del campo de la justicia restaurativa.

Contents

List of figures

List of tables

Nomenclature

Roman Symbols

$\mathbf{d}$	Vector representation of a document.
A	Points of the DFT.
A_b	Average agitation over all frames by the optical flow from the upper body.
$\dot{a}$	Point of the DFT signal.
A_h	Average agitation over all frames by both hands.
b	Bins.
$\dot{b}$	Point of the DFT signal, *s.t.* $\dot{a} \neq \dot{b}$.
C	Set of classes.
c_g	Class of a gesture.
$D(\cdot)$	Function of soft-distance measure of the probability.
$d(\cdot)$	Function of the Euclidean Distance.
d^F	Complementary of the histogram intersection as a Distance.
D^T	Discretized training set.
D^V	Discretized validation set.
d^V	Discretized validation sequence in D^V.
F	Set of frames from any multimodal channel.
f	Feature.

F_{const} Constant value for scale.

$\hat{f}_{lin}$ Koenig scale.

$\hat{f}_{mel}$ Approximation of scaling frequencies used in MFCC.

$\widetilde{F}_t$ Set of feature vectors at time t.

G Number of Gaussian components.

H Extension of the Harris detector, known as STIP detector.

h Hits.

S_D Histogram descriptor for the Depth channel.

h_{min} Minimum hits.

S_{RGB} Histogram descriptor for the RGB channel.

I Individual of the population.

i^* Set of $i-$th subgesture arguments/identifiers in $\vec{km}$.

j^* Set of $i-$th subgesture arguments/identifiers in $\vec{kt}$.

K Relative importance constant factor.

k Number of clusters or current component.

k_0 Number of initial segments (the minimum number of clusters).

k_{const} Constant value for frequency.

k_f Number of pair-wise generated segments.

$\vec{kt}$ Column vectors in KT.

KM Cost matrix $\mathbf{K}$ from the training sequences.

$\vec{km}$ Column vectors in KM.

KT Cost matrix $\mathbf{K}$ from the validation/test sequences.

l Length of the population.

L_g Length, in number of frames, of sequence S_g.

M	Cost Matrix.
m	Length of a time series or number of elements in a set, *s.t.* $n \neq m$.
m_c	Model sequence of the class c.
$\vec{m}$	Vector that represents the input sequence in terms of subgesture arguments.
N	Number of training samples in the training set.
n	Length of a time series or number of elements in a set.
n_{max}	Number of maximum frames of a segment.
n_{min}	Number of minimum frames of a segment.
O	An input time series, *s.t.* $O \neq Q$.
o	Frame element of the time series O.
P	Population of individuals in the Genetic Algorithm.
p	Position in the warping path Ω.
$P(\cdot)$	Function of Probability.
$p(k)$	Probability of increasing segments/decrease clusters given the cluster k.
p_s	Probability value to add/delete segments.
P_{xy}	Plane orthogonal to the viewing z-axis in the Cartesian coordinate system.
Q	An input time series.
q	Frame element of the time series Q.
R	Set of Resized training sequences.
r	Number of stable regions found on the cloud.
r_c	Mean of resized sequences for the class c.
S	Set of sequences (or subsequences).
s	Subgesture sequence.
$\bar{S}$	Median length sequence.

S_D Set of interest points from the Depth channel volume.

S_g Gesture sequence.

S_i The i^{th} sequence element in S.

S_i^g The i^{th} element representing a feature vector in the gesture sequence S_g.

$\tilde{S}_i$ Warped sequences in $\tilde{S}$.

$s_i^{\dot{x}}$ Feature vector of the $i-$th subgesture sequence at position $\dot{x}$.

$s_j^{\dot{y}}$ Feature vector of the $j-$th subgesture sequence at position $\dot{y}$.

S_{RGB} Set of interest points from the RGB channel volume.

$\tilde{S}$ Set of warped sequences.

T Number of thresholds.

t Content term.

t Frame or time instance within a sequence.

$Tr(\cdot)$ Function of trace computation.

u Dimension u of the volume.

υ Dimension υ of the volume.

V Vocabulary from a set of words.

v Element of a warping path or video session.

W Matrix of weight terminals.

w Terminal weight element in matrix W.

w_{ij} Cost between the $i-$th and the $j-$th subgestures.

x Feature vector, which may represent either a document, an image, or a video sequence.

$\dot{x}$ x-axis of the 2-D or pixel coordinate.

x' Element in $\mathcal{N}(x)$.

x_s Sequence segment in X_{seg}^T.

X^T Training dataset.

x^T Training sequence in X^T.

X^T_{seg} Set of segments in training.

X^V Validation dataset.

x^V Validation sequence in X^V.

$\tilde{x}(\dot{a})$ DFT input signal.

$\tilde{X}(\dot{b})$ Filter bank with several equal height triangular filters.

$\dot{y}$ y-axis of the 2-D or pixel coordinate.

D Normalized dissimilarity matrix among subgestures.

K Cost matrix of k updated cost vectors $\mho$.

M General representation of the set of class Models.

$\mathbf{W}^\intercal$ Transposed of matrix **W**.

W Cost similarity matrix among subgestures.

Greek Symbols

α Mixing value of the Gaussian Mixture Model.

$\alpha_{1..3}$ Angles of view for the different sensor cameras.

$\bar{\sigma}_{\iota_i}$ Average optical flow of the upper body for a given frame.

χ^2 Statistical Chi-Square.

Δ_β Offset angle among the head poses.

Δ_h Position offset among hands.

$(\delta_x, \delta_y, \kappa_x, \kappa_y)$ Intrinsics of the depth camera.

Δ_p Position offset among the left hand.

Δ_q Position offset among the right hand.

Δ_Θ Offset pixels among the mass centers.

Δ_{Ξ} Size difference factor among the region areas.

η Matrix of first order spatial and temporal derivatives.

γ Maximum cost value.

ι A frame instance from the set F.

Λ Representation of a Gaussian Mixture Model.

λ Eigenvalue.

μ Mean parameter of the Gaussian Mixture Model.

Ω Warping path.

ω Set of learning model parameters.

ω^* Global set of learned model parameters.

ϕ Angle between the normal point cloud ρ and the z-axis.

ψ Angle between the normal point cloud ρ and the y-axis.

Ψ_{β} Threshold for the offset angle among the head poses.

Ψ_{Θ} Threshold for the offset pixels among the mass centers.

Ψ_{Ξ} Threshold for the size difference factor among the region areas.

Ψ_{ζ} Threshold for the confidence.

ρ Point cloud.

Σ Co-variance parameter of the Gaussian Mixture Model.

τ Length of the warping path.

Θ Set of thresholds for each class.

θ A Threshold.

θ^{c_g} Set of thresholds learned for a gesture class.

ε Accumulated value of detection errors.

ϖ Dimension ϖ of the volume.

ς Weighting factor.

$\mathbf{v}$ Video session.

ζ Confidence value.

Superscripts

i sub/super script index for rows.

j sub/super script index for columns.

Other Symbols

$\flat$ Threshold.

ι Number of iterations for the aligned temporal clustering.

$\mho$ Minimum cost path, computed and updated from backtracking.

$\wp$ Person.

$\wp_{\mathbf{v}}$ Person appearing in a video session.

$\mathscr{F}$ Function set.

$\mathscr{N}(x)$ Set of three upper-left neighbor locations of x in M.

$\mathscr{P}$ Number of points in the cloud.

Acronyms / Abbreviations

BNS Bi-Normal Separation.

d_{hist} Histograms distance.

E.g./e.g. Abbreviation for the Latin phrase *exempli gratia*, meaning 'for example'.

FGT Global Term-Frequency.

FN False Negatives.

FP False Positives.

hist Abbreviation of Histogram.

IDF Inverse-Document-Frequency.

I.e./i.e. Abbreviation for the Latin phrase *id est*, meaning 'this is' (make the meaning of something clearer or show its true meaning).

IG Information Gain.

RF Relevance Frequency.

s.t. Abbreviation of '*such that*'.

TDR Term-Document Relevance.

TF Term-Frequency.

TN True Negatives.

TP True Positives.

TR Term-Relevance.

w.r.t. Acronym for making the comparison 'with respect to'.

The remainder set of acronyms / abbreviations are defined along the text.

Chapter 1

Introduction

This chapter presents the motivation of this thesis referred to the physiological perspective of the human brain that induces us to use Bag of Visual Words representations for learning computer vision problems, and mainly those ones where humans are present. Thus, a brief description of the different goals and contributions is proposed, making reference to the subsequent chapters. In the last section one can find the outline description for the different chapters so as to contextualize the reader in each part.

1.1 Motivation

Humans are experts on recognizing objects and events in the world. The brain is able to perform this type of complex cognitive tasks, such as efficiently correlating the information perceived from our senses with the information stored in memory, and hence to select the resulting output object from the perceived information or input stimulus. Humans achieve quite good performance even when objects are subject to situations that make the recognition much harder (*e.g.* rotation, translation, spatiotemporal changes, or occlusions among other objects). Similarly, this ability of humans is maintained even when not perceiving the whole objects themselves, but only parts of the objects that are representative enough for their recognition. Thus, the composition of these parts is what form the whole objects. Similarly to other subfields within Artificial Intelligence (AI), such as natural language processing, the community of computer vision and machine learning calls Bag of -Visual- Words (Bo**V**W) to these representative parts of the objects, as shown in Figure 1.1.

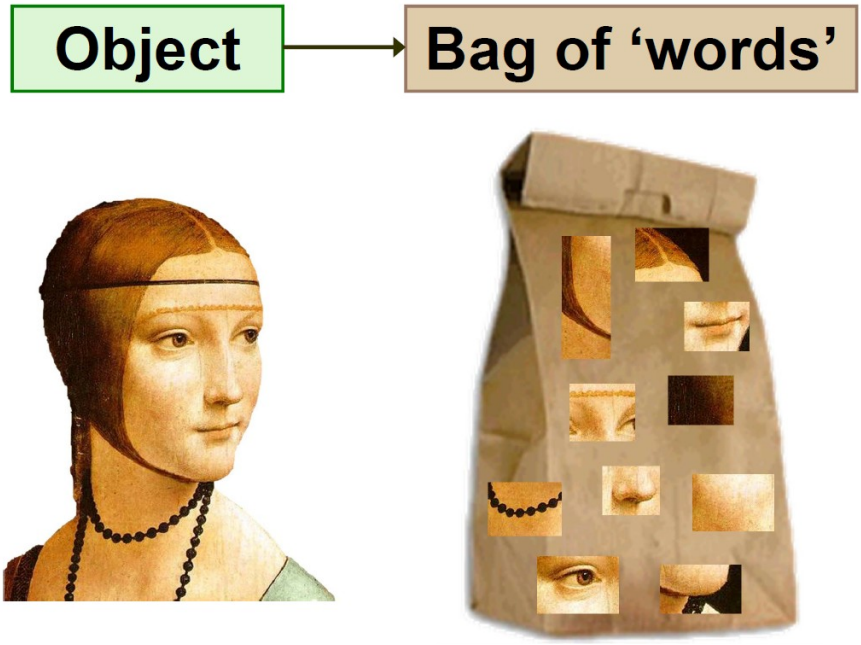


Fig. 1.1 Example of Bag of Visual Words[1].

Since a large number of approaches based on the BoVW paradigm have been applied successfully in many problems where data consist of still images, nowadays, many scientists claim its integration into time series data, where objects vary their characteristics in time. The analysis of such temporal data sequences is a key problem present in a large number of domains. The main challenge to address is the construction of computational models able to learn, in an unsupervised or semi-supervised fashion, temporal primitives that generalize

as better as possible. Therefore, the goal is to enhance the performance on different tasks, such as the detection or classification of actions, events, or gestures. Following the BoVW paradigm, in the case of gesture recognition, temporal primitives can be understood as Bag of SubGestures (BoSG). However, in this case the problem becomes much more complex and computationally demanding, since it requires to find a huge number of subsets in the data, while considering both spatial and temporal domains.

Recently, the emerging models inspired on deep learning include the BoVW approach as part of their networks' architecture. These novel approaches involve the development of robust systems featured by three main advantages: to model the invariance present in data, to transfer unsupervised learning, and to learn hierarchical structures. Due to the extensive use of deep learning and its competitiveness in a large number of domains, novel deep architectures include the use of algorithms based on evolutionary computation in order to improve the performance in machine learning tasks. Evolutionary learning techniques have grown in the last years due to their flexibility and proven effectiveness in computer vision tasks, while keeping the compatibility with the main mentioned advantages of deep learning based approaches. Nowadays, many applications launched on the market tend to keep the above theoretical foundations within their core development, under the new demands of the current society.

Fig. 1.2 Examples of human communication[2].

Moreover, such applications tend to handle with multimedia content or multimodal data for building BoW representations, as those that analyze the human behavior in communication of language. Figure 1.2 shows examples of some conversational contexts, as well as the application of several computer vision techniques that reveal visible behavioral components computed from BoVW-based descriptors. Such behavioral cues appear implicitly in these processes and are of particular interest to pay attention on. Indeed, human language is

[1]Image from http://vision.stanford.edu/

misconstrued if it is not seen as a unity of two main modalities: speech and gesture. Thus, human language is not the same as human speech. A fundamental divergence, proffered as an insight into the human mindset for language in general, is that gestures are components of speech, not accompaniments but actually integral parts of it. Language could not have come into existence without gestures, and both have been developed along evolution together. However, speech and gesture obviously differ in how they distribute information in time. Under the fact that a gesture is not necessarily composed out of parts, but the parts are composed out of it, the changes on the gesture continuum is determined in terms of the temporal alignments of gestures with speech. In this sense, there exist several evolution models for language, so that genetics plays an important role in the the way that different humans gesticulate. Therefore, to construct robust machines able to learn, understand, and perhaps, to imitate the human behavior in communication in a natural way, it is necessary to study the origins of human language jointly from such interdisciplinary fields as deep learning, evolutionary computation, neuroscience, linguistics, and psychology.

1.2 Goals of this thesis

In this thesis, we explore both the theoretical foundations of learning spatiotemporal representations that evolve and their applications in real domains[3]. We begin from the classical and very static approaches based on BoVW, which take into account the information contained in still images, and including different modalities by means of fusion strategies. Then, we extend them to those more dynamic approaches that include a temporal dimension for learning representations of image sequences over time. We use evolutionary computation in both sides, as part of the global optimization methods for evolving such representations, so that we use them in combination with other approaches. Finally, we differentiate a mid level of abstraction to define a feature space where several applications take place, and expose our studies in real scenarios.

1.2.1 Learning Visual Representations

As done in document analysis, we explore several representations based on the BoW approach. In computer vision, however, the main difference *w.r.t.* document analysis is that the words

[2]Images from http://victorponce.org

[3]The different approaches presented in Chapters 3, 4 and 5 have been are already published in international journals or conference proceedings. However, additional explanations and notations may change *w.r.t.* the original manuscripts so as to keep both the consistency and the storyline of this thesis amongst those different chapters.

are visual parts whose images are composed out of. For computers, as visual we refer to matrices of pixel values coming from the RGB signal channels, where operations are performed from this low level of abstraction for further subsequent analyses. Then, the main goals are to describe features that represent this information by means of computer vision techniques, and to learn from them through machine learning for recognition tasks. In particular, the recognition in our cases consist of predicting the class label (or category), either from still images that are independently labeled, or from video sequences whose labels describe the type of gestures, events, or actions performed by the people in each video sequence.

In addition, we consider evolutionary computation to enhance the BoVW-based representations over time. It mainly consist of computing representations in an iteratively way where, at each generation, richer representations are obtained by means of an optimization procedure with the goal of improving the final recognition task on image classification.

1.2.2 Learning SpatioTemporal Representations

Beyond the visual BoW (or BoVW), we consider fusion strategies for learning novel representations based on multimodal data that come from additional channels, such as those from a depth sensor. The goal is to generate richer descriptions, so called Bag of Visual and Depth Words (BoVDW), by adding useful, non-redundant, and discriminative information. These fusion approaches consist of grouping key information at different levels, usually by means of clustering methods, so as to generate sets of vocabulary descriptions that appear in each class category, either before or after a learning task depending on the particular fusion strategy. Moreover, the information that we add to the description may depend on time, especially when having temporal sequences (*e.g.* videos). In such cases, we consider the addition of information referred to motion patterns as part of the feature descriptions. Some of these descriptions can be represented as spatiotemporal volumes or pyramids, and may contain specific references at a certain image (*i.e.* key frames).

On the other hand, a large number of domains that include temporal information require a type of modelling for describing how the features evolve over time. Moreover, such domains may require an exhaustive analysis of time series which are naturally present in the data. We face these problems, mainly, by means of dynamic programming, temporal clustering, mixture models, and generative models. Many other approaches coming from the physics and mathematics, however, would be also very appropriate here, as those based on dynamical systems. The key idea of these methods is that information is added over time (*e.g.* in an incremental form), so that they allow to model the invariance of unsupervised data for the

learned representations, as well as to build architectures from complex structures that provide a clearer overview of the problem at hand.

1.2.3 Evolving Dynamic Representations

Similarly to BoW, it is common to select data subsets from such dynamic representations, as small temporal parts (*e.g.* segments) that will be used for training. Since the segment search on a dynamic feature space, however, could become an NP-hard problem, we use evolutionary computation (*e.g.* genetic algorithms) that acts as an alternative optimization method to those based on the backpropagation or gradient descent algorithms, which besides present simple and effective solutions. Furthermore, it can act as a reinforcement learning approach and may be applied in conjunction with other deep learning approaches.

1.2.4 Applications for Human Analysis

The overview of approaches presented above are widely used in artificial intelligence applications for the analysis of human behavior and language, both from the natural language processing and from the computer vision and machine learning communities. However, these approaches must be adapted to the specific problem so as to cover the requirements and demands of the experts of the application domain. Usually, the aforementioned methodologies use to model the problems at a different abstraction levels in order to make the problem more suitable. In our cases and domain settings for human analysis, we use different levels of descriptions computed from multiple data modalities for categorizing still images, image sequences, and their applications such as human communication. Our particular example of application belongs to the field of Restorative Justice, where we use mid level abstraction of features from the lowest-level multimodal features (based on BoVW representations) to a higher-level description of behavioral indicators that appear in conversations. Such behavioral indicators appear frequently in language in the form of social signals, and can be easily identified by computers and humans. However, for humans it consist of subconscious processes that takes part implicitly within the brain. Our goal is to use computers for keeping some of these processes out for reasoning, so as to make them more explicit and visible by means of several approximations and responses that have been used along the literature for similar purposes. The idea is to help experts on the domain to give ideas or feedback to improve their efficiency and expertise in order to achieve their final goals.

1.3 Contributions

1.3.1 Learning Visual Representations

- We explore the use of alternative weighting schemes for boosting the performance of methods based on Bag of Visual Words. More importantly, we explore whether it is possible to automatically learn and determine effective weighting schemes from scratch. Then, we analyze the suitability of using well-known supervised and unsupervised weighting schemes for landmark tasks in computer vision: image categorization and gesture recognition. For this purpose, we propose an evolutionary algorithm capable of learning weighting schemes for computer vision problems. We report experimental results of an extensive experimental study in several computer vision problems, showing the effectiveness of the proposed evolutionary algorithm in standard image and video datasets.

1.3.2 Learning SpatioTemporal Representations

- We present a methodology to address the problem of human gesture segmentation and recognition in video and depth image sequences. A Bag-of-Visual-and-Depth-Words model is introduced as an extension of the BoVW model. State-of-the-art RGB and depth features, including a newly proposed depth descriptor, are analysed and combined in a late fusion form. The method is integrated in a human gesture recognition pipeline, together with a novel probability-based Dynamic Time Warping (DTW) algorithm, which is used to perform prior segmentation of idle gestures. The proposed DTW variant uses samples of the same gesture category to build a Gaussian Mixture Model driven probabilistic model of the gesture class. Results of the whole human gesture recognition pipeline in public datasets show better performance in comparison to both standard BoVW and DTW approaches.

1.3.3 Evolving Dynamic Representations

- We introduce a framework for gesture and action recognition based on the evolution of temporal gesture primitives, or Bag of Sub-Gestures (BoSG). This is inspired on the principle of producing genetic variations within a population of gesture subsequences. The goal is to obtain a set of gesture units that enhance the generalization capability of standard gesture recognition approaches. In our context, gesture primitives are evolved over time using dynamic programming and generative models. This allows

to learn richer representations along generations for recognizing complex actions. In few generations, the proposed subgesture-based representation of actions and gestures outperforms the state of the art results on several and action datasets.

1.3.4 Applications for Human Analysis

- We expose several achievements obtained along the literature when characterizing some behaviors from visual data on different real applications, and discuss about the important issues to be considered from an interdisciplinary perspective: the low level vocabulary definition from Bags of Gesture Units (such as the aforementioned BoSG), the high-level modelling from BoW representations and the subsequent inference of human behavioral cues, and the traits discovery. The discussion is engaged under the purpose of developing Software tools able to obtain a set of subjects' features from automatic audiovisual analysis. This higher level of feature extraction obtained from language is of particular interest for the analysis of psychological factors that a subject presents, and it has been widely studied along the literature in the fields of social computation and social signal processing. This type of analysis is motivated both to improve the quality of communication in several domains (presentations, job interviews...) or to provide a feedback to experts of several domains in order to analyze their self-performance. In particular, we present a non-invasive ambient intelligence framework for the semi-automatic analysis of non-verbal communication, applied to conversational settings within the Restorative Justice field. We propose the use of computer vision and social signal processing technologies in real scenarios of Victim–Offender Mediations (VOM), applying feature extraction methods so as to obtain, from multi-modal audio-RGB-depth data, representations based either on BoW or other techniques. We subsequently compute a set of behavioral indicators that define communicative cues from the fields of psychology and observational methodology. We test our methodology on a dataset captured in real VOM sessions. We define the ground truth based on expert opinions when annotating the observed social responses. Using different state of the art binary classification approaches, our system achieves promising recognition performances on predicting social responses in such domains.

1.4 Thesis outline

The next five chapters describe the main content of this book. The adjacent second chapter packs the whole theoretical background of this thesis, which is divided accordingly on

the subsequent chapters. This will follow the type of this chapter, as well as the natural multidisciplinary of this thesis, *i.e.* through static and dynamic approaches, and finally to applications. Thus, the chapters three, four, and five, follow the next chapters of this thesis as: learning visual and spatiotemporal representations, learning dynamic representations, and applications in human language behavior.

For simplicity to the Reader, all figures, tables, and notations can be queried globally on the previous separated sections.

The Chapter 3, *evolving visual representations*, performs a deeper analysis into the Bag of Visual Words approaches and their extensions, and how evolutionary computation techniques are applied together with BoW.

The Chapter 4, *learning spatiotemporal representations*, encompasses the theoretical aspects regarding the fusion of multimodal features and data temporality, as well as the main approaches used to handle with them: dynamic programming, generative and probabilistic models.

The Chapter 5, *evolving dynamic representations*, aims at the integration of evolutionary computation frameworks into classical approaches used for gesture and action recognition by means of genetic algorithms.

The Chapter 6, *applications for human analysis*, presents an interdisciplinary discussion of the main abstraction levels, which correlate the theoretical contents of previous chapters with the several applications about practical studies performed in real-case scenarios.

The Chapter 7, *conclusions*, presents a discussion of the very global and philosophical aspects of this thesis to be considered, and present the Author's intuitions on future trends in the fields of human language behavior and their divergences between the theory and application.

Finally, the annexed appendix A, *publications*, provides a summary of related publications in impact factor journals and proceedings in several conferences and workshops on the related fields.

Chapter 2

Background

This chapter is divided in two main sections. The first section describes in detail a large number of models based on the BoW paradigm and their integration into the computer vision and machine learning communities for different applications, such as gesture recognition. The second section describes an intermediate level of abstraction of features emerging from those previous representations, which provide clearer represention of behavioral cues or indicators for human language communication that are present in specific contexts, such as conversational settings. The inner sections introduce the content to the next chapters as part of the contributions for the referred literature.

2.1 BoW and BoSG models

This section describes in detail the models based on the BoW paradigm. A motivation for the concept of *words* is presented, providing references on those widely used approaches along the literature. The main aspects to emphasize in this section are the effects of applying BoW-based approaches from the very *static* representations to those *dynamic* representations that include the temporal phenomena, which is usually present in data. In the middle of this literature, the works referring to evolutionary computation take an important role to be integrated as part of these approaches.

2.1.1 Towards Bag of Visual Words

In text mining and information retrieval, the BoW representation is a way to represent documents as numerical vectors, with the aim that such vectorial space captures information about the semantics and content of documents. The idea is to represent a document by a vector of length equal to the number of terms (*e.g.*, words) in the vocabulary associated to the corpus under analysis. Each element of this vector indicates the relevance/importance of the corresponding term for describing the content of the document. Although the BoW makes strong assumptions (*e.g.*, that word order is not important), it is still one of the most used representations nowadays[1]. Thus, in text mining each document is represented using the frequency of appearance of each word in a dictionary.

The success of the BoW representation in the natural language processing domain has inspired researchers in computer vision as well. In the image domain, however, these words become visual elements taken from a certain visual vocabulary. In the computer vision analogy, under the BoVW, an image is represented by a vector indicating the importance of visual words for describing the content of the image. Indeed, each image is decomposed into a large set of patches, either using some type of spatial sampling (grids, sliding window, etc.) or detecting points with relevant properties (corners, salient regions, etc.). Each patch is then described by a numerical descriptor. A set of representative visual words are selected by means of a clustering process over the descriptors. In this scenario, a visual word is a prototypical visual pattern that summarizes the information of other visual descriptors extracted from training images. More specifically, the vocabulary of visual words is typically learnt by clustering visual descriptors extracted from training images. The centers of the resultant clusters are considered as visual words. Commonly, visual descriptors are extracted from points or regions of interest, see [62, 179] for comprehensive descriptions of the BoVW

[1] One should note the text mining community has proposed variants that aim at alleviating such assumptions, *e.g.*, using n-grams [12], still the BoW is very competitive with such formulations.

representation. Some examples of these descriptors are: Scale Invariant Feature Transform (SIFT) [111], Histograms of Oriented Gradients (HOG and HOG3D) [30, 78], Oriented Histograms of Flow and appearance (HOF) [31], Partial Least Squares (PLS) [163], or 3D voxel reconstructions [26]). Once the visual vocabulary is defined, each new image can be represented by a global histogram containing the frequency of occurrences of visual words in the image. Finally, this histogram can be used as input for any classification technique (i.e. K-Nearest Neighbor or SVM) [27, 107].

Currently, the BoVW is among the most used representations for describing the content of images and videos [19, 27, 36, 62, 83, 86, 107, 147, 154, 179], and such representations have obtained outstanding results in a large number of scenarios, as those mentioned before. Moreover, extensions of BoVW from still images to image sequences have been recently proposed in the context of human action recognition, defining Spatio-Temporal-Visual-Words (STVW) [84, 114]. Furthermore, this formulation has trespassed the image and text boundaries and, in fact, it has been used for representing audio [101], time series [166], or accelerometer [60] signals, among others.

2.1.2 Multimodal Gesture Recognition

Nowadays, human gesture recognition is one of the most challenging tasks in computer vision. Current methodologies have shown preliminary results on very simple scenarios, but they are still far from human performance. Due to the large number of potential applications involving human gesture recognition in fields like surveillance [64], sign language recognition [150, 177], or clinical assistance [118] among others, there is a large and active research community devoted to deal with this problem. Independently of the application field, the usual human gesture recognition pipeline is mainly formed by two steps: *gesture representation* and *gesture classification*.

In order to represent these visual features automatically, most approaches are based on classic computer vision techniques applied to RGB data [56, 75, 162]. However, extracting discriminative information from standard image sequences is sometimes unreliable. In this sense, recent studies have included compact multi-modal devices which allow $3D$ partial information to be obtained from the scene. Besides, the release of the Microsoft Kinect™ sensor in late 2010 has allowed an easy and inexpensive access to almost synchronized range imaging with standard video data [1, 2]. Those data combine both sources into what is commonly named RGB-D images (RGB plus Depth). This data fusion has reduced the burden of the first steps in many pipelines devoted to image or object segmentation, and opened new questions such as how these data can be effectively described and fused. At this point, and also considering previous works of the literature [65, 73, 134, 139, 176], the

extraction of human body pose information opens the door to one of the most challenging problems nowadays, such as human gesture recognition.

In [145], the authors proposed a system for real-time human pose recognition including depth information for each image pixel. In this case, information is obtained by means of a Kinect™ device, which estimates a depth map based on the inverse of time response of an infrared sensor sampling within the scene. While some works focus on just the hand regions for performing gesture recognition [15, 38, 76, 90, 117, 165], Shotton introduced one of the greatest advances in the extraction of the human body pose using RGB-D as part of the Kinect™ human recognition framework. The method is based on inferring pixel label probabilities through Random Forest from learned offsets of depth features. Then, mean shift is applied to estimate human joints and representing the body in skeletal form. In [67], authors extended Shotton's work applying Graph-cuts to the pixel label probabilities obtained through Random Forest, in order to compute consistent segmentations in the spatio-temporal domain. Girshick, Shotton et al. [58] proposed later a different approach in which they directly regress the positions of the body joints, without the need of an intermediate pixel-wise body limb classification as in [145]. This source of information has been recently exploited for creating new human pose descriptors by combining different state-of-the-art RGB-D, as well as they are used in a large amount of Human Computer Interaction (HCI) applications [95].

Motivated by the information provided by depth maps, several 3-D descriptors have been recently developed [136, 137] (most of them based on codifying the distribution of normal vectors among regions in the 3D space), as well as their fusion with RGB data [80] and learning approaches for object recognition [16]. As an extension of BoVW for gesture recognition, these approaches also benefit from the multimodal fusion of visual and depth features. Thus, in [66, 68], a new depth descriptor is proposed and combined with state-of-the-art RGB descriptors in a late fusion fashion. The use of this descriptor shows better performance than the traditional BoVW approaches in gesture recognition datasets. Furthermore, this depth information has been particularly exploited for gesture recognition and human body segmentation and tracking.

Dynamic Programming and Generative Models

There exist a large number of works in the literature taking place once human body features are computed [23, 29, 32, 94, 148, 158, 182]. Mainly, these works focus on studying the trajectories generated from those features by means of pattern recognition approaches. In the context of human gesture recognition, some of the methods are based either on dynamic programming techniques such as Dynamic Time Warping (DTW) [68, 116, 133], since it

offers a simple yet effective temporal alignment between sequences of different lengths. Other common methods involve statistical approaches such as Hidden Markov Models (HMM) and Conditional Random Fields (CRF) [150, 152, 177]. These sequential models are especially known for their application in temporal pattern recognition, and they model the system assuming unobservable (or *hidden*) state variables that are inferred from the observations. In fact, an HMM can loosely be understood as a CRF with very specific feature functions that use constant probabilities to model state transitions and emissions. Conversely, a CRF can loosely be understood as a generalization of an HMM that makes the constant transition probabilities into arbitrary functions that vary across the positions in the sequence of hidden states, depending on the input sequence.

Specifically, in the gesture classification step there exists a wide number of methods based on dynamic programming algorithms for both alignment and clustering of temporal series, some of them were reviewed in [183]. However, the application of such methods to gesture detection in complex scenarios becomes a hard task due to the high variability of the environmental conditions among different domains. Some common problems are: wide range of human pose configurations, influence of background, continuity of human movements, spontaneity of human actions, speed, appearance of unexpected objects, illumination changes, partial occlusions, or different points of view, just to mention a few. These effects can cause dramatic changes in the description of a certain gesture, generating a great intra-class variability. Therefore, since usual DTW is applied between a sequence and a single pattern, it fails when taking into account such variability.

In this sense, Probability-based Dynamic Time Warping (PDTW) [11] is proposed as an alternative to the DTW for tackling these common problems. In PDTW, different samples of the same class-sequence pattern obtained from RGB-D data are used to build a Gaussian-based probabilistic model of the class. In particular, we refer to Gaussan Mixture Models (GMM), where a mixture model corresponds to the mixture distribution that represents the probability distribution of observations in the overall population of sequences. Finally, the cost of DTW is adapted accordingly to the new model in order to merge such approaches. The integration of PDTW within a gesture recognition pipeline is used to perform prior segmentation of idle gestures. This approach is tested in a challenging scenario, showing better performance *w.r.t.* to state-of-the-art approaches for gesture recognition in RGB-D data. In Chapter 4.1.1, we explain in more detail the classical DTW approach and describe such situations where generative models provide the possibility of handling with those common problems, integrating them as part of a gesture recognition framework. Moreover, in the section 2.1.4 of this chapter we introduce novel approaches that claim the integration

of evolutionary algorithms to evolve temporal representations based on either dynamic programming or generative models.

2.1.3 Evolutionary Computation over Weighting Schemes

Under the traditional BoW described in the section 2.1.1, the i^{th} document is represented by a vector $\mathbf{d}_i = \langle x_{i,1}, \ldots, x_{i,|V|} \rangle$, where $x_{i,j}$ is a scalar that indicates the importance of the term t_j for describing the content of the i^{th} document; V is the vocabulary, *i.e.*, the set of different words in the corpus. The way in which $x_{i,j}$ is estimated is given by the so called term-weighting scheme. There are many ways of defining $x_{i,j}$ in the text mining and information retrieval literature [33]. Usually, $x_{i,j}$ carries information about both: *term-document relevance* (TDR) and *term-relevance* (TR). The former, explicitly measures the relevance of a term for a document, *i.e.*, it captures local information. The most common TDR is the *term-frequency* (TF) weight, which indicates the number of times a term occurs in a document. On the other hand, TR aims to capture relevance of terms for the task at hand, *i.e.* global information. The most common TR is the *inverse-document-frequency* (IDF), which penalizes terms occurring frequently across the whole corpus. Usually, $x_{i,j}$ combines one TDR and one TR weight.

Perhaps the most common combination is the $TF \times IDF$ weighting scheme [10, 147]. Although this is the standard scheme, for some tasks this may not be the best choice. For instance, in supervised learning tasks, we have information of labels for training samples. However, standard schemes disregard this useful information. This is due to the fact that traditional schemes were originally proposed for information retrieval (an unsupervised problem) [141, 144].

The effectiveness of BoVW representations depends on a number of factors, including the interest-point detection phase, the choice of visual descriptor, the clustering step, and the choice of learning algorithm for the modeling task (*e.g.*, classification) [179]. A factor that has not been deeply studied is the role the term-weighting scheme plays. As in text mining, commonly term-frequency or Boolean term-weighting schemes are considered. Despite the fact these schemes have reported acceptable performance in many tasks (including tasks from natural language processing), it is worth asking ourselves whether alternative schemes can result in better performance. To the best of our knowledge, the only work that aims at exploring this issue is the work by Tirilly et al. [154]. The authors compare the performance of different term-weighting schemes for image retrieval. They considered the most common schemes from information retrieval and provide a comprehensive comparative study. In our work we focus on classification/recognition tasks and consider weighting schemes specifically designed for classification tasks: supervised weighting schemes.

On the other hand, evolutionary algorithms have a long tradition in computer vision. For instance, in [91, 92] genetic programming is used to learn descriptors for action recognition. In [155], evolutionary algorithms are used to evolve interest-point detectors. Moreover, Term-weighting learning with evolutionary algorithms has been studied within information retrieval and text categorization domains [28, 40, 57]. In [28], the authors learn information retrieval weighting schemes with genetic programming. They aim to combine a few primitives trying to maximize average precision. In [40, 57], authors use genetic programming for learning weighting schemes for text classification tasks. In [42], the same algorithms were used to evolve weigthing schemes for image representation.

However, it still remains unknown whether supervised weighting schemes would work for computer vision tasks as well. In the Chapter 3.1, we aim to answer such question throughout an extensive experimental evaluation. In addition, we propose a genetic programming algorithm to learn weighting schemes by combining a set of primitives. One should note that there are efforts for improving the BoVW in several directions, most notably, great advances have been obtained for incorporating spatio-temporal information [19, 68, 86, 96, 107]. The term-weighting schemes developed in this work can also be applied in those scenarios.

2.1.4 Bag of Sub-Gestures

Gesture and action recognition are landmark tasks of the so called Looking at People field [109]; that is, the visual analysis of humans. A wide variety of methods have been proposed since the early nineties [108]. As shown before, the release of the Kinect™ device caused an exponential growth on research in this field [3, 44, 63, 108]. Traditional gesture recognition methods were based on templates (*e.g.*, MHIs [17]), sequence alignment (*e.g.*, DTW [18]) or statistical sequential-modeling (*e.g.*, HMMs [151, 173]). Because of its effectiveness, DTW and HMM based methods are still among the most used techniques nowadays [77, 98, 112]. DTW-based methods align, via dynamic programming, sequences of different length to reference gesture models. The goal is to find the alignment that minimizes a cost given by a distance measure between elements of the sequences. HMMs, on the other hand, are generative models, typically applied to sequential decision problems. Observations sequences are assumed to be generated by a hidden stochastic process. Again, Chapter 4.1.1 provides further details on these methods.

Despite its effectiveness, traditional gesture recognition methodologies approach the problem in a holistic way, where gestures are processed as a whole. Results in related fields with part-based techniques (*e.g.*, in object detection [52] and action recognition [131]) have inspired researchers to build solutions based on *subgesture* models. For instance, in [99] HMMs based on subgestures were proposed. However, subgestures were manually provided

by the users. In [100] a HMM was used to learn subgestures, althought the model was only applied to the problem of hand gesture recognition. In [164] it was proposed a method for segmenting gestures into subgesture units at the frame level. In [122] the authors proposed to use DTW for subgesture modeling, but no results were reported. In [20] subgesture units (defined as cuboids) were learned together with their relationships (using Allen's relations) under a graph-learning framework. Recently, in [170] a relational model for action recognition using dynamic-keyposes was proposed.

Regarding the gesture representation step, literature shows a variety of methods that have obtained successful results. Traditionally applied in image retrieval or image classification scenarios, we have seen the BoVW as one of the most commonly used approaches. Some methods are based on key pose/frame extraction [93, 132, 181] in order to learn a subset of key frames that are highly representative and discriminative for an action class. In [181] an information-theory criterion is adopted for selecting keyframes, whereas in [93] it is used a boosted-based criterion. In [132] a max-margin formulation of the problem is proposed. Very recently, evolutionary algorithms have been also developed for keyframe extraction [21, 22]. In these works, a bag-of-key-poses representation was adopted and an evolutionary algorithm was used to select the number of key-poses for the vocabulary (using $k-$means for clustering), the training set, features and parameters of the model (using DTW for recognition). All of these methods look for a subset of frames, whereas in subgesture modeling we aim at learning spatio-temporal units (subgestures). On the other hand, the above works assume and demonstrate that class-specific key poses/subgestures give a good performance. Nevertheless, we include the fact that some classes may contain or share similar subgestures [122]. Under this additional assumption, our method also reaches the state of the art performance and provides considerable improvements in gesture and action recognition domains.

In the Chapter 5.1, a genetic algorithm is used to evolve gesture primitives integrated into an action recognition framework coupled with either DTW or HMMs. Different from most of the work reviewed in this section, our approach obtains dynamic subgestures (*i.e.*, sequences of frames of different lengths) and simultaneously learns the parameters of the recognition model (either DTW or HMM). Besides, the framework can operate directly as part of deep learning architectures and viceversa, which allows both to evolve deep representations within the evolutionary algorithm and to begin the evolutionary algorithm taking deep representations as the input features.

2.2 Behavioral Indicators in Social Computing domains

In most scenarios for human behavior analysis, it can be observed that both ambient intelligence and egocentric computing methods are defined. Ambient intelligence refers to electronic environments that are sensitive and responsive to the presence of people, whereas egocentric computing refers to the use of wearable devices. Often, existing techniques of data acquisition make use of interface devices [153], or special items such as gloves [50] to increase recognition accuracy. However, while these techniques give impressive results in simulated environments, their use becomes largely infeasible in real-case scenarios due to their intrusiveness and the uncontrollable nature of events that are present. Because of the need to avoid wearing intrusive egocentric devices, some ambient sensors that provide multi-modal data might be considered. In [102], a custom developed system is applied in a real-case scenario for job interviews. The data acquisition procedure is performed using different types of camera, by setting them up in different positions and with different ranges for capturing visual and depth information. Similarly, scenes with non-invasive systems have been proposed in other studies, such as [122], which provides trajectory analyses from body movements and gestures. Furthermore, audio information has been analyzed in [14], with the objective of modeling descriptors for speech recognition.

The analysis of the participants from a computer vision point of view use to be defined by region of interest detection, description, and tracking, usually involving the face or hands. These regions provide discriminative behavioral information, or adaptors, which are movements, such as head scratching, indicative of attitude, anxiety level and self-confidence [104]; or beat gestures, which are small baton-like movements of the hands used to emphasize important parts of speech with respect to the larger discourse [105]. However, as explained in [102, 106], body posture is also found to be an important indicator of a person's emotional state. Additionally, another potential source of information is provided by facial expressions [69, 135, 160, 161].

Once data from the environment have been acquired and processed to define a set of behavioral features, they serve as the basis for modelling a set of communication indicators. For instance, in [174], the authors outline a system for real-time tracking of the human body with the objective of interpreting human behavior. In particular, authors are mainly interested in behavioral traits that represent social signals, which are captured from the communication and the interactions between the participants in the context of conversations. In this sense, levels of agitation (or energy), activity, stress, or engagement are analyzed not only from their body movements, but also from their speech, facial expressions, or gaze directions, so as to predict behavioral responses.

2.2.1 Application in Restorative Justice

The Restorative Justice approach focuses on the personal needs of victims. Achieving success in the VOM (Victim-Offender Mediation) sessions depends largely on how the participants communicate with each other. A large number of techniques can be found in the literature for application in VOM. Rich examples of them can be found in the literature [157]. This resource offers an empirically grounded, state of the art analysis of the application and impact of VOM. It provides practical guidance and resources for VOM in the case of property crimes, minor assaults, and, more recently, crimes of severe violence, where family members of murder victims request a meeting with the offender. Since most of these cases are of a highly sensitive nature, participants manifest emotional states when interacting with the others that can be physically observed through their non-verbal communication [79]. This raises a controversy concerning the different legal frameworks discussed in [13]. However, the handbook [157] collects a set of outcomes demonstrating the competence of restorative justice, as well as several practices developed in the fields of psychology and observational methodology for analyzing both the VOM phases and the participant states.

Recently, a number of studies have proposed ways in which personality traits can be inferred from multimedia data [110] and which can be applied directly to the approach taken by Restorative Justice. The prediction of these responses takes a particular interest in meetings involving a limited number of participants. For instance, in [143] the goal was both to detect the social signals produced in small group interactions and to emphasize their importance markers. In addition, the works of [7, 102] combined several methodologies to analyze non-verbal behavior automatically by extracting communicative cues from both simulated and real scenarios. Additionally, information obtained from speech is commonly used [74, 160, 161]. This can be useful information to measure, for instance, the levels of activity from speech cues, including detection of speech/non-speech, interruptions, pauses, or segments obtained from a speaker diarization process.

Like in the aforementioned studies, in [125, 128] authors demonstrate that indicators of agreement during communication are highly dependent on social signals. As such, it is possible to perform an exhaustive analysis to detect the role played by each participant in terms of influence, dominance, or submission. For instance, In [47], both the interest of observers and the dominant participants are predicted solely on the basis of behavioral motion information when looking at face-to-face (also called *vis-a-vis* or dyadic) interactions. Furthermore, there are many interdisciplinary, state of the art studies examining related fields from the point of view of social computing, some of which are summarized in [118, 119].

In Chapter 6.3.4, we present an intermediate level of abstraction for obtaining behavioral indicators based on communicative cues, which are able to better describe those features

that are directly extracted from multi-modal data. Moreover, such behavioral features are combined together for describing additional behavioral indicators, which are useful to analyze their influence within VOM scenarios.

Chapter 3

Evolving Visual Representations

This chapter presents an evolutionary computation approach for Bag of Visual Words (BoVW) representations based on several weighting schemes. The improvement effects of integrating a genetic programming framework are demonstrated over different datasets of still images and video sequences.

3.1 Evolutionary Algorithms for Bag of Visual Words

As explained in Chapter 2.1.1 The BoVW is a widely adopted representation for describing the content of images and videos in computer vision problems [147]. This representation is the analogy of the Bag of Words (BoW) representation used in text mining and information retrieval: BoVW accounts for the presence and absence of prototypical patterns (called visual words, and playing the role of words in text processing) that are obtained from training images. This representation has obtained outstanding results in a large number of scenarios [19, 27, 36, 86, 107, 147, 154, 179].

In spite of its effectiveness and popularity, most implementations of BoVW adopt pretty standard weighting schemes, that is, the mechanisms that determine the contribution that visual words have for describing the content of images and videos. For instance, the most common scheme is term frequency where the BoVW representation is an histogram that accounts for the occurrences of visual words in the image or video. Although competitive performance has been obtained with this formulation, we think it is worth studying alternative weighting schemes.

This chapter explores the suitability of using alternative term-weighting schemes for image and video representations. On the one hand, we report an evaluation of the most common weighting schemes used in text mining, but rarely used for computer vision tasks. Our study comprises unsupervised and supervised weighting schemes. More importantly, we propose an evolutionary algorithm capable of automatically learning weighting schemes for computer vision problems from scratch. The evolutionary algorithm explores the search space of possible weighting schemes that can be generated by combining a set of primitives with the aim of maximizing the classification/recognition performance. We perform experiments in landmark problems in computer vision, namely: image categorization (different subsets of the Caltech-101 dataset [51]), gesture recognition (the newly introduced Montalbano dataset [44]), action recognition (MSRDaily3D Data) [167], places-scene recognition (the well known 15-scenes [86]), insect and bird classification [85, 87] and adult image classification [178]. Experimental results show the effectiveness of the proposed method.

The remainder of this chapter is organized as follows. Next section introduces the BoVW representation and reviews related work. Section 3.2 presents common and alternative weighting schemes that have been adopted in text mining and information retrieval but that have not been used in computer vision. Section 3.3 describes in detail the proposed methodology for evolving weighting schemes. Next, Section 3.4 reports experimental results. Finally, Section 3.5 outlines conclusions and future work directions.

3.2 Common and alternative weighting schemes

As described in Chapter 2.1.3, one of the most used weighting scheme for information retrieval and text mining tasks is the so called $TF \times IDF$ [10, 144]. Although good results have been reported in many applications using it, alternative weighting schemes have been proposed aiming to capture additional or problem-specific information with the goal of improving retrieval or classification performance [10, 33, 81, 156]. For instance, for text classification tasks, supervised term-weighting schemes have been proposed [33, 81]. These alternatives aim at incorporating discriminative information into the representation by defining TR weights that account for the discriminative power of terms. For instance, by replacing the IDF term (in the $TF \times IDF$ scheme) by a discriminative term IG (the *information gain* of the term), resulting in a $TF \times IG$ scheme. Common and alternative weighting schemes are described in Table 3.1.

Table 3.1 Weighting schemes used in text mining and information retrieval. For every scheme, $x_{i,j}$ indicates how relevant the term t_j is for describing the content of the i^{th} document under the corresponding weighting scheme. Here, N is the number of documents in training dataset, $\#(d_i,t_j)$ indicates the frequency of term t_j in the i^{th} document, $df(t_j)$ is document frequency of the term t_j, *i.e.*, the number of documents in which term t_j occurs, $IG(t_j)$ is the information gain of term t_j, $CHI(t_j)$ is the χ^2 statistic for term t_j, and TP, TN are the true positive and true negative rates for term t_j (*i.e.*, number of positive, respectively, negative, documents that contain term t_j).

Acr.	Name	Formula	Description	Ref.
B	Boolean	$x_{i,j} = \mathbf{1}_{\{\#(d_i,t_j)>0\}}$	Prescense/abscense of terms	[141]
TF	Term-Frequency	$x_{i,j} = \#(d_i,t_j)$	Frequency of occurrence of terms	[141]
TF-IDF	TF - Inverse Doc. Freq.	$x_{i,j} = \#(d_i,t_j) \times \log(\frac{N}{df(t_j)})$	TF penalizing corpus-based frequency	[141]
TF-IG	TF - Information Gain	$x_{i,j} = \#(d_i,t_j) \times IG(t_j)$	TF times term information gain	[33]
TF-CHI	TF - Chi-square	$x_{i,j} = \#(d_i,t_j) \times CHI(t_j)$	TF times χ^2 term relevance	[33]
TF-RF	TF - Relevance Freq.	$x_{i,j} = \#(d_i,t_j) \times \log(2+\frac{TP}{\max(1,TN)})$	TF times *RF* relevance	[81]

The first three weighting schemes in Table 3.1 are common in text mining and information retrieval, and their usage dates back to the 80s [141], being the Boolean scheme the simplest one (only accounting for the occurrence of terms). On the other hand, the last three schemes were proposed in the last decade and still are not well known within text mining. To the best of our knowledge, these alternative weighting schemes have not been evaluated in the context of computer vision (see Chapter 2.1). Therefore, a first contribution of this chapter is to assess the suitability of such schemes for computer vision problems. The next section introduces our evolutionary algorithm for learning term-weighting schemes for the BoVW.

3.3 Evolving visual-word weighting schemes

In addition to the evaluation of non traditional weighting schemes in computer vision, a second contribution of this work is the proposal of an evolutionary algorithm capable of automatically determining new weighting schemes from scratch. Our proposal is motivated by the following observations. First, we observe that traditional weighting schemes were proposed by researchers based on their own expertise, biases, and needs. Also, so far, it has been the norm to use the same weighting scheme for every dataset under analysis. In fact, in computer vision tasks, the weighting scheme is rarely considered a factor that can have an impact on the performance of models based on the BoVW formulation.

In this chapter, we address the question of whether the weighting-scheme design process can be automated by employing evolutionary algorithms. Our proposed method uses genetic programming to learn how to combine a set of TDR/TR primitives with the aim of obtaining a weighting scheme that optimizes classification performance. This term-weighting-scheme learning formulation removes, to some extent, the biases of designers and does not rely on user expertise[1]. Instead, weighting schemes are sought such that they maximize the performance in the task under analysis. Hence, our automatic technique allows us to learn tailored schemes for every dataset / task being approached.

Figure 3.1 presents a general diagram of the proposed approach. A set of primitives is extracted from the BoVW representation of training images. These primitives are obtained by counting visual word occurrence statistics. Next, they are feed into a genetic program that learns how to combine such primitives to generate a term-weighting scheme. The output of the genetic program is a way to represent images that has been learned automatically. Next, both training and test images are represented according to the learned scheme and, finally, a predictive model is learned and their performance evaluated. The remainder of this section describes our proposed method.

3.3.1 Genetic Programming

Our solution to learn term-weighting schemes is based on Genetic Programming (GP) [82]. GP is an evolutionary algorithm, that is an optimization algorithm inspired by biological evolutionary systems. In evolutionary algorithms solutions to the problem at hand are seen as individuals that interact among them and with the environment (the search space) in such a way that the survival of the population is sought (optimization criterion). The general flow of a typical evolutionary algorithm is shown in Figure 3.2: an initial population of

[1]Please note that traditional weighting schemes have been proposed by researchers based on their own experiences and biases, making strong assumptions and relying on intuition.

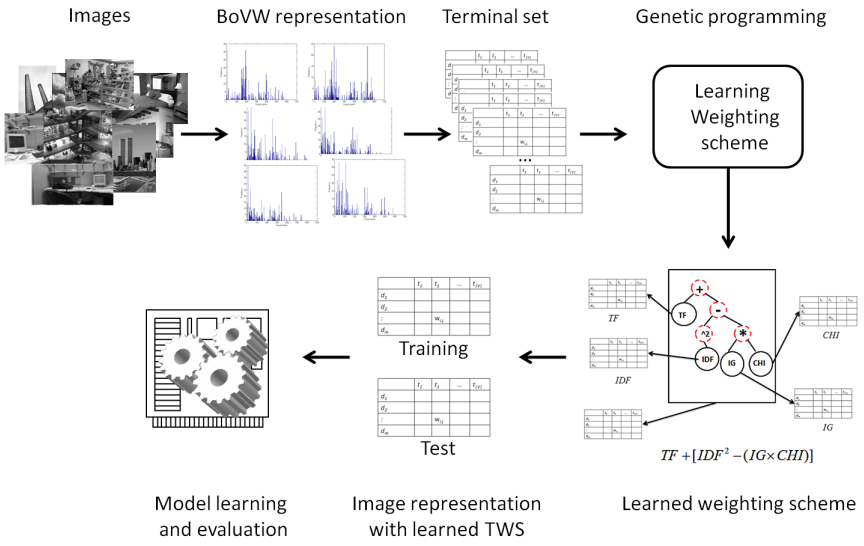


Fig. 3.1 General diagram of the proposed approach.

solutions/individuals is created (randomly or by a pre-defined criterion), after that, individuals are selected, recombined[2], mutated and then placed back into the solutions' pool, this process is repeated for a given number of generations and the algorithm returns the best individual found.

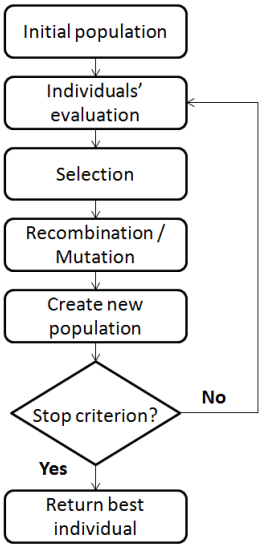


Fig. 3.2 A generic evolutionary algorithm.

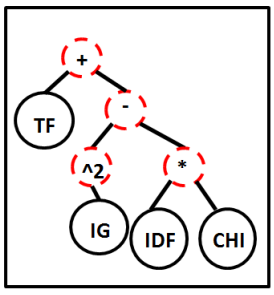
The main distinctive feature of GP, when compared to other evolutionary algorithms, is that in GP, nonlinear and complex data structures are used to represent solutions (individuals).

[2]Please note that in GP, for each individual, either mutation or crossover is performed each time, but not both. This is different from other variants like genetic algorithms.

For instance, the most common representations for individuals in GP are trees and graphs, whereas for most of evolutionary algorithms, numerical vectors are used. This feature of GP makes it appropriate for facing very complex problems, in most cases related to modeling tasks. This is one of the reasons for which we adopted GP for learning weighting schemes. Nevertheless, the main motivation for using GP for our problem is that we are interested in learning a function that tell us how to combine the different primitives (including the decision of telling which primitives are worth to combine). In this scenario, GP provides a natural solution to the problem, encoding candidate functions as individuals (*i.e.*, trees) and searching for the best one. Clearly, this problem cannot be approached with either traditional optimization or heuristic optimization techniques.

3.3.2 GP for Term-Weighting Scheme learning

Our approach to generate weighting schemes uses genetic programming to learn how to combine a set of primitives that have been used for building weighting schemes in the past (see Figure 3.1). That is, we devise a genetic program that searches for the combination of primitives that maximizes the classification performance of the task under analysis (*e.g.*, image classification). A standard tree representation is adopted in which leafs correspond to primitives and non-terminal nodes correspond to operators by which primitives can be combined; in such a way that the evaluation of a tree leads to a term-weighting scheme (see Figure 3.3).



$$W = TF + [IG^2 - (IDF \times CHI)]$$

Fig. 3.3 Adopted representation for individuals. Dashed nodes represent operators (taken from the function set) and solid-line nodes indicate terminals; below the tree we show the term-weighting scheme derived from it.

Therefore, under this formulation, we explore the search space of weighting schemes that can be coded by the trees, where, common/alternative weighting schemes are included in the search space. The remainder of the section elaborates on the different components of the proposed genetic program.

Representation

As mentioned before, weighting schemes are mainly composed out of two type of factors: TDR an TR weights, which determine the importance of terms into documents and the relevance of terms themselves, respectively. Accordingly, the proposed method uses as terminals TDR and TR primitives (together with useful constants and other weighting schemes), which can be combined by a predefined set of operators. An individual (*i.e.*, solution) in the genetic program is thus a tree formed by these terminals and operators, where the evaluation of the tree leads to a term-weighting scheme. Figure 3.3 depicts a typical individual and the resultant weighting scheme.

The set of terminals considered in this work is shown in Table 3.2, whereas for the operators (non-terminals) we considered the function set shown in Table 3.3.

Table 3.2 Terminal set.

Variable	Meaning
$\mathbf{W}_1$	N, Constant matrix, number of training documents.
$\mathbf{W}_2$	$\|\|V\|\|$, Constant matrix, number of terms.
$\mathbf{W}_3$	CHI, Matrix containing in each row the vector of χ^2 weights for the terms.
$\mathbf{W}_4$	IG, Matrix containing in each row the vector of information gain weights for the terms.
$\mathbf{W}_5$	$TF \times IDF$, Matrix with the TF-IDF term-weighting scheme.
$\mathbf{W}_6$	TF, Matrix containing the TF term-weighting scheme.
$\mathbf{W}_7$	FGT, Matrix containing in each row the global term-frequency for all terms.
$\mathbf{W}_8$	TP, Matrix containing in each row the vector of true positives for all terms.
$\mathbf{W}_9$	FP, Matrix containing in each row the vector of false positives.
$\mathbf{W}_{10}$	TN, Matrix containing in each row the vector of true negatives.
$\mathbf{W}_{11}$	FN, Matrix containing in each row the vector of false negatives.
$\mathbf{W}_{12}$	*Accuracy*, Matrix where each row contains the accuracy obtained when using the term as classifier.
$\mathbf{W}_{13}$	*Accuracy_Balance*, Matrix containing the AC_Balance each (term, class).
$\mathbf{W}_{14}$	Bi-normal separation, *BNS*, An array that contains the value for each BNS per (term, class).
$\mathbf{W}_{15}$	*DFreq*, Document frequency matrix containing the value for each (term, class).
$\mathbf{W}_{16}$	*FMeasure*, F-Measure matrix containing the value for each (term, class).
$\mathbf{W}_{17}$	*OddsRatio*, An array containing the OddsRatio term-weighting.
$\mathbf{W}_{18}$	*Power*, Matrix containing the Power value for each (term, class).
$\mathbf{W}_{19}$	*ProbabilityRatio*, Matrix containing the ProbabilityRatio each (term, class).
$\mathbf{W}_{20}$	*Max_Term*, Matrix containing the vector with the highest repetition for each term.
$\mathbf{W}_{21}$	RF, Matrix containing the RF vector.
$\mathbf{W}_{22}$	$TF \times RF$, Matrix containing TF-RF.

Each terminal in Table 3.2 is a matrix of size $N \times |V|$. TDRs are themselves matrices of that dimensions, but TRs are row vectors of length $|V|$ (*i.e.*, they indicate the relevance of each term). To make all matrices comparable (and henceforth suitable for combination under the function set $\mathscr{F}$), TRs are converted into matrices by repeating the row vector N

times. Therefore, all of the operators in the function set act on a scalar basis, that is, they are applied element-by-element. It is worth mentioning that for supervised TR factors, we use information extracted from training images only; *i.e.*, no supervised information is used from the test set.

Table 3.3 Considered function set for the genetic program.

Operator	Name	Arity
$+$	Addition	2
$-$	Substraction	2
$*$	Product	2
$/$	Division (protected)	2
$\log_2 x$	Logarithm b-2	1
$\sqrt{x}$	Square root	1
x^2	Square power	1

The initial population is generated with the ramped half-half strategy, which means that half of the population is created with the full method (*i.e.*, all trees have the same deep, $maxdepth$) and the other half is created with the grow method (*i.e.*, trees have deep of at most $maxdepth$), see [82] for details.

Fitness function

The goal of our genetic programming formulation is to obtain a weighting scheme that maximizes classification performance. Therefore, the goodness / fitness of each solution should be tied to the classification performance of a model using the representation induced by the weighting scheme. Specifically, given a solution to the problem, we first evaluate the tree to generate a weighting scheme using the training set, as shown in Figure 3.3. Once training documents are represented by the corresponding weighting scheme, we perform a $k-$fold cross-validation procedure, using a given classifier, to assess the effectiveness of the solution. In $k-$fold cross validation, the training set is split into k disjoint subsets, and k rounds of training and testing are performed; in each round $k-1$ subsets are used as training set and 1 subset is used for testing, the process is repeated k times using a different subset for testing each time. The average classification performance is used as the fitness function.

In particular, we evaluate the performance of classification models with the f_1 measure. Let TP, FP and FN to denote the true positives, false positives and false negative rates for a particular class, precision ($Prec$) is defined as $\frac{TP}{TP+FP}$ and recall (Rec) as $\frac{TP}{TP+FN}$. f_1-measure is simply the harmonic average between precision and recall: $f_1 = \frac{2\times Prec\times Rec}{Prec+Rec}$. The average across classes is reported (also called, macro-average f_1), this way of estimating the f_1-measure is known to be particularly useful when tackling unbalanced datasets.

Because under the fitness function k models have to be trained and tested for the evaluation of a single solution, we need to look for an efficient classification model. We considered Support Vector Machines (SVM) as they can deal naturally with the sparseness and high dimensionality of data. However, training and testing an SVM can be a time consuming process. Therefore, we opted for efficient implementations of SVMs that have been proposed recently [37, 180]. Those methods are trained online and under the scheme of learning with a budget. We use the predictions of an SVM as the fitness function for learning term-weighting schemes (TWS). Among the methods available in [37] we used the low-rank linearized SVM (LLSMV) [180]. LLSVM is a linearized version of non-linear SVMs, which can be trained efficiently with the so called block minimization framework [25]. We selected LLSVM instead of alternative methods because this method has outperformed several other efficient implementations of SVMs (see [37, 180]). Thus, we use this approximated SVM during the fitness function. Once a weighting scheme has been learnt, however, we use a deterministic SVM to classify the test set. This is to make results comparable and discard the randomness inherent to the approximate solutions.

Genetic operators

The proposed genetic program follows a standard procedure as depicted in Figure 3.2. We use the implementation from [146], which considers standard operators for crossover and mutation. Specifically, subtree crossover is considered where, given two parent trees, an intermediate node is randomly selected within each tree. Then, the subtrees below the selected nodes are interchanged between the parents, giving rise to two offspring. The mutation operator is quite standard as well, it consists of identifying a node within the parent tree and replacing the node with another randomly selected (terminals replaced by terminals and non-terminals replaced by operators in $\mathscr{F}$).

Final remarks

After the evolutionary process finishes, the genetic program returns a term-weighting scheme. Next, training and test images are represented according to this scheme. A classifier is learnt using the training representation and its performance evaluated in the test representation. For this evaluation we consider a deterministic SVM (from the CLOP toolbox [140]), hence, results are comparable to each other. The next section reports experimental results on several computer vision tasks obtained with learned weighting schemes.

3.4 Experiments and results

This section presents experimental results that aim at showing the effectiveness of the proposed methodology for learning term-weighting schemes in a variety of computer vision tasks. First we describe the experimental settings and then report results of our study.

3.4.1 Settings

For experimentation we considered standard datasets associated to landmark computer vision tasks. The considered datasets are described in Table 3.4. All of these datasets are associated to classification/recognition tasks, hence the same evaluation protocol (with slight variations described below for each dataset) was adopted. For all but one dataset we generated training and test partitions[3]; the exception was the MSRDaily3D dataset for which we report average performance over 5-fold cross validation, see below.

In every dataset, the training partition was used both to obtain the visual vocabulary and to learn the term-weighting schemes with the genetic program, recall the program maximizes the f_1 measure under $k-$fold cross validation. For evaluating the performance of the different weighting schemes, both, training and test images are represented with the schemes (either learned or predefined). Then, a classification model is learned using training images and the performance of the model is evaluated in test images.

Unless otherwise stated, we used the VLFEAT toolbox for processing images [159]. We considered PHOW[4] (Pyramid Histogram Of Visual Words) features as visual descriptors [19].

Regarding our proposed genetic program for term-weighting learning, the average and standard deviation performance of 5 runs is reported. The method was run in all cases for 50 generations with a population of 500 individuals. This is a very standard choice for GP [82], where it is common to use large number of individuals and a small number of generations. Default values were used for the remainder of GP parameters: generational selection mechanism with elitism, lexictour parent selection [97], crossover probability of 0.9, and mutation probability of 0.1.

Because the optimization process may be too time consuming for some datasets, we learned the weighting schemes by using subsets of the original training sets:

- Only samples belonging to a subset of classes were used. In some cases, the vocabulary was also reduced, see Table 3.4 column 6.

[3] Matlab files with the predefined partitions are publicly available under request.

[4] PHOW is an extension to the raw BoVW formulation that aims at incorporating spatial information by means of a pyramidal structure, see [19] for details.

Table 3.4 Datasets considered for experimentation. Column 6 shows the number of *images* | *terms* (*i.e.*, size of the visual vocabulary) considered during the search process.

Image Categorization					
Dataset	**Classes**	**|V|**	**# Train**	**# Test**	**images|terms**
Caltech-tiny	5	12000	75	75	15|12000
Caltech-102 (15)	101	12000	1530	1530	165|3000
Caltech-102 (30)	101	12000	3060	3060	330|3000
Birds	6	400	540	60	540|400
Butterflies	7	400	552	67	552|400
Action recognition					
Dataset	**Classes**	**|V|**	**# Train**	**# Test**	**im.|terms**
MSRDaily3D	12	600	192	48	192|600
Gesture recognition					
Dataset	**Classes**	**|V|**	**# Train**	**# Test**	**im.|terms**
Montalbano	20	1000	6850	3579	2055|600
Scene recognition					
Dataset	**Classes**	**|V|**	**# Train**	**# Test**	**im.|terms**
15 Scenes	15	12000	1475	3010	1475|2000
Pornographic image filtering					
Dataset	**Classes**	**|V|**	**# Train**	**# Test**	**im.|terms**
Adult	5	12000	6808	1702	6808|2000

- The selection of classes was done randomly; while the vocabulary reduction used a frequency criterion (the most frequent terms were retained).

Despite this reductions, at the end of the search process, all of the data and classes are considered for training the final classifier and evaluation. We emphasize that during the search process we use an approximate SVM for computing the fitness function. When evaluating the performance of weighting schemes in test set we used a deterministic linear SVM. Specific details and considerations for each dataset are reported below.

Finally, for comparing the statistical-significance of differences we used a Wilcoxon signed-rank test (as recommended in [35]).

Caltech-101

Caltech-101 [51] is a mandatory benchmark for image classification. It contains objects that belong to 101 different categories (102 including the background category). Sample images from this dataset are provided in Figure 3.4.

For experiments we considered three subsets: tiny, 101-15 and 101-30. Tiny considers 5 out 102 classes with 15 images per-class for training and 15 for testing; dataset 101-15 considers the 102 classes with 15 training and 15 testing images (per-class); finally, dataset 101-30 considers the 102 classes with 30 images for training and 30 for testing. Using 3 subsets of Caltech-101 allows us to evaluate the performance of our method for

Fig. 3.4 Sample images from the Caltech-101 dataset.

similar categorization problems but with different complexities in terms of the number of categories and samples. In fact, we use these subsets of Caltech-101 to assess the generality capabilities of the proposed approach, see below. For tiny we used all of the samples during the optimization process, whereas for the other two datasets we used examples from 10 category-classes and the background only, where the top 3000 terms where considered.

Birds and butterflies

We also considered two datasets related to animal recognition: birds and butterflies. Figure 3.5 shows sample images from these datasets. In both cases, the problem is to distinguish birds/butterflies species. Contrary to Caltech-101, these datasets comprise more fine-grained classification problems. Therefore, these datasets comprise a major challenge because instances of different classes may be very similar. For these datasets we represented images under the BoW using a Discrete Cosine Transform (DCT) descriptor. This choice is based on previous work in the same datasets [96]. For both datasets, we used 90 percent of images for training and 10 percent of images for testing.

Fig. 3.5 Sample images from different categories of the Birds and Butterflies datasets.

Adult image filtering

A dataset for adult image filtering was considered as well. The data was made available by [36], and it has been previously used in several publications, see [36, 178]. The dataset contains images belonging to five categories, where there is one category for inoffensive images and four categories of increasing level of *adultness*: lightly dressed, partly nude, nude and pornographic, see Figure 3.6.

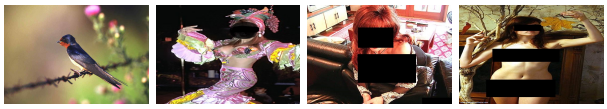


Fig. 3.6 Sample images from the dataset of adult image filtering. The categories are (from left to right): inoffensive images, lightly dressed persons, partly nude persons, nude persons, and pornographic images (not shown).

The goal in this task is to associate images with its correct category in such a way that the administrator of a filtering system can decide the level of restriction in the type of images users can have access to (*e.g.*, photos of lightly dressed persons may be allowed in most sites, even in schools, but nude-persons and pornography may be objectionable in most sites). About 80% of images were used as training set and the remainder as test set, as in [36].

Scene recognition

We consider a benchmark dataset for scene recognition [86]. The dataset comprises 15 indoor/outdoor categories, where images contain complex scenes. Figure 3.7 shows sample images from this dataset, clearly this is a very challenging task. For this dataset we used the same partitioning proposed in [86]: 100 images per category for training and the rest for testing.

Fig. 3.7 Sample images from the 15-Scenes dataset. Categories are from left to right and from up to bottom: *bedrom, suburb, industrial, kitchen, living-room, coast, forest, highway, inside-city, mountain, open-country, street, tall-building, office, and store.*

Montalbano

The BoVW has been used to represent videos as well, see *e.g.*, [68, 83, 147]. For this reason we also decided to include video datasets. Specifically, we considered the Montalbano dataset for gesture recognition as provided in [44]. The task consists of recognizing gestures from 20 categories (Italian cultural gestures), see Figure 3.8. The available data is depth and RGB video together with skeleton information. For our experiments we used the features proposed in [113], which combine depth, RGB video and skeleton information by means

of convolutional nets and other deep learning mechanisms. The deep-learning features were clustered and the vocabulary was built. One should note that we approach the gesture recognition problem, that is, given a segmented gesture, to tell the class of the gesture being performed.

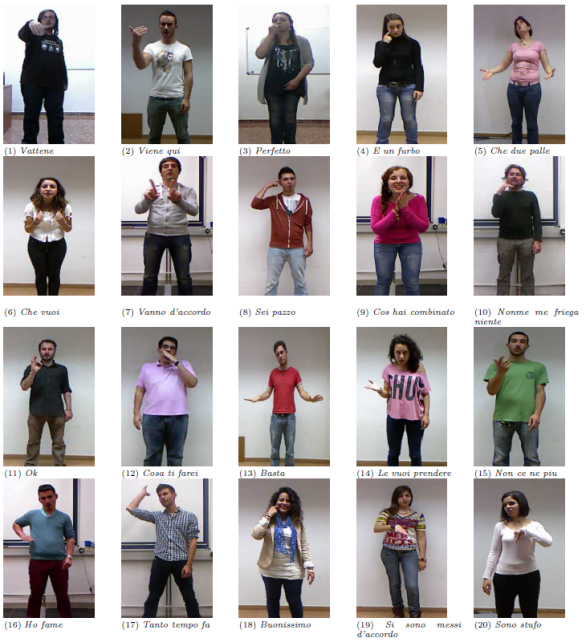


Fig. 3.8 Sample images from the Montalbano dataset. Images from each of the gesture categories are shown [44].

MSRDaily3D

Finally, we considered a benchmark dataset for action recognition: MSRDaily3D. This dataset comprises 16 actions associated to daily activities, where there are objects in the background and most actions involve human-object interaction. A sample sequence from this dataset is shown in Figure 3.9. For this dataset we adopted the protocol from [70–72, 175]. Under this setting we considered 12 out of the 16 actions and performed 5-fold cross validation. We adopted this protocol because it has been adopted in recent work that uses the BoW representation [70–72, 175], therefore we can compare the performance of our method with such works.Video sequences were represented with Depth Cuboid Similarity Features (DCSF) and the same parameters for the descriptor as in previous work were used. Descriptors were further processed to represent videos with their bag of features representation.

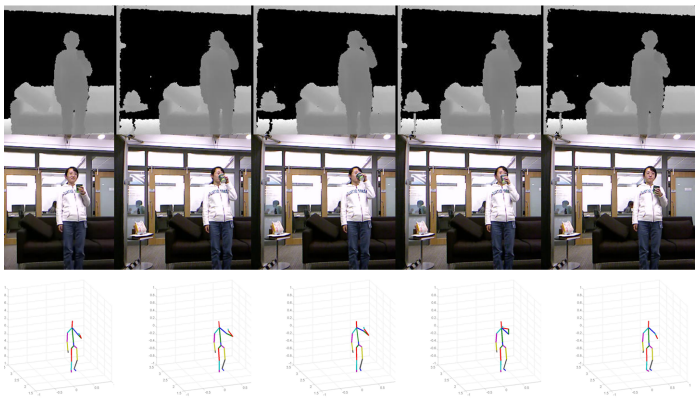


Fig. 3.9 Sample sequence from the MSRDaily3D dataset [167].

3.4.2 Results

Table 3.5 shows the results obtained by the different weighting schemes (traditional, alternative-supervised and learned) in all of the considered datasets. We report average f_1-measure performance in the test-partitions. The $\star$ symbol indicates a statistically significant difference between our approach and the method from the corresponding columns.

It can be seen from this table that, in average, the Boolean weighting scheme (column 3) outperforms both, traditional and alternative, term-weighting schemes. This is an interesting result, because, most of the times (normalized) TF or TF-IDF weighting schemes are considered in computer vision tasks. Please note that although the Boolean scheme is the best on average, it is clear from Table 3.5 that there is no single best weighting scheme for all of the datasets.

Regarding alternative-supervised term-weighting schemes, only TF-RF obtained comparable performance to the TF scheme, however its performance was lower than the Boolean scheme. The other two supervised schemes performed worse than the baseline. These results are somewhat disappointing, because, intuitively, the incorporation of discriminative information should yield better performance. In spite of these results, our study comparing traditional and alternative weighting schemes is a contribution that brings some light on the performance of such schemes for diverse computer vision tasks. More importantly, we showed the adequacy of the Boolean scheme.

On the other hand, it is clear from Table 3.5 that the proposed approach for learning visual-word weighting schemes outperforms all the other variants in all of the considered datasets (see column 8). For most of the datasets, our GP-based solution improves considerably the performance of all of the other weighting schemes. The average improvement of our genetic program over the Boolean scheme was of around 5%, we think this improvement makes worth applying our method instead of relying on standard weighting schemes. These results

Table 3.5 Classification performance obtained with traditional, alternative and learned weighting schemes. The $\star$ symbol indicates a statistically significant difference between our approach and the method from the corresponding columns.

	Traditional			**Alternative-supervised**			**Learned**
Dataset / TWS	**TF (baseline)**$\star$	**Bol.**$\star$	**TF-IDF**$\star$	**TF-RF**$\star$ [81]	**TF-CHI**$\star$ [33]	**TF-IG**$\star$ [33]	**GP (ours)**
Tiny	85.65	84.01	76.72	85.65	78.85	80.49	**90.75$^{+}_{-}$1.56**
101-15	52.26	58.43	48.08	52.30	52.00	51.43	**61.05$^{+}_{-}$1.12**
101-30	56.61	59.28	49.95	56.68	54.63	52.03	**63.04$^{+}_{-}$1.02**
Birds	44.68	48.53	30.55	44.68	44.6	43.95	**52.95$^{+}_{-}$5.11**
Butterflies	26.07	41.44	20.45	26.07	26.08	26.75	**42.12$^{+}_{-}$3.07**
Adult	52.53	58.35	55.39	52.53	46.39	47.23	**62.68$^{+}_{-}$2.08**
15 scenes	59.12	61.26	56.51	59.12	55.02	55.07	**63.43$^{+}_{-}$0.16**
Montalbano	88.55	86.46	88.49	88.55	88.5	88.58	**88.79$^{+}_{-}$0.12**
MSRDaily3D	75.22$^{+}_{-}$4.2	68.0$^{+}_{-}$6.22	74.72$^{+}_{-}$4.47	75.058$^{+}_{-}$3.9	73.94$^{+}_{-}$5.65	73.77$^{+}_{-}$4.9	**76.01$^{+}_{-}$4.01**
Average	54.34$^{+}_{-}$22.06	56.91$^{+}_{-}$18.78	50.81$^{+}_{-}$22.38	54.33$^{+}_{-}$22.04	52.46$^{+}_{-}$21.04	52.51$^{+}_{-}$21.11	**61.45$^{+}_{-}$18.67**

show that, if searched properly, weighting schemes that maximize classification performance may result in improved performance; this is in contrast to using discriminative information by using IG, CHI, etc.

Higher improvements were observed for image categorization and adult-image filtering datasets. Whereas marginal improvements were observed for Montalbano and MSRDaily. The latter behavior can be due to the fact that the descriptors used for these datasets are very discriminative as reported in [44, 113, 175]. In those cases it may be enough to verify the presence / absence of such discriminative patterns. This is not the case of image categorization datasets for which standard descriptors were used.

In addition to the competitive average performance, it is quite interesting that the standard deviation across runs is relatively low when compared to the other methods. Thus evidencing the stability and robustness of the proposed method.

In order to better appreciate the improvements offered by our method, Figure 3.10 shows the range of improvement of our method over the best traditional/alternative weighting scheme per dataset in terms of absolute and relative differences. That is, we plot the difference in performance between our method (column 8) and the best result among columns 2-7 for each particular dataset. This means that our method is not compared with the best scheme in average, but with the best overall for each dataset, a somewhat unfair comparison for our approach.

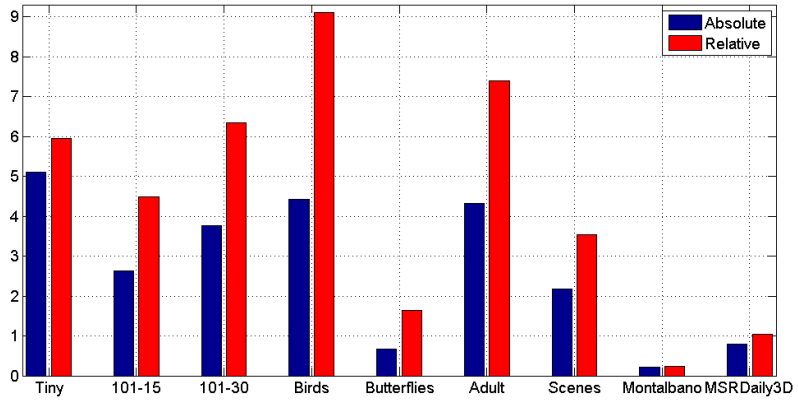


Fig. 3.10 Absolute (blue-first bar) and relative (red-right bar) improvement for the different datasets, taking as reference the best traditional/alternative weighting scheme for each dataset.

From Figure 3.10 it can be seen that the GP-based method offers considerable improvements for all but for the Montalbano dataset. The difficulty of this task may require running the genetic programm using the whole number of classes/samples (for this dataset we used only a third of the total of instances, see column 6 in Table 3.4). Although, as mentioned

above, we think this low improvement is due to the very effective visual descriptors over which the BoW representation was generated.

One should note that the proposed method relies on an iterative optimization process that is somewhat computational expensive. In particular, the adopted representation (tree-based structure), the fact that the terminals are associated to matrices and the estimation of the fitness function[5] (training and testing an SVM classifier under a cross validation) are the main factors that contribute to the computational expensiveness of our model. Nevertheless, in practice, the average running time of the proposed method takes of the order of a few hours. Thus, although the proposed method is somewhat computational expensive, the average running time is acceptable for most computer vision applications. Please note that the process of learning weighting schemes is a procedure that is performed offline, and has to be done a single time. Therefore, we think it is worthwhile spending a few hours using our method, given the potential improvement in performance that can be obtained. On the other hand, one may argue that alternative weighting schemes are less complex (and henceforth require of less processing time to generate the representation). We think this time is negligible, because it involves only a few additional arithmetic operations over more matrices (which are also computed a single time).

3.4.3 Qualitative analysis

This section presents a qualitative study on the proposed method for learning term-weighting schemes. Table 3.6 shows sample schemes learned for selected datasets. It can be seen that all of the learned schemes included primitives that capture from supervised information. Thus, showing the importance of such supervised components. Therefore, we can say that the proposed method effectively learns to combine supervised building blocks that result in competitive weighting schemes. This is in contrast with alternative-supervised schemes that showed limited performance (see Table 3.5).

From Table 3.6 it can be seen that the learned weighting schemes are indeed simple expressions (opposed to standard GP solutions that include very complex trees). This is a desirable property that suggests overfitting is not an issue for the proposed method.

Finally, it is interesting to note that very different weighting schemes were obtained for the different datasets, thus giving evidence that a tailored weighting scheme is required for each task.

[5]Please note that estimating the fitness function is quite efficient, as it is based on a fast approximation to a linear SVM. So this method can be used for most computer vision applications. Also, we emphasize that the fitness function is only estimated during the learning process, which has to be done a single time and most of the times is performed offline.

Table 3.6 Sample weighting schemes learned with the proposed approach for selected datasets. In column 2 each weighting is shown as a prefix expression. The names of the variables are self-explanatory. Column 3 shows the mathematical expression of each TWS using the terminal set from Table 3.2.

ID	Dataset	Learned TWS	Formula
1	Caltech101-15	sqrt((sqrt(RF× TF)+log 2(RF× TF)))	$\sqrt{\sqrt{W_{22}}+\log 2(W_{22})}$
2	Birds	log 2((FMeas × (CHI × log 2(TF × RF))))	$\log 2(W_{16} \times (W_3 \times \log 2(W_{22}))$
3	MSRDaily3D	((TF × FN) × sqrt(T))	$((W_6 \times W_{11}) \times \log 2(\sqrt{W_{22}}))$
4	Adult	(sqrt(IDF)×D)	$(\sqrt{W_5} \times D)$
5	Montalbano	log 2(log 2(CHI))× sqrt(IDF)	$(\log 2(\log 2(W_3)) \times \sqrt{W_5})$
6	15-Scenes	log 2((ProbR + TF × RF))	$\log 2(W_{19}+W_{22})$

Figure 3.11 shows the frequency of use of each of the terminals from Table 3.2 in the solutions returned by the genetic program for all of the datasets (*i.e.*, a bar in Figure 3.11 corresponds to a row in Table 3.2). It can be seen that three most used terminals are W_6, W_{22} and W_5, which correspond to TF, TF-RF and TF-IDF weighting schemes. This is interesting because, even when these were the most chosen terminals by solutions returned with the genetic program, such terminals were significantly outperformed by our proposal: compare columns 2, 4 and 5 to column 8 in Table 3.5.

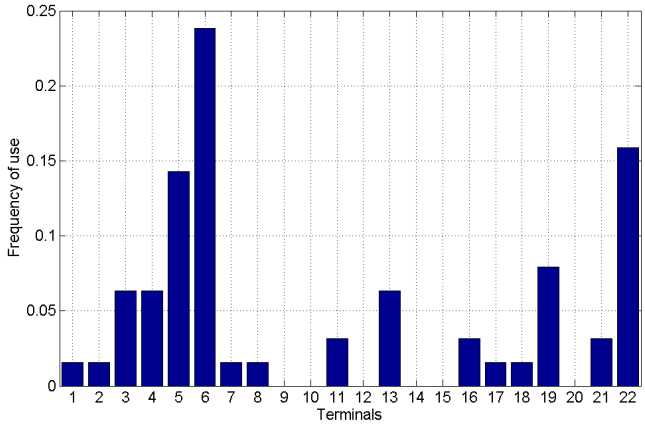


Fig. 3.11 Frequency of appearance of terminals into the solutions found by the genetic program, see Table 3.2 for terminals description.

Only 6 out of the 22 terminals did not appear in solutions returned by the genetic program. All of these terminals ($W_{9,10,12,14,15,20}$) corresponding to TR weights, mainly used for feature selection in text classification [54]. Although they have proved to be very effective in [54] (terminal W_{14} was the best criterion for feature selection in that study), they were not very helpful for building term-weighting schemes for computer vision tasks.

3.5 Conclusion

The BoVW is one of the most used representations in computer vision tasks. Despite being very effective, it is somewhat surprising that little research has been performed on term-weighting schemes for computer vision. In this direction, this chapter introduced a novel methodology for learning weighting schemes to boost the performance of classification models relying on the BoVW. The proposed methodology resulted very effective in a wide variety of computer vision tasks. Additionally, we report an in-depth study on the performance of standard and alternative weighting schemes commonly used in text mining. To the best of our knowledge, our work is the first that assesses alternative weighting schemes, and it is the first in proposing methods to learn weighting schemes for computer vision tasks. From our extensive experimental study, comprising 9 datasets of common computer vision task we can conclude the following:

- Among traditional and alternative weighting schemes, the Boolean one obtained the highest performance.
- Weighting schemes learned with our proposed approach outperformed consistently all other weighting schemes in all of the datasets.
- For different tasks, learning a term-weighting scheme with the proposed approach is much better than applying other schemes (either traditional / alternative or learned for another dataset).
- Computer vision tasks that are not too generic *e.g.*, gesture recognition or adult image filtering) require of tailored weighting schemes, accordingly, schemes learned for this datasets do not generalize well in other datasets.
- Among all of the considered terminals, three weighting schemes were used most often by solutions returned by the genetic program (TF, TF-IDF and TF-RF), however, the way in which the genetic program combined such primitives resulted in much better performance.

Future work includes studying alternative methodologies for learning term-weighting schemes. Specifically, we plan to pose the problem as one of learning/optimizing the representation matrix, where other evolutionary algorithms could be used. Also, we are interested on learning term-weighting schemes for other domains, like audio [101], time series [166] or accelerometer data [60].

Chapter 4

Learning SpatioTemporal Representations

This chapter presents an extension of two well-known approaches for gesture recognition: Dynamic Time Warping (DTW) and Bag of Visual Words (BoVW). Their extension consist, first, of integrating probabilistic modelling into the classical DTW for the segmentation of sequences, and the inclusion of the depth modality for describing novel multimodal features for the task of gesture recognition.

4.1 BoVDW and Probability-based DTW for Human Gesture Recognition

The problem of gesture recognition in which an idle or reference gesture is performed between gestures is addressed in this section. In order to solve this problem, we introduce a continuous human gesture recognition pipeline based on: First, a new feature representation by means of a Bag-of-Visual-and-Depth-Words (BoVDW) approach that takes profit of multimodal RGB-D data to tackle the gesture representation step. The BoVDW is empowered by the combination of both RGB images and a new depth descriptor which takes into account the distribution of normal vectors with respect to the camera position, as well as the rotation with respect to the roll axis of the camera. Next, we propose the definition of an extension of DTW method to a probability-based framework in order to perform temporal gesture segmentation. In order to evaluate the presented approach, we compare the performances achieved with state-of-the-art RGB and depth feature descriptors separately, and combine them in a late fusion form. All these experiments are performed in the proposed framework using the public dataset provided by the ChaLearn Gesture Challenge[1]. Results of the proposed BoVDW method show better performance using late fusion in comparison to early fusion and standard BoVW model. Moreover, our BoVDW approach outperforms the baseline methodology provided by the ChaLearn Gesture Recognition Challenge 2012. In the same way, the results obtained with the proposed PDTW outperform the ones from the classical DTW approach.

As pointed out above, we address the problem of gesture recognition, with the constraint that an idle or reference gesture is performed between gestures. The main reason for such constraint is that in many real-world settings there always exists an idle gesture between movements rather than a continuous flux of gestures. Some examples are sports like tennis, swordplay, boxing, martial arts, or choreographic sports. However, the existence of an idle gesture is not only related to sports, some other daily tasks like cooking or dancing contain idle gestures in certain situations as well. Moreover, the proposed system can be extended to be applied to other gesture recognition domains without the need of modelling idle gestures, but any other kind of gesture categories.

In this sense, our approach consists of two steps: *a temporal gesture segmentation* step (the detection of the idle gesture), and *the gesture classification* step. The former one aims to provide a temporal segmentation of gestures. To perform such temporal segmentation, a novel probabilistic-based DTW models the variability of the idle gesture by learning a GMM on the features of the idle gesture category. Once the gestures have been segmented, the latter step is gesture classification. Segmented gestures are represented and classified by means of

[1] http://gesture.chalearn.org/

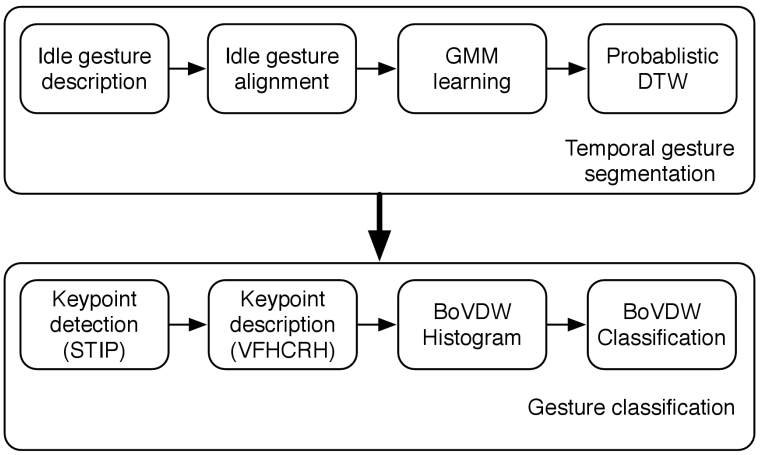


Fig. 4.1 General pipeline of the proposed approach.

a BoVDW method, which integrates in a late fusion form the information of both RGB and Depth images.

The global pipeline of the approach is depicted in Figure 4.1. The proposal is divided in two blocks, the temporal gesture segmentation step and the gesture classification step, which are detailed in next sections. Next, the probability-based DTW for gesture segmentation is introduced in Section 4.1.1, and the BoVDW model in Section 4.1.2. Experimental results and their analysis are presented in Section 4.2. Finally, Section 4.3 presents some conclusions of the section for this chapter.

4.1.1 Gesture Segmentation: Probability-based DTW

The original DTW is introduced in this section, as well as its common extension to detect a certain sequence given an indefinite data stream. In the following subsections, DTW is extended in order to align patterns taking into account the Probability Density Function (PDF) of each element of the sequence by means of a Gaussian Mixture Model (GMM). A flowchart of the whole methodology is shown in Figure 4.2.

Dynamic Time Warping

The original DTW algorithm was defined to match temporal distortions between two models, finding an alignment/warping path between two time series: an input model $Q = \{q_1, .., q_n\}$ and a certain sequence $O = \{o_1, .., o_m\}$. In our particular case, the time series Q and O are video sequences, where each q_j and o_i will be feature vectors describing the $j-$th and $i-$th frame respectively. In this sense, Q will be an input video sequence and O will be the gesture

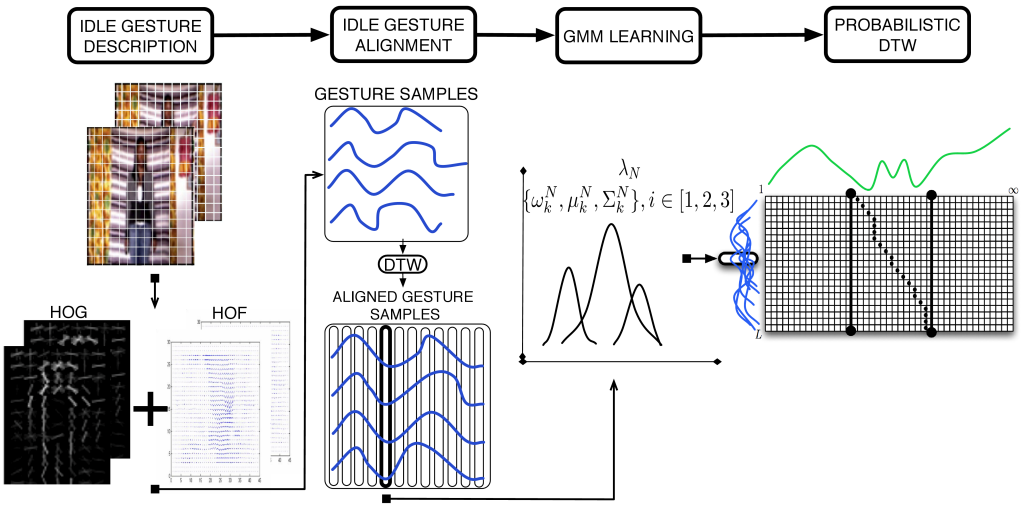


Fig. 4.2 Flowchart of the Probabilistic DTW gesture segmentation methodology.

we are aiming to detect. Generally, in order to align these two sequences, a $M_{m\times n}$ matrix is designed, where position (i,j) of the matrix contains the alignment cost between o_i and q_j. Then, a warping path of length τ is defined as a set of contiguous matrix elements, defining a mapping between O and Q: $\Omega = \{v_1, v_2, ..., v_\tau\}$, where v_p indexes a position in the cost matrix M. This warping path is typically subject to several constraints:

- *Boundary conditions:* $v_1 = (1,1)$ and $v_\tau = (m,n)$.

- *Continuity and monotonicity:* Given $v_{\tau'-1} = (a',b')$, $v_{\tau'} = (a,b)$, then $a - a' \leq 1$ and $b - b' \leq 1$. This condition forces the points in the cost matrix with the warping path Ω to be monotonically spaced in time.

Interest is focused on the final warping path that, satisfying these conditions, minimizes the warping cost,

$$DTW(M) = \min_{\Omega}\left\{\frac{M(v_\tau)}{\tau}\right\}, \tag{4.1}$$

where τ compensates the different lengths of the warping paths at each time t. This path can be found very efficiently using dynamic programming. The cost at a certain position $M(i,j)$ can be defined as the composition of the Euclidean distance $d(i,j)$ between the feature vectors o_i and q_j of the two time series, and the minimum cost of the adjacent elements of the cost matrix up to that position, as

$$M(i,j) = d(i,j) + \min\{M(i-1,j-1), M(i-1,j), M(i,j-1)\}. \tag{4.2}$$

However, given the streaming nature of our problem, the input video sequence Q has no definite length (it may be an infinite video sequence) and may contain several occurrences of the gesture sequence O. In this sense, the system considers that there is correspondence between the current block $\flat$ in Q and the gesture when the following condition is satisfied: $M(m,\flat) < \theta, \flat \in [1,..,\infty]$, for a given cost threshold θ. At this point, if $M(m,\flat) < \theta$, $\flat$ is considered a possible end of a gesture sequence O.

Once detected a possible end of the gesture sequence, the warping path Ω can be found through backtracking the minimum cost path from $M(m,\flat)$ to $M(0,g)$, being g the instant of time in Q where the detected gesture begins. Each path of green/orange cells in Figure 5.2 represents each unique block[2] $\flat$. Note that $d(i,j)$ is the cost function which measures the difference among descriptors o_i and q_j, which in standard DTW is defined as the Euclidean distance between o_i and q_j. An example of a begin-end gesture recognition together with the warping path estimation is shown in Figure 4.2 (last 2 steps: GMM learning and Probabilistic DTW).

Handling variance with Probability-based DTW

Consider a training set of N sequences, $S = \{S_1, S_2, \ldots, S_N\}$, that is, N gesture samples belonging to the same gesture category. Then, each gesture sequence $S_g = \{s_1^g, \ldots, s_{L_g}^g\}$, (a gesture sample) is composed by a feature vector [3] for each frame t, denoted as s_t^g, where L_g is the length in frames of sequence S_g. In order to avoid temporal deformations of the gesture samples in S, all sequences are aligned with the median length sequence using the classical DTW with Euclidean distance. Let us assume that sequences are ordered according to their length, so that $L_{g-1} \leq L_g \leq L_{g+1}, \forall g \in [2,..,N-1]$, then, the median length sequence is $\bar{S} = S_{\lceil \frac{N}{2} \rceil}$.

It is worth noting that this step of alignment by using DTW has no relation to the actual gesture recognition, as it is consider a pre-processing step to obtain a set of gesture samples with few temporal deformations and a matching length.

Finally, after this alignment process, all sequences have length $L_{\lceil \frac{N}{2} \rceil}$. The set of warped sequences is defined as $\tilde{S} = \{\tilde{S}_1, \tilde{S}_2, \ldots, \tilde{S}_N\}$ (See Figure 4.3(b)). Once all samples are aligned, the N feature vectors corresponding to each sequence element at a certain frame t, denoted as $\widetilde{F}_t$, are modelled by means of a $G-$component Gaussian Mixture Model (GMM) $\Lambda_t = \{\alpha_k^t, \mu_k^t, \Sigma_k^t\}, \quad k = 1, \ldots, G$, where α_k^t is the mixing value, and μ_k^t and Σ_k^t are the

[2] About the nomenclature, a feature vector x in Chapter 5.1 refers to a sequence having the same properties as Q. In that chapter 5.1, we use a set of thresholds $\Theta = \{\theta_1, \theta_2, ..., \theta_T\}$ to obtain sets of blocks, so that the minimum cost paths $\mho$ shown in Figure 5.2 are computed by means of backtracking the minimum costs for the whole sequence, whose paths are updated taking into account all the blocks found along the sequence.

[3] HOG/HOF descriptors in our particular case, see Sec. 4.2.2 for further details.

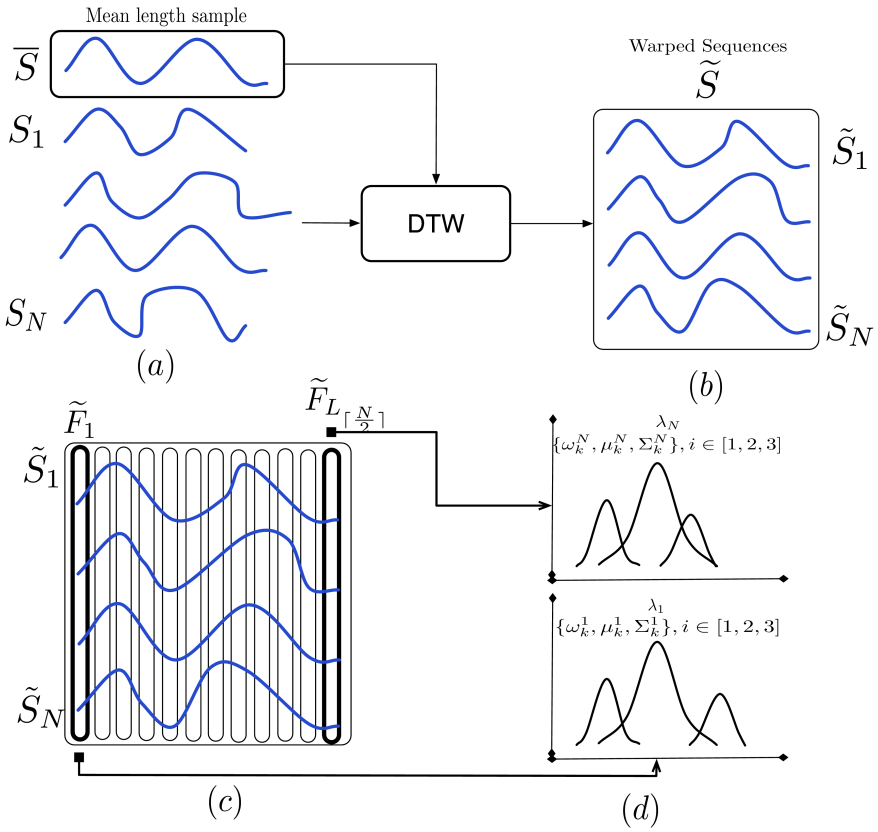


Fig. 4.3 (a) Different sequences of a certain gesture category and the median length sequence. (b) Alignment of all sequences with the median length sequence by means of Euclidean DTW. (c) Warped sequences set $\tilde{S}$ from which each set of t-th elements among all sequences are modelled. (d) Gaussian Mixture Model learning with 3 components.

parameters of each of the G Gaussian models in the mixture. As a result, each one of the GMMs that model each $\widetilde{F}_t$ is defined as follows:

$$p(\widetilde{F}_t) = \sum_{k=1}^{G} \alpha_k^t \cdot e^{-\frac{1}{2}(x-\mu_k^t)^T \cdot (\Sigma_k^t)^{-1} \cdot (x-\mu_k^t)}. \tag{4.3}$$

The resulting model is composed by the set of GMMs that model each set $\widetilde{F}_t$ among all warped sequences of a certain gesture class. An example of the process is shown in Figure 4.3.

Distance measures

In the classical DTW, a pattern and a sequence are aligned using a distance metric, such as the Euclidean distance. However, since our gesture samples are modelled by means of

probabilistic models, in order to use the principles of DTW, the distance must be redefined. In thise sense, a soft-distance based on the probability of a point x belonging to each one of the G components in the GMM is considered, i.e. the posterior probability of x is obtained according to Eq. (4.3). Therefore, since $\sum_{k=1}^{G} \alpha_k^t = 1$, the probability of a element $q_j \in Q$ belonging to the whole GMM Λ_t can be computed as:

$$P(q_j, \Lambda_t) = \sum_{k=1}^{G} \alpha_k^t \cdot P(q_j)_k, \tag{4.4}$$

$$P(q_j)_k = e^{-\frac{1}{2}(q_j - \mu_k^t)^T \cdot (\Sigma_k^t)^{-1} \cdot (q_j - \mu_k^t)}, \tag{4.5}$$

which is the sum of the weighted probability of each component. Nevertheless, an additional step is required since the standard DTW algorithm is conceived for distances instead of similarity measures. In this sense, a soft-distance based measure of the probability is used, which is defined as:

$$D(q_j, \Lambda_t) = \exp^{-P(q_j, \Lambda_t)}. \tag{4.6}$$

In conclusion, possible temporal deformations of different samples of the same gesture category are taken into account by aligning the set of N gesture samples with the median length sequence. In addition, by modelling with a GMM each set of feature vectors which compose the resulting warped sequences, we obtain a methodology for gesture detection that is able to deal with multiple deformations in gestures both temporal (which are modelled by the DTW alignment), or descriptive (which are learned by the GMM modelling). The algorithm that summarizes the use of the probability-based DTW to detect start-end of gesture categories is shown in Table 4.1. Figure 4.6 illustrates the application of the algorithm in a toy problem.

4.1.2 Gesture Representation: BoVDW

In this section, the BoVDW approach for Human Gesture Representation is introduced. Figure 4.4 contains a conceptual scheme of the approach. In this figure, it is shown that the information from RGB and Depth images is merged, while circles representing the spatio-temporal interest points are described by means of the proposed novel VFHCRH (Viewpoint Feature Histogram and Camera Roll Histogram) descriptor.

Table 4.1 Probability-based DTW algorithm.

Input: A set of GMM models $\Lambda = \{\Lambda_1, .., \Lambda_m\}$ corresponding to a gesture category, a threshold value θ, and the streaming sequence $Q = \{q_1, .., q_\infty\}$. Cost matrix $M_{m \times \infty}$ is defined, where $\mathcal{N}(x), x = (i,t)$ is the set of three upper-left neighbor locations of x in M.
Output: Warping path W of the detected gesture, if any.
// Initialization
for $i = 1 : m$ **do**
  for $j = 1 : \infty$ **do**
    $M(i, j) = \infty$
end end
for $j = 1 : \infty$ **do**
  $M(0, j) = 0$
end
for $j = 0 : \infty$ **do**
  for $i = 1 : m$ **do**
    $x = (i, j)$
    $M(x) = D(q_j, \Lambda_i) + \min_{x' \in \mathcal{N}(x)} M(x')$
  end
  if $M(m, j) < \theta$ **then**
    $\Omega = \{\arg\min_{x' \in \mathcal{N}(x)} M(x')\}$
    return
  end
end

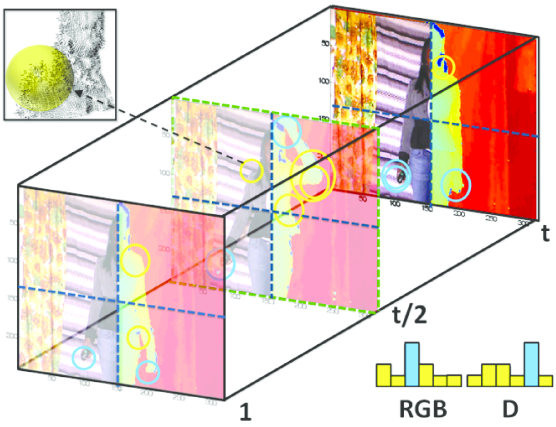


Fig. 4.4 BoVDW approach in a Human Gesture Recognition scenario. Interest points in RGB and depth images are depicted as circles. Circles indicate the assignment to a visual word in the shown histogram – computed over one spatio-temporal bin. Limits of the bins from the spatio-temporal pyramids decomposition are represented by dashed lines in blue and green, respectively. A detailed view of the normals of the depth image is shown in the upper-left corner.

Keypoint detection

The first step of BoW-based models consists of selecting a set of points in the image/video with relevant properties. In order to reduce the amount of points in a dense spatio-temporal sampling, the Spatio-Temporal Interest Point (STIP) detector [83] is used, which is an exten-

sion of the well-known Harris detector in the temporal dimension. The STIP detector firstly computes the second-moment 3×3 matrix η of first order spatial and temporal derivatives. Finally, the detector searches regions in the image with significant eigenvalues λ_1,λ_2,λ_3 of η, combining the determinant and the trace of η,

$$H = |\eta| - K \cdot Tr(\eta)^3, \tag{4.7}$$

where $|.|$ corresponds to the determinant, $Tr(\cdot)$ computes the trace, and K stands for a relative importance constant factor. As multi-modal RGB-D data is employed, the STIP detector is applied separately on the RGB and Depth volumes, so two sets of interest points S_{RGB} and S_D are obtained.

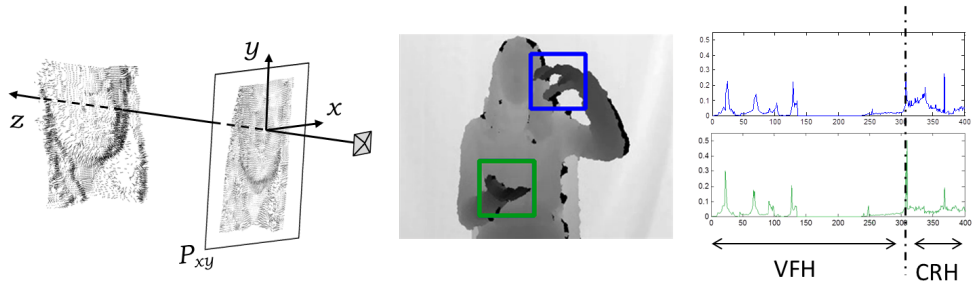
Keypoint description

In this step, the interest points detected in the previous step should be described. On one hand, state-of-the-art RGB descriptors are computed for S_{RGB}, including Histogram of Gradients (HOG) [30], Histogram of Optical Flow (HOF), and their concatenation HOG/HOF [84]. On the other hand, a new descriptor VFHCRH is introduced for S_D, as detailed below.

VFHCRH

The recently proposed Point Feature Histogram (PFH) and Fast Point Feature Histogram (FPFH) descriptors [137] represent each instance in the 3-D cloud of points with a histogram encoding the distribution of the mean curvature around it. Both PFH and FPFH provide $\mathscr{P}$6 DOF (Degrees of Freedom) pose invariant histograms, being $\mathscr{P}$ the number of points in the cloud. Following their principles, Viewpoint Feature Histogram (VFH)[136] describes each cloud of points with one descriptor of 308 bins, variant to object rotation around pitch and yaw axis. However, VFH is invariant to rotation about the roll axis of the camera. In contrast, Clustered Viewpoint Feature Histogram (CVFH) [5] describes each cloud of points using a different number of descriptors r, where r is the number of stable regions found on the cloud. Each stable region is described using a non-normalized VFH histogram and a Camera's Roll Histogram (CRH), and the final object description includes all region descriptors. CRH is computed by projecting the normal of the point cloud $\rho^{(i)}$ for the i-th point onto a plane P_{xy} that is orthogonal to the viewing axis z, the vector between the camera center and the centroid of the cloud, under orthographic projection,

$$\rho_{xy}^{(i)} = ||\rho^{(i)}|| \cdot \sin(\phi), \tag{4.8}$$



(a) (b)

Fig. 4.5 (a) Point cloud of a face and the projection of its normal vectors onto the plane P_{xy}, orthogonal to the viewing axis z. (b) VFHCRH descriptor: Concatenation of VFH and CRH histograms resulting in 400 total bins

where ϕ is the angle between the normal $\rho^{(i)}$ and the viewing axis. Finally, the histogram encodes the frequencies of the projected angle ψ between $\rho_{xy}^{(i)}$ and y-axis, the vertical vector of the camera plane (see Fig. 4.5(a)).

In order to avoid descriptors of arbitrary lengths for different point clouds, the whole cloud is described using VFH. In addition, a 92 bins CRH is computed for encoding 6*DOF* information. The concatenation of both histograms results in the proposed VFHCRH descriptor of 400 bins shown in Fig. 4.5(b). Note how the first 308 bins of the concatenated feature vector correspond to the VFH, that encode the normals of the point cloud. Finally, the remaining bins corresponding to the CRH descriptor, encode the information of the relative orientation of the point cloud to the camera.

BoVDW histogram

Once all the detected points have been described, the vocabulary of V visual/depth words is designed by applying a clustering method over all the descriptors. Hence, the clustering method –k-means in our case– defines the words from which a query video sequence will be represented, shaped like a histogram h that counts the occurrences of each word. Additionally, in order to introduce geometrical and temporal information, spatio-temporal pyramids are applied. Basically, spatio-temporal pyramids consist of dividing the video volume in b_u, b_υ, and b_ϖ bins along the u, υ, and ϖ dimensions of the volume, respectively. Then, $b_u \times b_\upsilon \times b_\varpi$ separate histograms are computed with the points lying in each one of these bins, and they are concatenated jointly with the general histogram computed using all points.

These histograms define the model for a certain class of the problem –in our case, a certain gesture. Since multi-modal data is considered, different vocabularies are defined for

the RGB-based descriptors and the depth-based ones, and the corresponding histograms, h^{RGB} and h^{D}, are obtained. Finally, the information given by the different modalities is merged in the next and final classification step, hence using *late fusion*.

BoVDW-based classification

The final step of the BoVDW approach consists of predicting the class of the query video. For that, any kind of multi-class supervised learning technique could be used. In our case, a simple k-Nearest Neighbour classification is used, computing the complementary of the histogram intersection as a distance,

$$d^F = 1 - \sum_i \min(h^F_{model}(i), h^F_{query}(i)), \tag{4.9}$$

where $F \in \{RGB, D\}$. Finally, in order to merge the histograms h^{RGB} and h^{D}, the distances d^{RGB} and d^{D} are computed separately, as well as the weighted sum,

$$d_{hist} = (1-\varsigma)d^{RGB} + \varsigma d^{D}, \tag{4.10}$$

to perform late fusion, where ς is a weighting factor.

4.2 Experiments for PDTW and BoVDW

To better understand the experiments, firstly the data, methods, and evaluation measurements are discussed.

4.2.1 Data

Data source used is the ChaLearn [63] dataset, provided by the CVPR2011 Workshop's challenge on Human Gesture Recognition. The dataset consists of 50,000 gestures each one portraying a single user in front of a fixed camera. The images are captured by the Kinect device providing both RGB and depth images. A subset of the whole dataset has been considered, formed by 20 development batches with a manually tagged gesture segmentation, which is used to obtain the idle gestures. Each batch includes 100 recorded gestures grouped in sequences of 1 to 5 gestures performed by the same user. The gestures from each batch are drawn from a different lexicon of 8 to 15 unique gestures and just one training sample per gesture is provided. These lexicons are categorized in nine classes, including: (1) body language gestures (scratching your head, crossing your arms, etc.), (2) gesticulations

performed to accompany speech, (3) illustrators (like Italian gestures), (4) emblems (like Indian Mudras), (5) signs (from sign languages for the deaf), (6) signals (diving signals, mashalling signals to guide machinery or vehicle, etc.), (7) actions (like drinking or writing), (8) pantomimes (gestures made to mimic actions), and (9) dance postures.

For each sequence, the actor performs an idle gesture between each gesture to classify. These idle gestures are used to provide the temporal segmentation (further details are shown in the next section). For this dataset, background subtraction was performed based on depth maps, and a 10×10 grid approach was defined to extract HOG+HOF feature descriptors per cell, which are finally concatenated in a full image (posture) descriptor. Using this dataset, the recognition of the idle gesture pattern will be tested, using 100 samples of the pattern in a ten-fold validation procedure.

4.2.2 Methods and Evaluation

The experiments are presented in two different sections. The first section considers the temporal segmentation experiment while the second section aims the gesture classification experiments.

Temporal Segmentation Experiments

In order to provide with quantitative measures of the temporal segmentation procedure, we first describe the subset of the data used and the feature extraction.

- *Data and Feature extraction*
  For the temporal segmentation experiments we used the 20 development batches provided by the challenge organizers. These batches contain a manual labelling of gesture start and end points. Each batch includes 100 recorded gestures, grouped in sequences of 1 to 5 gestures performed by the same user. For each sequence the actor performs an idle gesture between each gesture of the gestures drawn from lexicons. Finally, this means that we have a set of approximately 1800 idle gestures.

  Each video sequence of each batch was described using a 20×20 grid approach. For each patch in the grid we obtain a 208 feature vector consisting of HOG (128 dimensions) and HOF (80 dimensions) descriptors which are finally concatenated in a full image (posture descriptor). Due to the huge dimensionality of the descriptor of a single frame (83200 dimensions), we utilized a Random Projection to reduce dimensionality to 150 dimensions.

- *Experimental Settings*
  For both of the DTW approaches the cost-threshold value θ is estimated in advance using

Table 4.2 *Overlapping* and *accuracy* results.

	Overlap.	Acc.
Probability-based DTW	**0.3908± 0.0211**	**0.6781 ± 0.0239**
Euclidean DTW	0.3003 ± 0.0302	0.6043 ± 0.0321
HMM	0.2851 ± 0.0432	0.5328 ± 0.0519

ten-fold cross-validation strategy on the set of 1800 idle gesture samples. This involves using 180 idle gestures as the validation data, and the remaining observations as the training data. This is repeated such that each observation in the sample is used once as the validation data. Finally, the threshold value θ chosen is the one associated with the largest overlapping performance. For the probabilistic DTW approach, each GMM was fit with 4 components. The value of G was obtained using a ten-fold cross-validation procedure on the set of 1800 idle gestures as well. In this sense, the cross-validation procedure for the probability-based DTW is a double loop (optimizing on the number of GMM components G, and then, on the cost-threshold θ). On the other hand, we used the Baum-Welch algorithm for training an Hidden Markov Model (HMM), and 3 states were experimentally set for the idle gesture, using a vocabulary of 60 symbols computed using k-means over the training data features. Final recognition is performed with temporal sliding windows of different wide sizes, based on the idle gesture samples length variability.

- *Methods, Measurements and Results*
  Our probability-based DTW approach using the proposed distance D shown in Eq. (4.6) is compared to the usual DTW algorithm and the HMM approach. The evaluation measurements presented are *overlapping* and *accuracy* of the recognition for the idle gesture, considering that a gesture is correctly detected if overlapping in the idle gesture subsequence is greater than 60% (the standard overlapping value).

  The results of our proposal, HMM and the classical DTW algorithm are shown in Table 4.2. It can be seen how the proposed probability-based DTW outperforms the usual DTW and HMM algorithms in both experiments. Moreover, confidence intervals of DTW and HMM do not intersect with the probability-based DTW in any case. From this results it can be concluded that performing dynamic programming increases the generalization capability of the HMM approach, as well as a model defined by a set of GMMs outperforms the classical DTW on RGB-Depth data without increasing the computational complexity of the method. Figure 4.6 shows qualitative results from two sample video sequences.

Fig. 4.6 Examples of idle gesture detection on the Chalearn dataset using the probability-based DTW approach. The line below each pair of depth and RGB images represents the detection of a idle gesture (step up: beginning of idle gesture, step down: end)

.

BoVDW Classification Experiments

In all the experiments shown in this section, the vocabulary size was set to $V = 200$ words for both RGB and depth cases. For the spatio-temporal pyramids, the volume was divided in $2 \times 2 \times 2$ bins (resulting in a final histogram of 1800 bins). Since the nature of our application problem is one-shot learning (only one training sample is available for each class), a simple Nearest Neighbor classification is employed. Finally, for the late fusion, the weight $\varsigma = 0.8$ was empirically set, by testing the performance of our method in a small subset of development batches from the dataset.

For the evaluation of the methods, in the context of Human Gesture Recognition, the Levenshtein distance or edit distance was considered. This edit distance between two strings is defined as the minimum number of operations (insertions, substitutions or deletions) needed to transform one string into the other. In our case, strings contain gesture labels detected in a video sequence. For all the comparison, the mean Levenshtein distance (MLD) was computed over all sequences and batches.

Table 4.3 shows a comparison between different state-of-the-art RGB and depth descriptors (including our proposed VFHCRH), using our BoVDW approach. Moreover, we compare our BoVDW framework with the baseline methodology provided by the ChaLearn 2012 Gesture Recognition challenge [41]. This baseline first computes differences of contiguous frames, which encode movement information. After that, these difference images are divided into cells forming a grid, each one containing the sum of movement information among it. These 2D grids are then transformed then into vectors, one for each difference image. Moreover, the model for a gesture is computed via Principal Component Analysis (PCA), using all the vectors belonging to that gesture. The eigenvectors are just computed and stored, so when a new sequence arrives, its movement signature first is computed, and then projected and reconstructed using the different PCA models from each gesture. Finally, the classification is performed by choosing the gesture class with lower reconstruction error. This baseline obtains a MLD of 0.5096. The bar plot in Figure 4.8 shows the results in all the 20 development batches separately.

When using our BoVDW approach, in the case of RGB descriptors, HOF alone performs the worst. In contrast, the early concatenation of HOF to HOG descriptor outperforms the simple HOG. Thus, HOF contributes adding discriminative information to HOG. In a similar way, looking at the depth descriptors, it can be seen how the concatenation of the CRH to the VFH descriptor clearly improves the performance compared to the simpler VFH. When using late fusion in order to merge information from the best RGB and depth descriptors (HOGHOF and VFHCRH, respectively), a value of 0.2714 for MLD is achieved. Figure 4.7 shows the confusion matrices of the gesture recognition results with this late

Table 4.3 Mean Levenshtein distance for RGB and depth descriptors.

RGB desc.	MLD	Depth desc.	MLD
HOG	0.3452	VFH	0.4021
HOF	0.4144	VFHCRH	**0.3064**
HOGHOF	**0.3314**		

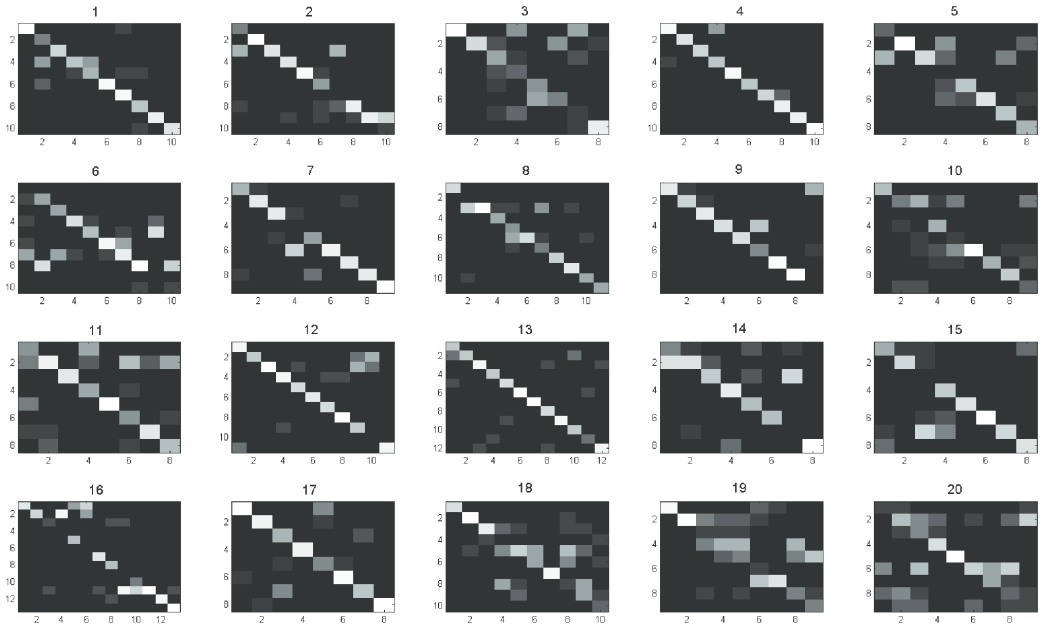


Fig. 4.7 Confusion matrices for gesture recognition in each one of the 20 development batches.

fusion configuration. In general, the confusion matrices follow an almost diagonal shape, indicating that the majority of the gestures are well classified. However, the results of batches 3, 16, 18, 19 are significantly worse, possibly due to the static characteristics of the gestures in these batches. Furthermore, late fusion was also applied in a 3-fold way, merging HOG, HOF, and VFHCRH descriptors separately. In this case the weight ς was assigned to HOG and VFHCRH descriptors (and $1-\varsigma$ to HOF), improving the MLD to 0.2662. From this result it can be concluded that HOGHOF late fusion performs better than HOGHOF early fusion.

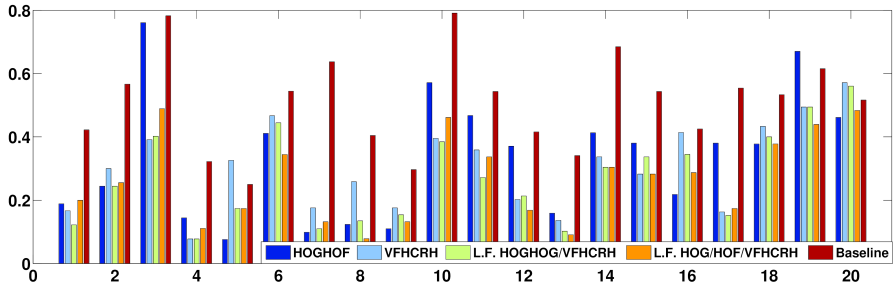


Fig. 4.8 Performance of the best RGB and depth descriptors separately, as well as the 2-fold and 3-fold late fusion of them. Results obtained by the baseline from the ChaLearn challenge are also shown. x-axis represent different batches and y-axis represents the MLD of each batch.

4

4.3 Conclusion

In this chapter, the BoVDW approach for Human Gesture Recognition has been presented using multi-modal RGB-D images. A new depth descriptor VFHCRH has been proposed, which outperforms VFH. Moreover, the effect of the late fusion has been analysed for the combination of RGB and depth descriptors in the BoVDW, obtaining better performance in comparison to early fusion. In addition, a probabilistic-based DTW has been proposed to asses the temporal segmentation of gestures, where different samples of the same gesture category are used to build a Gaussian-based probabilistic model of the gesture in which possible deformations are implicitly encoded. In addition, to embed these models into the DTW framework, a soft-distance based on the posterior probability of the GMM was defined. In conclusion, a novel methodology for gesture detection has been presented, which is able to deal with multiple deformations in data.

Chapter 5

Evolving Dynamic Representations

This chapter introduces a novel approach for evolving representations based on dynamic programming and generative models. The capabilities of the presented evolutionary framework is demonstrated in several well-known datasets for the task of action recognition.

5.1 Gesture and Action Recognition by Evolved Dynamic Subgestures

Gesture and human action recognition are two widely studied topics in computer vision. Great advances have been reported in the last few years [3], mainly boosted by the release of the Kinect sensor [145]. Most of existing recognition methods learn gesture/action models that attempt to capture and recognize whole gestures (*i.e.*, an holistic approach). Classical approaches under this scheme are those based on dynamic time warping (DTW) [18] and hidden Markov models (HMM) [151, 173].

Although the previous methods have obtained high performance in several domains, recent research is moving towards approaches that model the problem in terms of gesture primitives (subgestures) [88, 99, 100, 122, 170]. The underlying assumption of this type of methods is that whole gestures are composed by primitives (that can be shared or not among gestures from different categories), and the hypothesis is that learning with primitives leads to better recognition performance. Whereas the subgesture-based techniques have proved to be successful, it remains open the question on how to define/learn subgestures and, more importantly, how to perform inference using subgesture models.

This section describes a novel approach for human action and gesture recognition based on subgesture modeling. Unlike other primitive modeling approaches, our proposal learns subgestures by searching for temporal patterns that improve recognition performance when used to represent and classify complex gestures and actions. An evolutionary algorithm is implemented for this purpose, with adhoc variation operators suitable for learning primitive recognizers of actions/gestures. This algorithm takes as reference two standard methods for learning from sequential data: DTW and HMMs. Besides learning the primitives from scratch, it determines the inference procedures for DTW and HMM when using subgestures. The proposed framework is evaluated in MSRDaily3D and MSRAction3D datasets, outperforming state of the art results.

5.2 Training Dynamic Subgestures

This section describes the methodology to automatically learn gesture primitives (hereinafter referred to as subgestures). Consider a training dataset $X^T = \{x_1^T, x_2^T, ..., x_n^T\}$, where each $x^T \in X^T$ is a *sequence* example of a gesture. Similarly, consider a validation dataset $X^V = \{x_1^V, x_2^V, ..., x_m^V\}$ of gesture sequence examples. Both X^T and X^V are subsets of a dataset, whose sequence examples belong to different classes $C = \{c_1, c_2, ..., c_g\}$. Our goal is to find a subgesture set $S = \{s_1, s_2, ..., s_k\}$ from X^T, being s_i a sequence representation of the

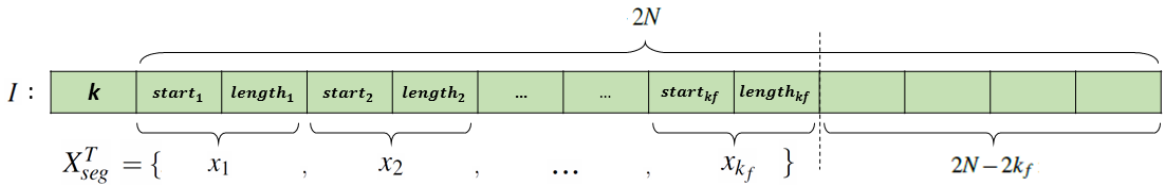


Fig. 5.1 Representation of an individual I formed by $1+2N$ genes: k for the number of subgestures, X^T_{seg} the set of k_f pair-wise generated segments, and $2N-2k_f$ empty genes.

subgesture i, that maximizes the recognition performance of gestures in X^V, given a particular gesture recognition method (see Algorithm[1] 1).

Data: Population P; Training data X^T; Validation data X^V
Result: Models of k subgestures S for each individual and its score
Current generation:
foreach *new unique valid I in the population P* **do**
  $k, segments \leftarrow decode(I)$;
  $X^T_{seg} \leftarrow getDataPartitions(X^T, segments)$;
  $S \leftarrow k\text{-}meansDTW(X^T_{seg}, k)$; `// Section 5.2.2`
  if *use dynamic programming* **then**
    $R \leftarrow getResizedClassModels(X^T, S)$; $\mathbf{D} \leftarrow getDissimilarities(S)$;[1]
    $M \leftarrow getUpdatedCosts(R, S)$; $D^V \leftarrow getUpdatedCosts(X^V, S)$;[1] `// Figure 5.2`
    $\omega \leftarrow addParamsToStruct(M, \mathbf{D})$;
  else if *use generative model* **then**
    $D^T \leftarrow getUpdatedCosts(X^T, S)$; $D^V \leftarrow getUpdatedCosts(X^V, S)$;[1] `// Figure 5.2`
    $M \leftarrow learnGM(D^T, \omega)$;
    $\omega \leftarrow addParamsToStruct(M)$;
  end
  $s, \omega^* \leftarrow g(D^V, \omega)$; `// Section 5.2.3`
end

Algorithm 1: Pseudocode for learning Subgesture Models at each generation

5.2.1 Evolutionary Optimization

Let $P = \{I_1, I_2, ..., I_l\}$ be a population of l individuals, each one composed of $1+2N$ genes. The first *gene* refers to the number k of subgestures and the remainder $2N$ genes refer to pairs of start-length segments from X^T. Initially, there is a probability p_s of generating each pair-wise *segment*. Those candidate segments are generated via a *random selection* over the whole continuous sequence X^T (*i.e.* the concatenation of all training sequences), ensuring that the length of each possible segment is within $[n_{min}, n_{max}]$ frames. Thus, each individual I has $k_f \leq N$ pair-wise generated segments. Finally, the value of the first gene is

[1] The lines having more than one instruction in Algorithm 1 can be computed in parallel.

randomly chosen between the range $[k_0, k_f]$, so that $k_0 \leq k \leq k_f$. It means that the number of k allowed *clusters is set depending on the generated segments.* The training procedure ignores the remaining $2N - 2k_f$ empty segments in the fitness function. Figure 5.1 shows the representation of an individual.

Fitness function

The goal of the proposed genetic algorithm is to maximize the *score* given by the evaluation function, described in Section 5.2.3. It consists of obtaining a measure of performance for the learned *models*, expressed in terms of subgestures, over validation sequences in the classification task. Section 5.2.2 provides details of the *aligned* temporal clustering method developed to obtain subgestures. Once subgestures are computed, we provide either dynamic programming or generative model approaches to learn and evaluate the subgesture models.

Dynamic Programming: As presented in Algorithm 1, we obtain a model for each class represented in subgestures. Each subgesture within the set $S = \{s_1, s_2, ..., s_k\}$ is the centroid sequence obtained from the $k-means DTW$ algorithm. Therefore, we design each class model $m_c \in M$, where M is the set of class models, by 1) computing $r_c \in R$ as the mean of all resized training samples of each class, where R is the set of all resized training samples, and 2) representing r_c in subgestures. This procedure is done by means of a backward loop over the DTW warping paths (see Figure 5.2). On the other hand, we compute the dissimiliarity matrix as:

$$\mathbf{D} = \mathbf{W} + \mathbf{W}^{\intercal}, \quad s.t. \quad \mathbf{W} = \frac{1}{\gamma} \begin{bmatrix} w_{11} & w_{12} & ... & w_{1k} \\ w_{21} & w_{22} & ... & w_{2k} \\ . & . & ... & . \\ w_{k1} & w_{k2} & ... & w_{kk} \end{bmatrix}, \tag{5.1}$$

where $\mathbf{W}$ is a squared matrix obtained from aligning all subgestures among them. This is, to compute each element as the DTW cost by:

$$w_{ij} = DTW(s_i, s_j) = min_{\Omega}\{\sum_{p=1}^{\tau} d_p, \Omega =< v_1, v_2, ..., v_\tau >\}, \tag{5.2}$$

where d_p is the Euclidean distance between feature vectors $s_i^{\dot{x}}$ and $s_j^{\dot{y}}$ given the coordinates $v_p = (\dot{x}, \dot{y})$ of the warping path Ω. Then, each element of the matrix $\mathbf{W}$ is normalized *w.r.t.* the maximum cost value γ of all elements $w_{ij} \in \mathbf{W}$. To express both each class representative sequence r_c and each validation sequence x^V in terms of subgestures, we assign to each frame ι the subgesture identifiers that give the minimum costs, respectively, as:

$$i = \arg\min(\vec{km}^{\iota}) \quad , \quad j = \arg\min(\vec{kt}^{\iota}); \tag{5.3}$$

where $\vec{km}^{\iota}$ and $\vec{kt}^{\iota}$ are vectors of length k subgestures corresponding to the *columns* of the cost matrices KM and KT for the current training and validation sequences, respectively

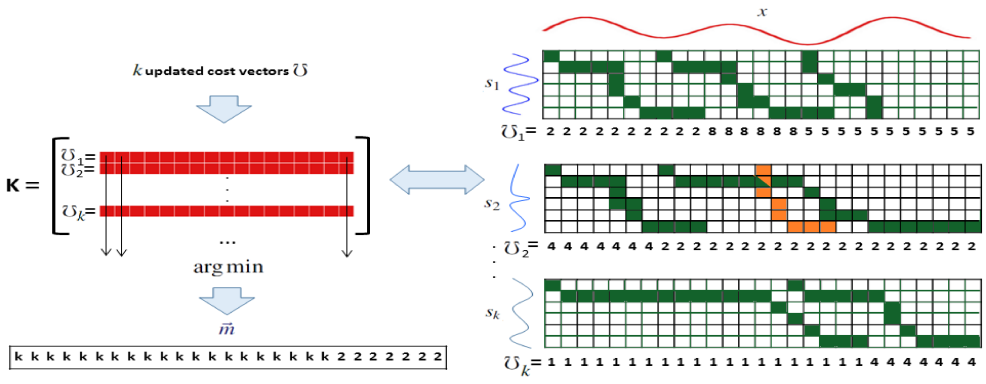


Fig. 5.2 Graphical example of the computation of KM and KT from the backward loop over the DTW warping paths Ω (best seen in color). Input sequence x is aligned to the subgesture sequences from S so as to obtain the k updated cost vectors υ that construct the matrix K. In the different modules of our approach we use the common DTW alignment that initializes the first row and column of the DTW matrix as infinity, which indicates that the warping path goes from the first position until the last position of the DTW matrix. In this current step, however, to compute each υ, we initialize first DTW row to zeros to compute multiples warping paths and perform backward search, starting from the last position of last row and stopping when the path reaches the first row. While computing the backward path, we prioritize the left-steps when the neighbor positions have same cost values, in order to maximize the length of the paths. Finally, we assign to each position of υ the minimum cost values of the paths found that involves that position. We refer to matrix $\mathbf{K}$ as KM and KT when input sequences are from the training and validation (or test) set, respectively. The vector $\vec{m}$ is the final input sequence represented in subgestures, *i.e.* from the arguments obtained in Eq. 5.3. Figure 5.4 shows two real examples of $\vec{m}$, identifying subgestures in real skeleton-based gesture sequences.

(see description in Figure 5.2). Therefore, the set of arguments $i^* = \arg\min(\vec{km})$ and $j^* = \arg\min(\vec{kt})$ are the subgesture identifiers that construct the class models $m_c \in M$ and the validation sequences $d^V \in D^V$ in terms of subgestures. Then, final evaluation is obtained as:

$$DTW(m_c, d^V) = min_{\Omega}\{\sum_{p=1}^{\tau} \mathbf{D}(i_p, j_p), \Omega =< v_1, v_2, ..., v_\tau >\}. \tag{5.4}$$

Note that the expression of Eq. 5.4 takes the same form as Eq. 5.2, but instead of using the Euclidean distance, each distance $\mathbf{D}(i_p, j_p)$ in the warping path considers the similarities among subgestures.

Generative models: Still looking at algorithm 1, our generative model deals with $1D$ discrete sequences. The first step is thus to obtain discrete representations of training and validation sequences. Similarly to the DTW approach and the Figure 5.2, we represent each training and validation sequence in terms of subgestures using Eq. 5.3 so as to construct the discrete sequences D^T and D^V. This is, therefore, how we represent the observations of the HMM from the discrete sequences in D^T and D^V, given the original sequences in X^T and X^V,

respectively. Then, considering D^T as the set of training sequences, we train every HMM for each class so as to learn our generative models.

Genetic operators

We consider standard *selection*, *crossover* and *mutation* operators from [59]. Specifically, we apply these operators to all genes of each individual (*i.e.* k clusters and N segments). Before applying the mutation operator, however, each of the N segments has again a probability p_s either to *add* if it is empty, or to *delete* if it already exists. The *offspring* also requires to meet several constraints that might be violated once we apply these standard genetic operators. To ensure they are met, we apply a *repair* algorithm to fix the new incorrect segments immediately after applying the crossover and mutation operators. Basically, it consists of a brute force criteria that fixes those incorrect segments either by moving them so as to stay within the length of X^T (even though keeping the segment length proportions when they are correct), or by generating new segments within the range $[n_{min}, n_{max}]$ when they are out of bounds. Moreover, we use Eq. 5.5 either to increase k_f and hence generate new segments, or to decrease k, the number of clusters:

$$p(k) = \frac{k - k_0}{k_f - k_0} \Rightarrow \begin{cases} \text{if } p(k) \leq 1 & \text{increase } k_f \text{ segments} \\ \text{Otherwise} & \text{decrease } k \text{ clusters.} \end{cases} \tag{5.5}$$

This procedure ensures, not only that the offspring that pass throughout the next generations are evaluable, but also that we respect the new trends of the genes caused by these genetic operators. The repair function accelerates the convergence of the genetic algorithm.

5.2.2 Aligned Temporal Clustering

Let X^T_{seg} be the set of k_f sequence segments decoded from an individual I and the whole continuous training sequence X^T. Similarly to the classical $k-means$ algorithm, our method groups the X^T_{seg} examples into k clusters. In our setting, however, each example $x_s \in X^T_{seg}$ is a sequence, so that it is a point in the space and time. Therefore, it is convenient to consider an appropriate measure as DTW so as to treat temporal deformations. Thus, in the expectation step we obtain the costs of aligning each sequence to all the *centroids* (initially k random sequences of X^T_{seg}). Then, we assign each sample to the cluster having the minimum cost of the DTW warping path. In the maximization step, first we update the centroids by means of resizing all sequences that belong to the same cluster *w.r.t.* the median length sequence of that cluster. Then, we calculate the new centroid as the mean of all resized sequences for each cluster. The algorithm converges either when the costs of aligning the current centroids

to the ones from the previous iteration are 0, or when it reaches the maximum number of iterations ι. Once the algorithm converges, we assign the set of S subgesture sequences as the final centroids.

5.2.3 Evaluation

The evaluation function computes the mean score of classifying each sequence given the learned model parameters. In training, moreover, we learn the thresholds that provide the maximum score of classifying each class. Then, we use these thresholds in test time as part of the learned model parameters. We learn and test thresholds as follows:

Dynamic Programming. Once we compute the costs of aligning all validation sequences in X^V to the class-models M, we learn a set of class-thresholds $\Theta = \{\theta^{c_1}, \theta^{c_2}, ..., \theta^{c_g}\}$ as those DTW costs that maximize the score per class, being part of the global set of learned model parameters ω^*. These thresholds are used to compute classification rate of test samples represented in subgestures.

Generative models. Once we learn a HMM per class, we compute the probabilities of generating each discrete sequence in D^V, $P(d^v \in D^V | m_c)$, and learn the class-thresholds Θ, included in ω^*. These thresholds are used to compute classification rate of test samples represented in subgestures.

5.3 Experiments for BoSG

5.3.1 Datasets

For the evaluation of the proposed framework we considered two widely used datasets for human action recognition: MSRDaily3D and MSRAction datasets (see Figure 5.3). We evaluate the performance of our methods and compare its results with state of the art techniques that have used the same datasets.

The MSRDaily3D dataset comprises 16 actions associated to daily activities, where there are objects in the background and most actions involve human-object interaction. For comparison with previous work we used this dataset under two settings: cross-validation and half-subject split. The former setting allows us to compare the results of our methods with recent work that has used the same descriptor [70–72, 175]. Under this setting we considered 12 out of the 16 actions and performed 5-fold cross validation (as in [70–72, 175]). For the other setting we considered the 16 categories and used the sample half-training / half-testing subject split (*e.g.*, as in [89, 167, 168]). In either configuration, video sequences were

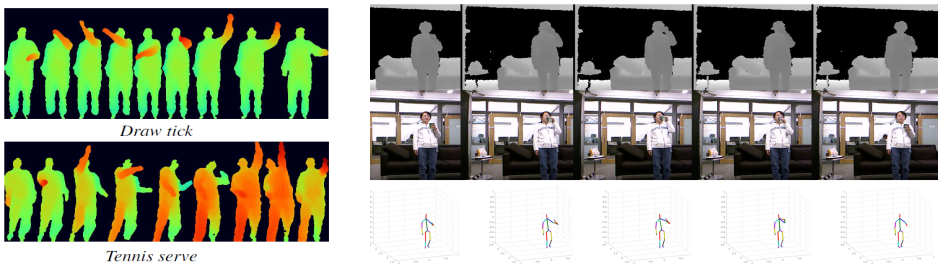


Fig. 5.3 Sample images from the MSRAction3D (left,depth information is available, image taken from [89]) and MSRDaily3D [167] (right, skeleton, RGB-D video available) datasets.

represented with Depth Cuboid Similarity Features (DCSF), the same parameters for the descriptor as in [71, 72, 175] were used.

The MSRAction3D dataset comprises 20 actions, recorded by 10 subjects, where subjects are isolated and no objects in the background are present. Together with the MSRDaily dataset, this is one of the most used datasets for the assessment of human action recognition techniques when using the depth/skeleton information. As before, video sequences were represented with a bag of DCSF descriptors. For this dataset, the standard half-training (subjects 1,3,5,7,9) / half-testing (rest of subjects) split was adopted (see [115] for a complete analysis of results on this dataset).

5.3.2 Setting and metrics

All of the methods were implemented in MATLAB/C++[2], integrating functionalities from the GA optimtool [59] and PMTK3 libraries. The parameters of our method were fixed as follows: $P_s = 0.2$, $n_{min} = 5$ and $n_{max} = 25$ (as in [183]). We set our population length to $l = 20$, with 2 elitist members that pass throughout the next generations.

The $k-meansDTW$ described in section 5.2.2 requires both to resize the segments samples of each cluster in X^T_{seg} and to align them *w.r.t.* the k centroids so as to obtain the new clusters. The computational cost of this step is about $\mathcal{O}(\iota \times k \times n^2)$, where the number of iterations is set by defalt as $\iota = 20$. Hence, we defined $N = 500$ in our experiments to generate the pairs of start-length segments, providing a trade-off between number of segment and computation requirements. Moreover, we defined $k_0 = 3$ to consider a low value for the minimum number of clusters, so that we allow to set k between a large enough range $[k_0, k_f]$ for the $k-meansDTW$ algorithm. Finally, in the evaluation we use $T = 20$ for the range of thresholds to learn Θ, and compute mean accuracy among all test sequences.

[2] **Library publicly available at https://github.com/vponcelo/Subgesture**

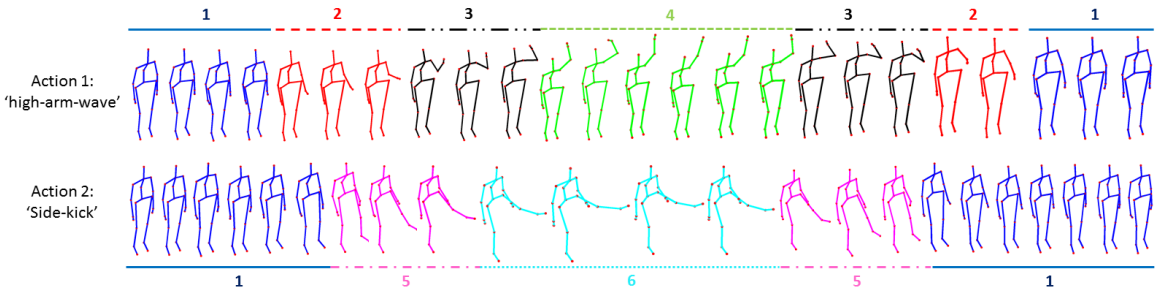


Fig. 5.4 Visual scheme of frame-skeletons grouped into (temporal) subgesture-clusters for the MSRAction3D dataset (best seen in color).

5.3.3 Results

The DTW baseline consists of using the classical DTW with Euclidean distance to classify the test sequences. Thus, instead of learning subgesture models, our baseline models are formed by means of direct resizing all sequence samples of each class *w.r.t.* the median length sequence from that class. On the other hand, in the HMM baseline we split each gesture sequence in 3 parts having the same length to construct the set of sequence segments for learning the subgesture models. The number of clusters k is the half of the total number of resulting segments. To reduce the computational complexity of the HMM baseline we get a reduced number of samples as input to the $k-means DTW$, so that for each class we choose 10 random gestures rather than considering all the training gesture sequences.

Figure 5.4 shows an example of representing two sequences of different actions into subgestures on the MSRAction3D dataset, applying the procedure described in Figure 5.2. The two sequence actions are 'high-arm-wave' and 'side-kick', and the subgestures are those from the last generation that gave the best performance in the evolved DTW version. At the frame level, one can observe that the skeletons that fall into the same cluster are quite similar, though there are some skeletons that are visually similar to those belonging to a different cluster (*e.g.* frame-skeleton 5 in comparison to the frame-skeletons that belong to the cluster 1). At the temporal level, we can observe that the cluster 1, formed by similar segments of different length, is shared among the two different action sequences. The same phenomena happens for the clusters 2, 3 and 5, though these are shared clusters along the same action sequence. This shows the qualitative performance of the $k-means DTW$ algorithm, which provide effective clusters by computing temporal deformations over the input segments of different lengths.

In Table 5.1, we report the mean results of running our genetic algorithm 5 times both to the half-subject split of the MSRAction3D and MSRDaily3D datasets, and to the 5-fold

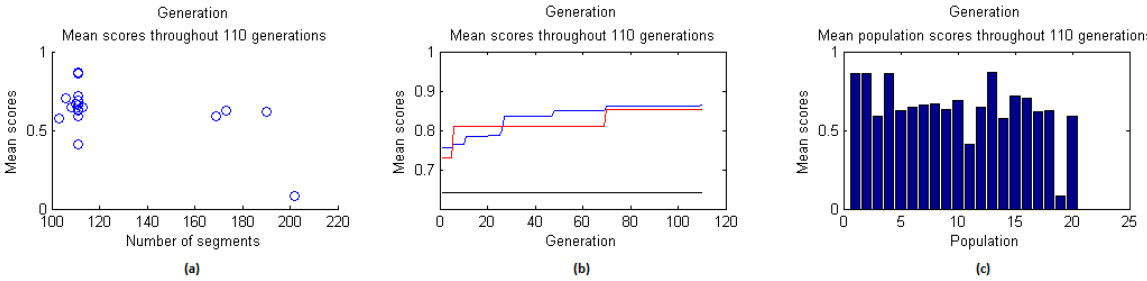


Fig. 5.5 Example of the evolution of the genetic algorithm for the first fold of the MSR-Daily3D dataset. The plot (a) shows the number of segments that belong to each individual and their scores on the last generation. The plot (b) shows the score (accuracy) of the best individuals at each generation so as to see the evolution of the different settings: baseline in validation (black) where there is no evolution, and the evaluation of the models from the best individual of the generation in validation (blue) and test (red). The barplot (c) shows the distribution of scores of each individual of the population on the last generation.

cross-validation of the MSRDaily3D dataset. In all cases, the evolution of the subgesture models learned with the HMM outperforms the state of the art in these datasets, achieving results above the 91% from the *initial generations*. Specially for the MSRAction3D dataset, the improvement of evolving subgesture models with the HMM is the greatest *w.r.t.* the HMM baseline, achieving the best result of the 95%. The evolved DTW version also provides a considerable improvement *w.r.t.* the DTW baselines, outperforming the state of the art on the MSRDaily3D dataset, and achieving comparable performances on the MSRAction3D.

To illustrate the evolution of the genetic algorithm. In the left plot (a) of figure 5.5 one can see a clear trend of the individuals to go towards the number of segments that give the best performance (111). The middle plot (b) shows that from the starting generations the performance both in validation and test are above the baseline. Their performance improve along the generations and keep very similar on the last generations. From the distribution of scores on the right barplot (c), one can observe that all individuals have positive scores and some of them achieve similar values, showing that the repair algorithm using Eq. 5.5 forces the individuals to become valid, speeding up convergence.

5.4 Conclusion

We introduced a novel approach for learning dynamic gesture primitives for gesture and action recognition. An evolutionary computing framework was presented incorporating two most notable gesture recognition methodologies, namely DTW and HMMs. Experimental results show the competitiveness of our methods, outperforming state of the art results in benchmark

MSRAction3D-HS		MSRDaily3D-CV		MSRDaily3D-HS	
Method	Accuracy	**Method**	Accuracy	**Method**	Accuracy
[168] (LOP+J.)	88.2%	[71] (SOSVM)	68.3%	[167] (LOP)	42.5%
[175] (DCSF)	89.3%	[72] (SMMED)	73.20%	[112] (DTW)	54%
[130] (HOPC)	91.64%	[175] (DCSF)	83.60%	[168] (MKL)	80.0%
[49] (PBR)	92.3%	[175] (DCSF+Skl.)	88.2	[91] (GP)	85.6%
[169] (MMTW)	92.7%	-	-	[168] (LOP+J.)	85.75%
Dynamic Time Warping					
Baseline	85.76%	Baseline	77.36%	Baseline	70.20%
Evolved	90.89%	Evolved	**89.51%**	Evolved	**88.16%**
Hidden Markov Model					
Baseline	70.85%	Baseline	74.62%	Baseline	69.29%
Evolved	**95%**	Evolved	**91.39%**	Evolved	**92.30%**

Table 5.1 Recognition results in the MSRAction3D and MSRDaily3D datasets for half-split (HS) and cross-validation (CV), for the latter setting we report the 4 results available in published literature.

datasets after few generations. Our results suggest that the proposed subgesture learning methodology enhances the recognition performance of traditional techniques. Future work includes extending the framework for related tasks (*e.g.* gesture spotting, event detection) and an extensive evaluation under different parameter settings. In addition, this framework can operate directly as part of deep learning architectures and viceversa. Thus, we plan, first, to model representations based on deep learning approaches and use them as input features, so as to begin the evolution from such richer representations (as those computed from deep neural networks). Finally, we plan to model subgesture primitives based on deep learning methods at the inner steps of the evolutionary algorithm (*e.g.* as part of the fitness function).

Chapter 6

Applications for Human Analysis

This appendix provides an interdisciplinary approach for analyzing real conversations in the field of restorative justice. The use of several multimodal descriptors for discovering behavioral cues proves the effects of learning these features in real and sensitive scenarios, and provide a feeback for the experts in those multidisciplinar areas.

6.1 Non-verbal communication analysis in Victim–Offender Mediations

Restorative justice is an international social movement for the reform of criminal justice. This approach to justice focuses on the needs of the victims, who take an active role in the process, while offenders are encouraged to take responsibility for their actions *to repair the harm they have done* [172]. One of the common procedures offered to victims is the possibility of exchanging their impressions with a mediator, in a program known as the Victim-Offender Mediation (henceforth VOM) program. Given the sensitive nature of the cases, the process consists initially of a set of individual encounters, where each party involved (i.e. victim or offender) attends an interview or meeting with a mediator to analyze the problem in depth. The decision is then taken as to whether the victim and the offender might engage in a joint encounter. Figure 6.1 (a) shows an example of a real VOM scenario.

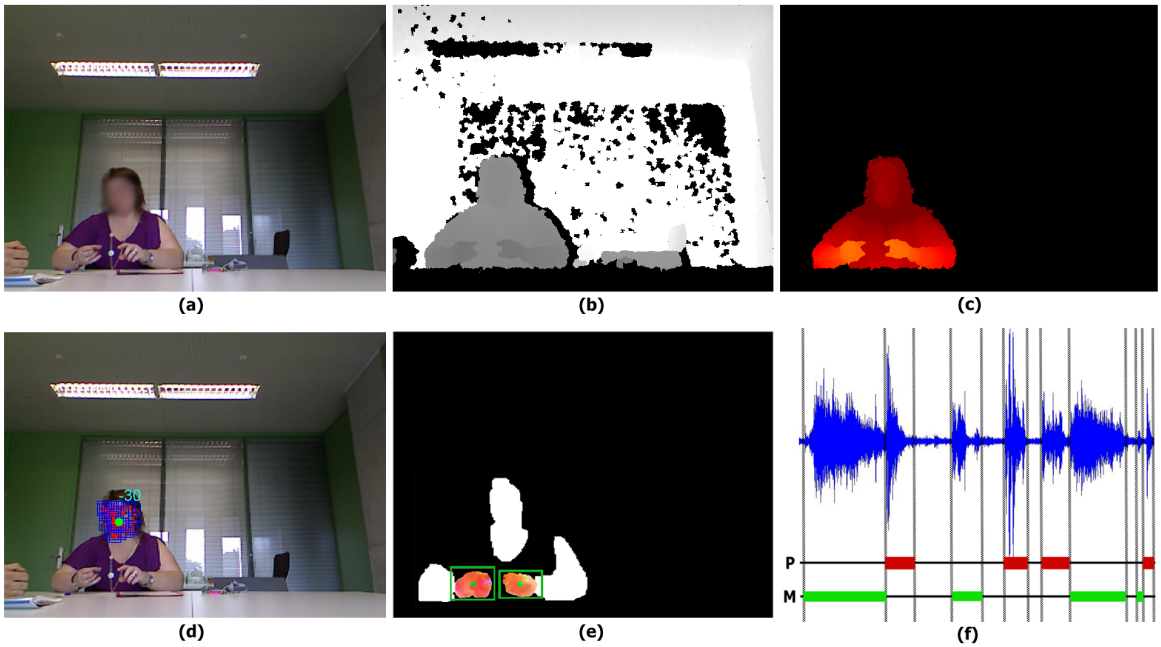


Fig. 6.1 Examples of the multi-modal feature extraction. Images (a) and (b) are the RGB and depth images, respectively. Image (c) shows the upper body obtained from the Random Forest user segmentation. In image (d), both face detection and head pose estimation are shown. Hand segmentation is shown in image (e). Across the regions segmented by color, the optical flow is shown in the regions in which there is greatest movement, identified as being the hands. Finally, image (f) illustrates the speaker diarization process with the two participants involved in the VOM session. The participants belong either to a party P or to the mediators M. Clusters belonging to each participant are obtained from the input signal, estimating the speech time of each segment, as well as the speech pauses/interruptions.

In the VOM process, the goal is to reach a restitution agreement by seeking to balance the interests of each of the parties, conditioned by the events that have occurred and the associated legal proceedings. This agreement can be reached in one of two ways. First, there are pre-conditioning factors to a case, given its particular facts, which make mediation feasible or not. Second, high levels of agreement and expressed satisfaction between the parties and the mediator are indicators of whether the VOM process is likely to end in success or failure [157]. The emergence of these indicators depends on a large set of factors that are not only concerned with the professionalism of the mediator, but are also related to other factors including the applicability of mediation, the participants' traits, human relationships, the first impressions, among others. Furthermore, if we examine each of the participants (victim, offender, and mediator), certain characteristics, including their cultural background, education, and social status, are likely to have a high impact on the success or otherwise of the process [118, 119].

Participant roles are clearly defined in these conversational processes, as they are in similar scenarios, such as job interviews. The mediator explains the process and listens to the other parties, maintaining his or her impartiality at all times, whereas the victim and offender are more concerned with protecting their own interests and may appear quite wrapped up in the problem they face. Indeed, no standard guidelines exist for establishing the best course of actions or identifying the psychological mechanisms for achieving the desired mediation goals. There exist, however, a set of body communicative cues that are present in the conversation and affect the way of how participants perceive each other. This non-verbal communication has been of high interest to intensively analyze the human interaction in social psychology and cognitive sciences [79].

In this context, multi-modal intelligent systems can be used to analyze this information by means of extracting features separately for the different data sources, such as those captured from low-cost sensor devices. They can then be combined so as to define and recognize communicative indicators. In this chapter, we present the first pattern recognition method of the state of the art for extracting multi-modal features and recognize social signals in VOM processes.

The rest of this chapter is organized as follows. Section 6.2 presents the material acquired and used in this study. In section 6.3, we describe the system modules. Section 6.4 outlines the proposal setup and the experimental results. Finally, section 6.5 concludes this chapter.

To the best of our knowledge, this chapter presents the first non-invasive ambient-intelligence framework of the state of the art for the semi-automatic analysis of non-verbal communication in VOM processes. We extract a set of multi-modal audio-RGB-depth features and behavioral indicators, which are then used to measure the degree of receptivity,

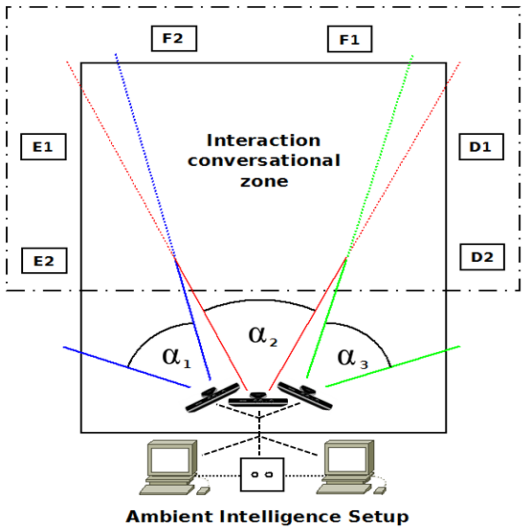


Fig. 6.2 Acquisition architecture. $E1$, $E2$, $F1$, $F2$, $D1$, $D2$ are the participants codified by their respective positions (E: left, F: front, D: right); the angles of view for the different cameras are the same, and hence $\alpha_1 = \alpha_2 = \alpha_3$.

agreement, and satisfaction using state of the art machine learning approaches and the ground truth defined by the mediators in the VOM sessions. As a result, we find that our technology achieves a high correlation between the most relevant features obtained by the behavioral indicators and the information provided by the experts.

6.2 Data collection

An environmental study was undertaken in the various rooms in which recording was to take place, and in which the non-invasive devices were to be set up. Once the environmental study had been completed, decisions regarding the ethical constraints that had to be satisfied were taken in order to protect the recorded data. This procedure involved the drawing up of three fundamental ethical documents: the researchers' signed undertaking, informed consent, and the case-codification.

As the sessions typically involve two or three participants, the homogeneous distribution of the cameras enabled us to capture at most two people-per-camera. Specifically, the devices used were three Kinect™ sensors and two laptops (which varied depending on the number of participants). Thus, a maximum of six people could be recorded[1]. Figure 6.2 shows the ambient intelligence setup with all the elements involved and their distribution.

[1]The maximum number of people in the recorded sessions was five.

Recordings were made in various towns and cities of Catalonia. Most of them were made in the capital city of Barcelona with a total of 15 sessions, followed by Vilanova i la Geltrú with a total of four. Two sessions were recorded in each of Manresa, Tarragona, and the youth penitentiary center in Granollers. Finally, one session was recorded in Terrassa.

Thus, 26 VOM sessions were recorded, with a duration from 20 minutes to 2 hours depending on the session, and an overall average of 35 minutes among all sessions. For each session, a mediator engaged in a conversational process with different parties. Of the total number of sessions, 15% were joint encounters, with both parties (victim and offender) being present in the VOM. The remaining sessions were individual encounters involving one or other of the parties and the mediator. Some of the sessions also involved accompanying persons, either a professional from the specific center, or experts in some particular field relevant to the case under discussion.

Each recorded session[2] provided audio-RGB-depth information. These modalities were registered using the camera parameters, and synchronized between the various devices through the system clock. The set of images for each session were recorded at a resolution of 640×480 and at an average of 12 frames per second (fps), both for RGB and depth information. Each audio channel, belonging to one of the four microphones spread out linearly along a multi-array microphone, processed 16-bit audio at a sampling rate of 16 kHz. The distance between participants and the Kinect™ device was between 1 and 2 meters depending on the recording facility.

As the data protection regulations only allow one mediator to annotate each session, the annotators were those mediators that had greatest familiarity with the case being dealt with in each session. Only in a few isolated cases there were two mediators in the session. Thus, in some cases the questionnaires completed by the mediators, recording their impressions and feelings regarding the party/ies and the overall sessions, were subsequently confirmed by a second mediator from the team so as to guarantee the consistency of the defined ground truth values. The system responses were determined by considering both the state of the art methods for the study of behavioral traits in people involved in similar scenarios, as those presented in Chapter 2.2 [7, 47, 74, 102, 110, 118, 119, 142, 143, 157, 160, 161], and in the subsequent discussion held with the mediators, taking into account the aims of their work with the Department of Justice. Finally, we defined the system's ground truth as:

- **Receptivity**: degree of engagement shown by each party during the session.
- **Agreement**: degree of agreement reached between the parties (quantified globally for each session).

[2] See an example of the different modalities and visual extracted visual features in the **supplementary video material sample.**

- **Satisfaction**: degree of agreement reached between the parties in relation to the mediator's expectations (quantified globally for each session).

Table 6.1 Summary of data acquired.

Individual encounters	22
Joint encounters	4
Total sessions	26
Penitentiary centers	1
Office centers	4
Total justice centers	5
Mediators	7
Parties	30
Total n^o participants	37
Total n^o frames	1,436,400
Average n^o minutes/session	35

The quantitative nature of these social responses was validated by a randomly selected mediator who had not been involved in that case so as to obtain a more objective evaluation. This approach was likewise applied to two features describing the evolution in the level of nervousness manifest by each party at the beginning and at the end of the process, respectively. Therefore, for each session and for each party, mediators ranked the observed quantity of these behavioral indicators from 1 to 5, where 1 is the lowest value and 5 the highest. Table 6.1 shows a numerical summary of the data acquired.

6.3 Proposed Methods

The proposed framework consists of three main sequential modules illustrated in Figure 6.3. The first module includes the multi-modal feature extraction from audio-RGB-depth data, which is described in Figure 6.4. As shown in the scheme, the steps for obtaining multi-modal features from different sources of information are the speaker diarization, user segmentation, and region detection. Once the multi-modal features have been extracted, they are used to define the behavioral indicators to be learnt and classified.

For all the system's modules, consder a set of recorded sessions from a set of VOM cases. Since a case is divided into one or more VOM sessions, one session $\mathbf{v}$ may belong either to the same case as another session, or to a different case.

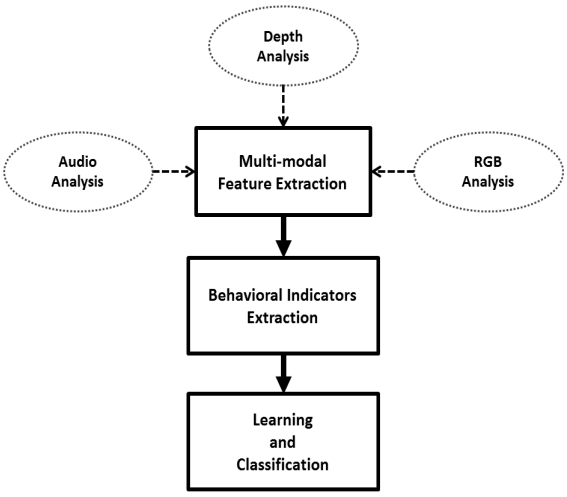


Fig. 6.3 Modules of the proposed system.

The remainder of this section describes the different blocks of Figure 6.3. First, the multi-modal feature extraction illustrated on Figure 6.4, followed by the behavioral indicators, and finally the learning and classification of receptivity, agreement, and satisfaction labels.

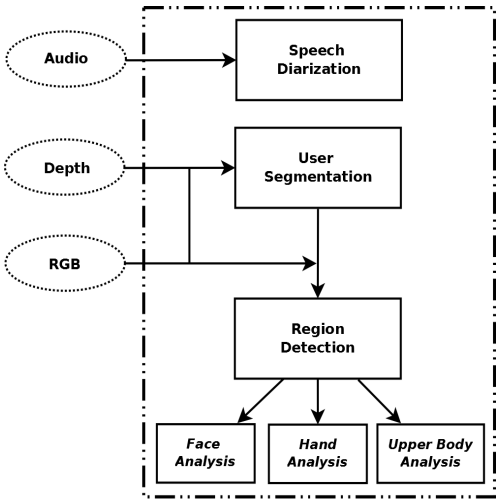


Fig. 6.4 Multi-modal feature extraction module.

6.3.1 Audio Analysis: Speaker Diarization

In order to obtain the audio features, we use a diarization scheme based on the approach presented in [34]. These features correspond to state of the art methods for audio descriptions, which have been successfully applied in several audio analysis applications [4, 6, 129]. The process is described below:

Description: The input audio is analyzed using a sliding-window of 25 ms, with an overlap of 10 ms between consecutive windows, and each window is processed using a short-time Discrete Fourier Transform (DFT), mapping all frequencies to the Mel scale. A more precise approximation of this scaling for frequencies used in Mel Frequency Cepstral Coefficients (MFCC) implementations, is represented as:

$$\hat{f}_{mel} = k_{const} \cdot \log_a \left(1 + \frac{\hat{f}_{lin}}{F_{const}}\right), \tag{6.1}$$

where F_{const} and k_{const} are constant values for frequency and scale, respectively. The Koenig scale $\hat{f}_{lin}$ is exactly linear below 1000 Hz and logarithmic above 1000 Hz. In brief, given A-point DFT of the discrete input signal $\tilde{x}(\dot{a})$,

$$\tilde{X}(\dot{b}) = \sum_{\dot{a}=0}^{A-1} \tilde{x}(\dot{a}) \cdot \exp\left(\frac{-2\pi\dot{a}\dot{b}}{N}\right), \dot{b} = 0, 1, ..., A-1, \tag{6.2}$$

a filter bank with several equal height triangular filters is constructed. Each of these filters has boundary points expressed in terms of position, which depends on the sampling frequency and the number of points A in the DFT. Finally, the Discrete Cosine Transform (DCT) is used to obtain the first 13 MFCC coefficients. These coefficients are complemented with the first and second time-derivatives of the Cepstral coefficients.

Speaker segmentation: Once the audio data are properly described by means of the aforementioned features, the next step involves identifying the segments of the audio source which correspond to each speaker. A first coarse segmentation is generated according to a Generalized Likelihood Ratio, computed over two consecutive windows of 2.5 s. Each block is represented using a Gaussian distribution, with a full covariance matrix, over the extracted features. This process produces an over-segmentation of the audio data into small homogeneous blocks. Then, a hierarchical clustering is applied to the segments. We use an agglomerative strategy, where initially each segment is considered as a cluster, and at each iteration the two most similar clusters are merged, until the stopping criterion of the Bayesian Information Criterion (BIC) is met. As in the previous step, each cluster is modeled by means of a Gaussian distribution with a full covariance matrix and the centroid distance is used as the link similarity. Finally, a Viterbi decoding is performed in order to adjust the segment boundaries. Clusters are modeled by means of a one-state HMM using GMM as our observation model with diagonal covariance matrices. Figure 6.1 (f) represents an example of this procedure, showing the clusters where the speech signal falls at each instant. Since most of the participants appear in just a single mediation session, we do not learn any speaker

models from the cluster GMMs. Therefore, models extracted from one session are not used in the diarization process of other sessions.

6.3.2 User Detection

Both RGB and depth data are used for the postural and behavioral analyses of the parties. Examples of these images are illustrated in Figure 6.7 (b) and (b), respectively. In this sense, the first step involves performing a limb-segmentation of the body based on the Random Forest method of [145]. Figure 6.7 (c) shows a user detection example of applying this segmentation. Once regions of interest have been located, it is of particular interest to obtain real-world distance values for certain computed features so that they are comparable between different subjects. To do this, we employed a similar procedure to that explained in [Fisher], which converts the 2D pixels into 3D real-world coordinates using the Kinect™ depth values. However, since these raw sensor values returned by the depth sensor are not directly proportional to the depth, in [Fisher], they scale with the inverse of the depth. Therefore, each pixel $(\dot{x},\dot{y})$ of the depth camera can be projected to metric $3D$ space as:

$$x = (\dot{x} - \delta_x)\frac{d(\dot{x},\dot{y})}{\kappa_x}, y = (\dot{y} - \delta_y)\frac{d(\dot{x},\dot{y})}{\kappa_y}, z = d(\dot{x},\dot{y}), \tag{6.3}$$

where (x,y,z) will be the real world coordinates, and δ_x, δ_y, κ_x, κ_y, the intrinsics of the depth camera. These values will be computed over the detected interest regions in order to define the communicative indicators described in next sections.

6.3.3 Region Detection

This section describes the different feature extraction modules applied to the visual data source once the user has been segmented. Specifically, we perform an analysis of the face, hands, and upper body, as well as visual movements in these regions during conversations.

Face Analysis

We are primarily concerned with obtaining the head pose angle of each of the participants in the session. To do this, we base our approach on that of [184] which uses a set of face models. The face model is based on a mixture of trees with a shared pool of parts, where every facial landmark is modelled as a part and global mixtures are used to capture topological changes due to viewpoint. Global mixtures can also be used to capture gross deformation changes for a single viewpoint, such as changes in expression. On the other hand, the detection of

the head pose angle is performed by averaging HOG feature as a polar histogram over 18 gradient orientation channels, as computed from the entire PASCAL 2010 dataset [48]. In Figure 6.7 (a) we can visualize the set of computed features plotted on the detected face.

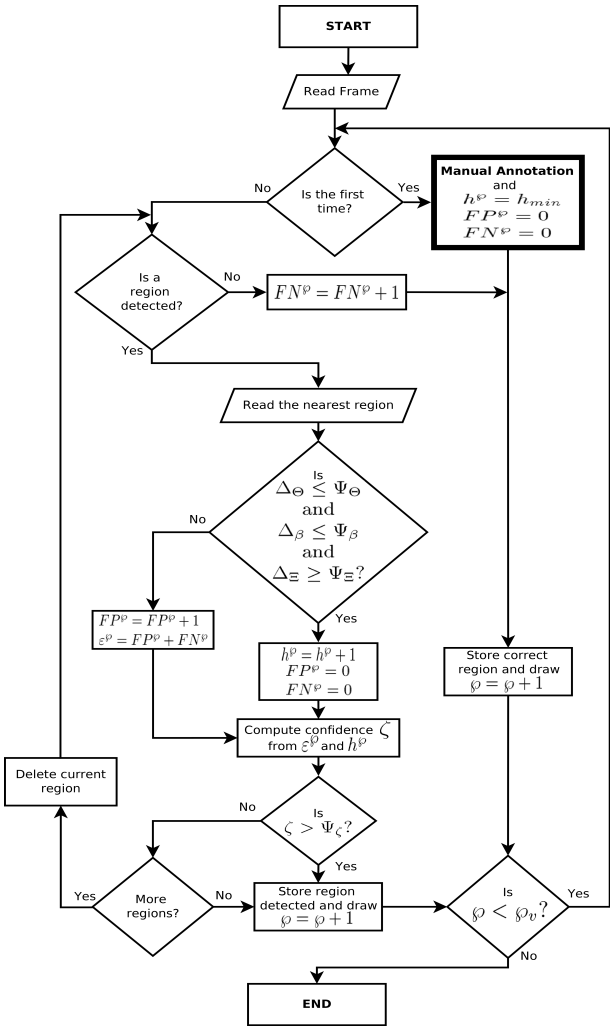


Fig. 6.5 Flowchart of the heuristic procedure applied to each frame. The total number of people that appear in the current video v is denoted by $\wp_v$. Constraints of the main condition at the center of the flowchart are denoted by Δ_Θ, Δ_β, Δ_Ξ, and their respective thresholds Ψ_Θ, Ψ_β, Ψ_Ξ. The counting variables are $FN^{\wp}$, $FP^{\wp}$, $h^{\wp}$, representing the accumulated number of false negatives, false positives, and hits for the current person $\wp$. They are used to compute the confidence ζ from the accumulated detection errors $\varepsilon^{\wp}$ and the hits $h^{\wp}$, and to decide whether the current detected region has to be stored or discarded through the threshold Ψ_ζ.

While face detection takes place for each tested image, we use a semi-automatic heuristic procedure of [125] so as to improve the continuity of positive detections of regions of interest in the person between consecutive frames, and to correct possible erroneous detections due to the inherent difficulties of the problem at hand. Figure 6.5 shows the flowchart of the

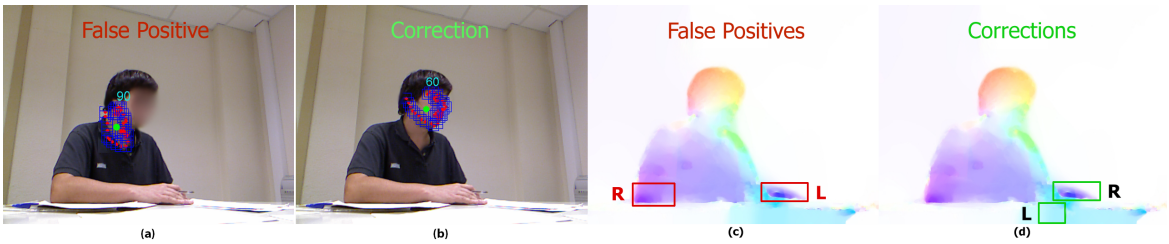


Fig. 6.6 Examples of how the semi-automatic heuristic procedure of [125] works on two pairs of frames of a session. The correction of false positives is shown, improving the continuity of the detection of positive regions of interest between consecutive frames. Image (a) shows a false positive detection for the face region, whereas image (b) shows its correction with the proper fitting. Image (c) shows false positive detections for the hand regions, choosing those blobs obtained by means of skin segmentation having highest optical flow with respect to the previous frame. Image (d) shows the correction of these regions by comparing them with the positive hand detections, recovered from the previous frames.

procedure applied to each frame. In short, it consists of a temporal filtering methodology of detected regions (faces) between one-by-one consecutive frames. It is based on three main constraints that enable us to choose the detected regions in the current frame by comparison to the previous one: offset pixels produced by the mass centers, offset angle produced by head poses, and the size difference factor produced among the region areas. Thus, three thresholds Ψ_Θ, Ψ_β, and Ψ_Ξ are respectively used to discriminate the occurred cases on each constraint, whose values may vary depending on the session conditions. Moreover, there are three counting variables that accumulate, for each person, the number of correct detections (hits) $h^{\wp}$, false positives $FP^{\wp}$ and false negatives $FN^{\wp}$. Then, a confidence ζ is computed from $h^{\wp}$ and the sum of false detections $\varepsilon^{\wp}$ to decide whether the current detected region has to be stored or discarded by means of the threshold Ψ_ζ. These counting variables are highly dependent on constraint thresholds, as they make the system more or less restrictive when choosing detected regions. Therefore, a trade-off between constraint thresholds and control thresholds should be reached when assigning their values in order to assure the continuity of positive region of interest detections for that person (even though the method could not detect any region in the image), and to decide whether a manual annotation is required to re-initialize the detection process in the (approximately) desired frequency rate. Figure 6.6 (a) and (b) shows an example of correcting a false positive detection.

Hand Analysis

Given that the skeletal model computed from the person segmentation image [145] does not offer an accurate fit of the hand joints in our particular scenario, we designed a semi-automatic procedure for hand detection.

First, hands are manually annotated in the starting frames of each session to perform posterior color segmentation for the rest of the frames. In this way, a GMM is learned with the marked set of most significant pixels, defining the skin color model of the person. Then, subsequent frames are tested within the GMM built using a threshold ϑ, discriminating those pixels belonging to the skin color from those belonging to the background. The resulting blobs are filtered using mathematical morphology closing operation with a 3×3 square structured element to discard noise and to obtain smoother regions. Once the set of blobs has been obtained, we need to choose those two candidates that belong to the hand regions. This is performed by computing the optical flow between consecutive frames, which allows to discard noise in those cases in which we obtain more than two blobs by retaining those with higher movement. The bounding boxes of Figure 6.6 (c) show an example of detections (left is incorrect) using this procedure.

To improve the detection, we use the same heuristic procedure as that applied to the face analysis step for choosing, in this case, the two best hand candidates. Image (d) of Figure 6.6 shows an example of how the heuristic procedure corrects false positive detections on the regions of the hands. The incorrect regions detected in the first instance are the blobs presenting the highest optical flow, and then the heuristic procedure corrects these regions by comparing them with the hand regions obtained from the previous frame. As in face detection, manual annotation may be required in those cases where the heuristic procedure needs to be re-initialized. For this task, an interface has been designed for the manual annotation of the hand regions for the set of frames in which this occurs. When the user makes any annotation, the GMM color model is newly re-constructed at this frame using the marked pixel positions, and the whole process is repeated. In this case, using the proposed heuristic we also found similar reduction regarding manual interaction effort as in the case of face region detection.

Once we have obtained the blobs belonging to the hand regions, the extremes with higher optical flow magnitude are used to obtain $2D$ hand positions. Finally, these positions are transformed to $3D$ real world coordinates using Eq. 6.3.

Upper Body Analysis

The probability of each pixel of an image belonging to a labeled body part is computed using depth features. This information is used for the subsequent calculation of optical flow

on RGB images where the upper body region appears. Therefore, each pixel of the image, detected by Random Forest, with high probability of being part of the person, is used to calculate the optical flow. Finally, an average of optical flow is computed for the upper body region, which is later used to define behavioral indicators. An example of user detection where upper body region is highlighted is shown in Figure 6.1 (c).

6.3.4 Behavioral Indicators

Once the multi-modal features have been extracted, we use them to build a set of behavioral indicators that reveal communicative cues. This set of behavioral indicators defines the final feature vector for each party within the VOM process. This information is of great interest in detecting the response of subjects to certain feelings or emotional states during the conversation [79]. In particular, since the behavioral cues of the mediators are not of interest for our purposes here, we focus mainly on those of the parties.

Target Gaze Codification

The head pose and the face is obtained by applying the methodology explained in section 6.3.3. In a given session, we compute the correlations between the head pose angles belonging to each participant and the positions taken by the remainder participants in that session. Hence, we identify the visual focus of attention among the different participants in the conversation [7, 9, 103]. For this purpose, different ranges are assigned to each participant in terms of angle limits. Given that the participants belonging to the same party are seated in adjacent positions (see acquisition architecture in Figure 6.2), each range represents a possible participant vision field of his/her gaze towards the target party. Thus, given a frame of the session and a participant, if his/her head pose angle falls within a particular range, then the party found within that range is identified as the target gaze of this participant for that frame, which means the participant is looking at this party. Since sessions have different setups, they may consist of one or two parties (and the mediator), each with a different number of participants. Therefore, the ranges require manual assignment depending on each session setup. Then, the target gazes are automatically identified for all the frames of the session.

Figure 6.7 (a) shows an example of crossed gazes between the mediator and a party in a real VOM session. Finally, we compute the time percentages of target gazes for each party. Therefore, for any given party, there is a total of 6 indicators for representing the target gazes ($\{f_{15}, f_{16}\}$ and $\{f_{18} - f_{21}\}$ from Table 6.2).

Agitation Estimation

As explained in section 6.3.3, $3D$ positions belonging to the hand regions are computed from the extreme positions of higher optical flow. From these positions, we are able to quantify the movement for each region between consecutive frames. For this purpose, let $F = \{\iota_1, \iota_2, \iota_3, ..., \iota_n\}$ be a set of consecutive frames, This set of frames belongs to a video session, being n the maximum length of the set.

Then, for each region we compute the average agitation over all the frames $\iota \in F$ as:

$$A_h = \frac{1}{n} \sum_{i=1}^{n} \Delta_h^{\iota_i}, \tag{6.4}$$

where $\Delta_h^{\iota_i} = \Delta_p^{\iota_i} + \Delta_q^{\iota_i}$ are the displacements among $3D$ positions of hands Δ_h (left Δ_p and right Δ_q) between frames ι_i and ι_{i-1}, computed using Euclidean distance. Therefore, A_h contains the accumulated average of displacements produced by both hands between frames F.

On the other hand, in section 6.3.3 we presented how the average optical flow can be obtained from the upper body region. Therefore, if we denote as $\bar{\sigma}_{\iota_i}$ the average optical flow of the upper body for a given frame $\iota \in F$, then:

$$A_b = \frac{1}{n} \sum_{i=1}^{n} \bar{\sigma}_{\iota_i}, \tag{6.5}$$

where A_b contains the accumulated average of optical flow produced by the upper body between frames F.

In short, for each party and session, agitation averages are computed over processed frames, with a total of 8 agitation indicators ($\{f_{14} - f_{21}\}$ from Table 6.2), either alone or in combination with other indicators. The idea of combining these indicators with other behavioral features is inspired by [39, 47]. In this case, we consider a combination between the features describing the agitation from the upper body and those describing the hands while looking at the participants, as in [125].

Posture Identification

From the $3D$ body position, we detect the body posture as one behavioral indicator, which may describe the engagement (or involvement) of the party within the VOM session. Our description of body posture is classified into three main positions (tilted backward, normal, tilted forward), where the posture selected is the one that has the most occurrences over the processed frames.

Fig. 6.7 Visual instances of some situations where behavioral indicators are detected in VOM sessions. Image (a) shows the detection of crossed gazes between the mediator and the other participant. Images (b) and (c) show a depth image and its segmentation for the person (white point cloud) and the table (red point cloud), respectively, which is used to detect a situation in which the target subject appears with his or her hands under the table.

In addition, $3D$ hand positions are used to detect where the hands are along the processed frames, in terms of average and time percentages. In particular, we discriminate three cases (i.e. 3 indicators): hands together, hands touch the face, and hands under the table. This is done in a similar way as for the agitation estimation, using Euclidean distance computed over $3D$ positions.

- Hands together: We compute for each frame the distance between left and right hand positions belonging to the target subject, and we consider the frames where the distance values are below that of a threshold. Finally, we compute the time percentage for those frames where the target subject appears with their hands together.

- Hands touch the face: We compute for each frame the distance between each hand position and the position belonging to the face center of mass obtained in section 6.3.3. Then, we consider the frames where the distance values are below that of the threshold. Finally, we compute the average distance for those frames where the target subject appears with their hands touching their face.

- Hands under the table: For each frame, we first perform a segmentation of the tables using [138] to obtain planar objects within images. Then, we compare the $3D$ positions of both hands with the position of the tables in order to discriminate the two possibilities where the hands may appear under or above the table. Finally, we compute the time percentage for those frames where the target subject appears with their hands under the table. Figure 6.7 (b) and (c) illustrate an example of this procedure, showing respectively the input depth image and its segmentation.

Table 6.2 Summary of behavioral indicators defining each feature vector. The last two features derive from the mediator surveys.

Feature	Brief description
f_1	Party's role within the VOM session (victim or offender)
f_2	This party looks at the other
f_3	The other party looks at this party
f_4	This party looks at the mediator
f_5	The mediator looks at this party
f_6	Body posture inclination of this party
f_7	Gender of the mediator
f_8	Gender of this party
f_9	Gender of the other party
f_{10}	Age of the mediator
f_{11}	Age of this party
f_{12}	Age of the other party
f_{13}	Session type (individual/joint encounter)
f_{14}	Upper body agitation of this party
f_{15}	Upper body agitation of this party while looking at the other party
f_{16}	Upper body agitation of this party while looking at the mediator
f_{17}	Hands agitation of this party
f_{18}	Hands agitation of this party while looking at the other party
f_{19}	Hands agitation of this party while looking at the mediator
f_{20}	Hands agitation of the mediator while looking at this party
f_{21}	Hands agitation of the other party while looking at this party
f_{22}	Hands together of this party
f_{23}	Hands of this party touching the face
f_{24}	Hands of this party are under the table
f_{25}	Mediator speaking time
f_{26}	Speaking time of this party
f_{27}	Speaking time of the other party
f_{28}	Mediator speaking turns
f_{29}	Speaking turns of this party
f_{30}	Speaking turns of the other party
f_{31}	Mediator interrupts this party
f_{32}	This party interrupts the mediator
f_{33}	This party interrupts the other party
f_{34}	The other party interrupts this party
f_{35}	Nervousness of this party at the beginning
f_{36}	Nervousness of this party at the end

Speech Turns/Interruptions Detection

The speaker diarization process of section 6.3.1 detects time segments belonging to each participant in the VOM process. In order to extract the degree of interaction, we not only use the length of time during which each participant speaks, but we also count the number of turns in each session. This enables to differentiate between a session where each party expresses its position from a session in which a conversation is maintained between the VOM participants. Apart from the quantification of turn taking, a relevant indication in the social communication analysis is the detection of interruptions, which are related to the dominance and respect between two persons [45]. Using the time between turns, we compute the percentage of turns in which a participant interrupts another one. For instance, in the first three turns of Figure 6.1 (f) a participant (red) interrupts the mediator (green), while the mediator waits until the other participant ends his turn before starting to speaking again.

6.3.5 Classification

The total number of behavioral indicators is 36 (see Table 6.2, which define the feature vector for each sample in our dataset). Here, we define a sample as each party participating in a VOM session. Thus, if a session involves two parties and the mediator, we introduce one sample of 36 features for each of the two parties. On the other hand, if a session involves just one party and the mediator, we introduce only one sample corresponding to the party involved. Each party of a video session is a sample for the classification task, and the total number of used samples is 28.

As explained in section 6.2, the observations of the classification task are the accuracies achieved by the system when predicting receptivity, agreement, and satisfaction. Then, the correlation can be observed between the observations predicted by the system and the impressions recorded by the mediators. These opinions are quantified values of receptivity, agreement, and satisfaction presented in relation to the parties involved in the VOM session, and represent the ground truth of our system. The ground truth values are assigned to each sample of the dataset. Since agreement and satisfaction are globally assigned for each session, those sessions containing two parties will share the same ground truth labels of agreement and satisfaction for both generated samples, meanwhile the receptivity ground truth value is assigned to each sample (party) independently.

Learning is then performed on these samples and their features as a binary classification problem, grouping into two classes the quantifications performed by the mediators. To do this, we employ four classical techniques from the machine learning field: AdaBoost [55], Support Vector Machines (SVM) using a Radial Basis Function (RBF) [24], Linear

Discriminant Analysis (LDA), and three kinds of Artificial Neural Networks (ANN), in particular Probabilistic Neural Networks (PNN) [149], and Cascade-Forward (CF) and Feed-Forward neural networks (FF) [61]. In addition to the binary classification analysis we also conduct a regression study using epsilon-SVR (Support Vector Regression) [24] in order to predict continuous quantifications of the three labels.

6.4 Experiments

6.4.1 Setting and Validation Measurements

The measurements for the features referring to the gaze, interaction of hands, and the position of hands respect to the table, are time percentages. The features referring to agitation, combination of agitation and gazes, and interaction of hands with the face, contain averaged values of optical flow or distances, all of them taking into account the processed frames of a session. The features referring to the speech are turn taking percentages, where a turn means that the speaker changes. Finally, the remaining features, including nervousness features, are codified either into binary values or discrete values within a certain range, having 5 as the maximum range length.

In addition, an alternative was implemented where some features are divided into two -one belonging to the first half, the other to the second half of the session-. This procedure was initially performed to identify behavioral changes in subjects during the different halves of the session. However, no significant differences were found and, hence, we finally used the set of features without any temporal segmentation.

Table 6.3 Accuracy considering the first grouping case and all features.

Label	AdaBoost	LDA	PNN	CF	FF	SVM
Satisfaction	57%	32%	57%	57%	**86%**	57%
Agreement	50%	54%	64%	64%	75%	64%
Receptivity	64%	50%	71%	71%	68%	75%

Table 6.4 Accuracy considering the second grouping case and all features.

Label	AdaBoost	LDA	PNN	CF	FF	SVM
Satisfaction	82%	43%	21%	82%	75%	82%
Agreement	71%	43%	29%	71%	75%	75%
Receptivity	75%	36%	39%	68%	75%	61%

Table 6.5 Accuracy considering the first grouping case and withholding the nervousness features.

Label	AdaBoost	LDA	PNN	CF	FF	SVM
Satisfaction	57%	57%	57%	68%	64%	57%
Agreement	50%	43%	64%	57%	71%	64%
Receptivity	68%	46%	71%	75%	68%	75%

Table 6.6 Accuracy considering the second grouping case and withholding nervousness features.

Label	AdaBoost	LDA	PNN	CF	FF	SVM
Satisfaction	82%	61%	21%	71%	**86%**	82%
Agreement	71%	57%	29%	71%	**79%**	75%
Receptivity	**79%**	46%	39%	64%	71%	61%

Learning is performed using leave-one-out validation, keeping one sample out of the testing each time. Since the total number of samples is small and the ground truth values are quantified within ranges $[1-5]$ (as for the nervousness features), we simplified the problem by grouping the different response degrees into binary groups, but we also performed a posterior regression analysis. In the case of a binary setup, the value 3 can be considered as being either high or low. For this reason, we ran the experiments twice to test each grouping case, as we show in the result Tables 6.3, 6.4, 6.5, and 6.6:

- First grouping case: Degrees of quantification $\{1,2,3\}$ versus $\{4,5\}$.
- Second grouping case: Degrees of quantification $\{1,2\}$ versus $\{3,4,5\}$.

In our experiments, we awarded the standard value of 50 to the number of decision stumps in the AdaBoost technique. For the SVM-RBF and epsilon-SVR, we experimentally set the cost, gamma, and epsilon parameters by means of the leave-one-out validation for each social response and minimizing regression deviation on the training set. Finally, we applied the same tuning procedure for the three standard neural network parameters: a Probabilistic Neural Network (PNN) with a spread value of 0.1 for the radial basis functions, and Cascade-Forward (CF) and Feed-Forward (FF) neural networks, both with a single hidden layer with 10 neurons values and Levenberg-Marquardt back-propagation training function. The results obtained are shown in terms of accuracy percentages.

Due to the sensitive nature of the VOM process, never before (to the best of our knowledge) have mediators recorded their sessions so that they might subsequently analyze the cases. In this respect, therefore, the first results to emerge from this study are the session

videos themselves, which are valuable materials via which the mediators can share their experiences and obtain feedback to improve their mediation skills.

6.4.2 Results

The predictions addressed in our classification task focus on three indicators: the degree of receptivity of the parties, the level of agreement reached, and the degree of mediator satisfaction. Tables 6.3 and 6.4 show the results obtained when employing the different techniques and using the complete set of behavioral indicators of Table 6.2. Note that as the features of nervousness are subjective indicators that are not automatically computed, we repeated the experiments without these two features. These results are shown in Tables 6.5 and 6.6, where the prediction is also analyzed under the grouping hypotheses. The most accurate results among the four tables for the three responses are shown in bold, showing both which classifier and which grouping case give the best performance for each feature description. Once again, the results show a correlation between the features extracted and the categories selected. The percentage degree of accuracy in the predictions is then compared for the different techniques: AdaBoost, SVM, LDA, PNN, CF, and FF. It can be noted that, except for PNN and LDA (which are not good techniques for use with our dataset), all the classifiers are able to make predictions about the random decision. This indicates that there is a correlation between the captured data and the information that we want to predict. The most predictable social response is that of satisfaction, presenting an accuracy of 86% with the FF, followed by 82% with AdaBoost, SVM, and CF. The best result when predicting agreement was an accuracy of 79% with FF and, similarly, when predicting receptivity, the best accuracy was 79% with AdaBoost. These results are quite significant since most of the sessions presenting high values for this combination of responses resulted in satisfactory VOM outcomes. However, since the number of samples is, in general, small, all responses vary in their performance depending on the grouping hypothesis, despite the low level of presence of the 3-value among the quantitative responses. This means that the uncertainty of the mediator when assigning a value of 3 to the answers tends to add noise to the overall data with respect to the evaluation.

The result tables show that CF and FF (and even LDA) vary significantly in their predictions depending on whether the nervousness features are considered or not. This indicates that the subjective evaluation of the mediator adds an important weight to the system for half of our classifiers. Moreover, the variability in performance presented by the remaining classifiers in relation to these two cases leads us to analyze the relevance of these features in each case. Thus, we performed a comparison to identify the most relevant features for each social response. In this way, we also analyzed the influence of the nervousness features

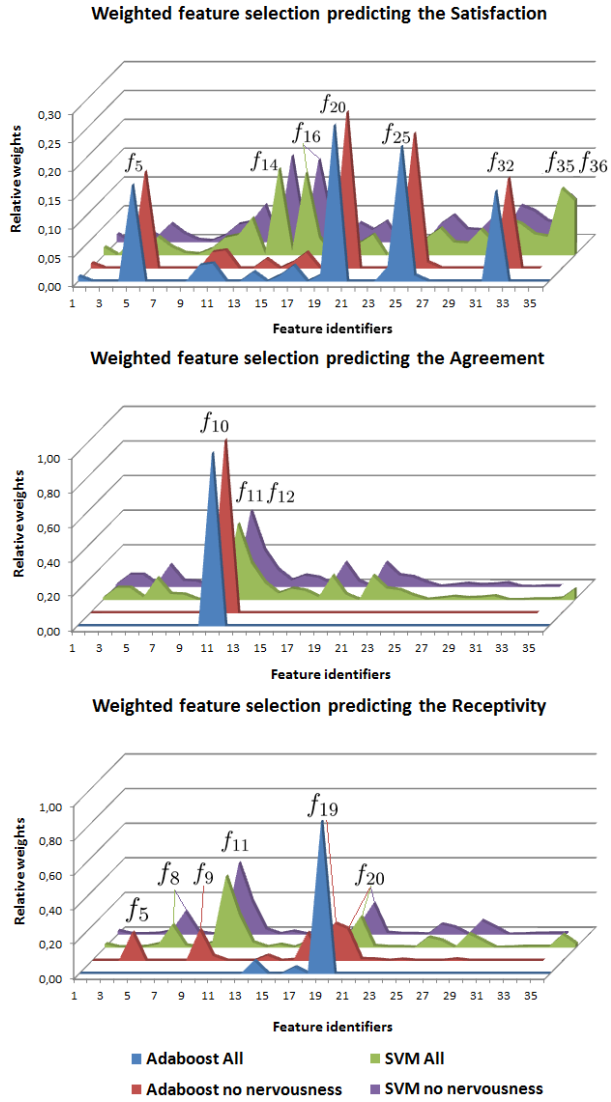


Fig. 6.8 Weighted feature selection when using AdaBoost and SVM for the grouping response cases presenting the highest accuracy when predicting receptivity, agreement, and satisfaction. Each line represents the relative feature weights assigned by the classifiers within the range $[0, 1]$, either employing all features or without the nervousness features f_{35} and f_{36}.

when choosing the most relevant of the other features. We performed a weighted feature selection using [46] and [171] for AdaBoost and SVM, respectively. For each response (receptivity, agreement, and satisfaction), we selected those features only for the cases giving the highest degree of accuracy (see the different plots in Figure 6.8). In general, we observe that agitation features and the mediator's speaking turns are chosen as the most relevant features when predicting satisfaction. By contrast, the feature chosen as being most relevant for predicting agreement is the age of the mediator. In the case of receptivity, the fact of withholding the nervousness features results in the most significant changes in the feature

selection with respect to the other responses. However, both hand agitation, gaze, and the combination of the two are chosen as being the most relevant features when predicting receptivity. On average, the most relevant features for all the responses are those involving the combination of gaze and the agitation of the body regions. This means that these are the most discriminating behavioral indicators in the prediction of the degree of receptivity, agreement, and satisfaction in a conversation such as that maintained in a VOM process. This feature selection procedure has direct implications for the observational methodology of non-verbal communication, since it allows experts in the field of psychology and restorative justice to focus, in any given conversation, on the most discriminating behavioral indicators automatically selected through artificial intelligence.

Finally, we relate the overall training data to the different ground truth annotations using the epsilon-SVR regression strategy. In this case, when using the leave-one-out strategy, we obtain a prediction for each sample within the same range as the quantified annotations $[1,5]$. In this setting, we also ran the experiment twice: first, we considered all features, and then left out the nervousness features. Both cases gave similar average distances when predicting satisfaction, agreement, and receptivity, with values of 0.59, 0.64, and 0.68, respectively. This mean deviation with respect to the ground truth labels was found accurate and of interest to the team of mediators.

6.5 Conclusion

We proposed a multi-modal framework for the semi-automatic analysis of non-verbal communication in VOM sessions. We showed the usability of computer vision, signal processing, and machine learning strategies in conversational processes. Specifically, we computed a set of multi-modal features from multimodal data. Then, we defined an automatic computation of behavioral indicators used as final features for learning and classification tasks. We demonstrated the applicability of the system to be used in the restorative justice field as a tool for mediators, obtaining recognition accuracies of 86% when predicting satisfaction, 79% when predicting both agreement and receptivity, and a high correlation in the regression analysis.

As future work, we plan to improve the dataset and responses, and to incorporate new features. In the case of the data, we hope to capture more samples so as to be able to perform more accurate predictions, providing continuous ground truth information by means of intra-mediator estimations. In the case of the predictions, new data should allow the continuous prediction of each degree of the behavioral indicators. Moreover, it will enable us to perform frame-based predictions, analyzing the evolution of each indicator throughout the

VOM process, and to detect the exact instant when a party accepts the possibility of reaching an agreement. Finally, we plan to incorporate emotional state features obtained from facial expressions [135] and audio data [8].

Chapter 7

Conclusions

In this dissertation, we presented novel theoretical approaches based on the BoW paradigm and a system built for describing a new abstraction from low-level features into behavioral cues for language communication.

In Chapter 2, we described a general background of common representations that are widely used in the field of artificial intelligence. Indeed, we presented a review of the state of the art for those methods used in the literature of the computer vision and machine learning communities, and how the developed approaches cover the different aspects to be considered in several domains, with particular emphasis referred to the human behavior in language communication.

In Chapter 3, the use of evolutionary algorithms for improving the BoVW representations based on weighting schemes is described. In this sense, we designed different combinations of weighting schemes that are commonly used in text mining, and demonstrated the effectiveness on their application for computer vision tasks. In order to demonstrate their effectiveness, we made a wide comparison amongst several domains, where such evolved representations enhance the generalization capabilities for recognizing class-categories, both in still images and videos. Therefore, we showed that learning weighting schemes by means of genetic programming leads to an improvement of performance of traditional schemes in image classification and gesture recognition. Future work includes studying alternative methodologies for learning term-weighting schemes based on inner optimization of the representation matrix. Also, we are interested on learning term-weighting schemes on multimodal data coming from different modalities.

In Chapter 4, we claimed the temporal modelling of multimodal data by means of dynamic programming and generative models in order to be used in those domains demanded by the nature of data. At the same time, we described how fusion strategies are used to model multimodal data coming from different sources of information for the task of gesture recognition. We showed how a combination of such approaches may entail an improvement of performance in some well-known gesture recognition datasets. Future work lines are the inclusion of samples with different points of view for the same gesture classes, and the definition of powerful descriptors to obtain gesture-discriminative features.

In Chapter 5, we proposed the integration of evolutionary algorithms to model dynamic representations of data by means of iterative feature selection and temporal clustering. We showed the capabilities of these approaches as global optimization methods that can be used either in conjunction with deep learning architectures or to evolve deep representations. We demonstrated how these methodologies outperforms the task of recognizing categories in several action datasets. Future work consist of modelling representations based on deep learning approaches and use them as input features, so as to begin the evolution from such

richer representations. On the other hand, we plan to model subgesture primitives based on deep learning methods at the inner steps of the evolutionary algorithm. Finally, we plan to extend the framework for related tasks (*e.g.* gesture spotting, event detection) and an extensive evaluation under different parameter settings.

Since the approaches presented in the aforementioned chapters are mostly applied to datasets of actions and gestures, some domains of those datasets refer to different ways of human communication through language. Hence, in Chapter 6 we presented a system developed for a real application in conversation settings within the field of Restorative Justice. We described a mid-level representation of features computed from multimodal data, acquired from sensor devices that were present in VOM sessions as part of the scenario and environment. We explained how these features can be modelled as behavioral indicators in order to predict several responses of the conversations, and we provided an extensive analysis about them with the goal of producing a feedback for the experts in that field. Thus, we defined an automatic computation of behavioral indicators used as final features for learning and classification tasks, achieving promising results when predicting agreement, satisfaction, or receptivity in such conversational settings. Future work consists, first, of increasing the volume of the dataset by capturing more samples and computing more features so as to improve the overall performance on predicting the responses. At the same time, a continuous ground truth for annotating the responses would enable us both to the continuous prediction of each degree of the behavioral indicators, and to perform frame-based predictions to analyze the evolution of each indicator throughout the sessions. Finally, we plan to incorporate emotional state features obtained from facial expressions.

As an overall feeling of the work made in this thesis, we showed how learning is present in machines to model different representations for several domains, making special emphasis to applications for language communication. In this sense, we tried to promote the sense of learning through the system feedback, *i.e.* how humans can learn from the outcomes of a system built for a particular objective for improving themselves in their areas of knowledge. For instance, from the analysis of those particular aspects that are difficult to capture by humans involved in a conversation, but that can be easily 'seen' by machines. Therefore, this thesis also evidences that both humans and machines depend each other by means of learning, so that they should learn to trust and live together in harmony as part of our future nature of being.

References

[1] (2010). Open natural interface. http://www.openni.org.

[2] (2010). *Prime Sensor NITE 1.3 Algorithms notes*. PrimeSense Inc. http://www.primesense.com.

[3] Aggarwal, J. K. and Ryoo, M. S. (2011). Human activity analysis: A review. *ACM Computing Surveys*, 43(3).

[4] Ajmera, J., McCowan, I. A., and Bourlard, H. (2004). *Robust Audio Segmentation*. PhD thesis, École Polytechnique Férale de Lausanne, Switzerland.

[5] Aldoma, A., Vincze, M., Blodow, N., Gossow, D., Gedikli, S., Rusu, R., and Bradski, G. (2011). Cad-model recognition and 6dof pose estimation using 3d cues. In *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 585 –592.

[6] Anguera, X. and Pardo, J. (2006). Robust Speaker Diarization for Meetings: ICSI RT06S Evaluation System. In *ICSLP*. Springer Verlag.

[7] Aran, O. and Gatica-Perez, D. (2013). One of a Kind: Inferring Personality Impressions in Meetings. In *International Conference on Multimodal Interaction*, ICMI '13, pages 11–18, New York, NY, USA. ACM.

[8] Ayadi, M. E., Kamel, M. S., and Karray, F. (2011). Survey on speech emotion recognition: Features, classification schemes, and databases. *Pattern Recognition*, 44(3):572–587.

[9] Ba, S. and Odobez, J.-M. (2009). Recognizing Visual Focus of Attention From Head Pose in Natural Meetings. *SMC-B*, 39(1):16–33.

[10] Baeza-Yates, R. and Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. Addison-Wesley.

[11] Bautista, M. A., Hernández-Vela, A., Ponce, V., Perez-Sala, X., Baró, X., Pujol, O., Angulo, C., and Escalera, S. (2013). Probability-based dynamic time warping for gesture recognition on rgb-d data. In *Advances in Depth Image Analysis and Applications*, volume 7854 of *Lecture Notes in Computer Science*, pages 126–135.

[12] Bekkerman, R. and Allan, J. (2004). Using bigrams in text categorization. In *Tech. Report, Department of Computer Science, University of Massachusetts, Amherst, 1003:1-2*.

[13] Beven, J. P., Hall, G., Froyland, I., Steels, B., and Goulding, D. (2005). Restoration or renovation? evaluating restorative justice outcomes. *Psychiatry, Psychology and Law*, 12(1):194–206.

[14] Biel, J.-I. and Gatica-Perez, D. (2011). VlogSense: Conversational Behavior and Social Attention in Youtube. *ACM TOMCCAP*, 7S(1):33:1–33:21.

[15] Biswas, K. K. and Basu, S. (2011). Gesture recognition using microsoft kinect. In *2011 5th International Conference on Automation, Robotics and Applications (ICARA)*, pages 100–103.

[16] Bo, L., Ren, X., and Fox, D. (2011). Depth kernel descriptors for object recognition. In *IROS*, pages 821–826.

[17] Bobick, A. and Davis, J. (2001). The recognition of human movement using temporal templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(3):257–267.

[18] Bobick, A. and Wilson, A. (1997). A state-based approach to the representation and recognition of gestures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(12):1325–1337.

[19] Bosch, A., Zisserman, A., and Munoz, X. (2007). Image classifcation using random forests and ferns. In *Proceedings of ICCV*.

[20] Brendel, W. and Todorovic, S. (2011). Learning spatiotemporal graphs of human activities. In *International Conference on Computer Vision*.

[21] Chaaraoui, A. and Florez-Revuelta, F. (2014). Adaptive human action recognition with an evolving bag of key poses. *IEEE Transactions on Autonomous Mental Development*, 6(2):139–152.

[22] Chaaraoui, A., Padilla, J. R., Climent-Perez, P., and Florez-Revuelta, F. (2014). Evolutionary joint selection to improve human action recognition with rgb devices. *Expert systems with applications*, 41:786–794.

[23] Chan, C. S., Liu, H., and Brown, D. J. (2007). Recognition of human motion from qualitative normalised templates. *Journal of Intelligent and Robotic Systems*, 48:79–95.

[24] Chang, C.-C. and Lin, C.-J. (2011). LIBSVM: A Library for Support Vector Machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27.

[25] Chang, K. W. and Roth, D. (2011). Selective block minimization for faster convergence of limited memory large-scale linear models. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

[26] Cheung, G., Kanade, T., Bouguet, J.-Y., and Holler, M. (2000). A real time system for robust 3d voxel reconstruction of human motions. In *Proceedings. IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 714–720 vol.2.

[27] Csurka, G., Dance, C., Fan, L., Willamowski, J., and Bray, C. (2004). Visual categorization with bags of keypoints. In *International Workshop on Statistical Learning in Computer Vision, ECCV*, pages 1–22.

[28] Cummins, R. and O'Riordan, C. (2006). Evolving local and global weighting schemes in information retrieval. *Information Retrieval*, 9:311–330.

[29] D. Gehrig, H. Kuehne, A. W. T. S. (2009). Hmm-based human motion recognition with optical flow data. In *IEEE International Conference on Humanoid Robots (Humanoids 2009)*, Paris, France.

[30] Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. *CVPR*, 2:886–893.

[31] Dalal, N., Triggs, B., and Schmid, C. (2006). Human detection using oriented histograms of flow and appearance. In *Proceedings of the 9th European Conference on Computer Vision - Volume Part II*, ECCV'06, pages 428–441, Berlin, Heidelberg. Springer-Verlag.

[32] Darrell, T., Gordon, G., Harville, M., and Woodfill, J. (1998). Integrated person tracking using stereo, color, and pattern detection. In *IEEE Proceedings Computer Society Conference on Computer Vision and Pattern Recognition*, pages 601–608.

[33] Debole, F. and Sebastiani, F. (2003). Supervised term-weighting for automated text categorization. In *Proceedings of the 2003 ACM Symposium on Applied Computing*, SAC '03, pages 784–788, New York, NY, USA. ACM.

[34] Deléglise, P., Estève, Y., Meignier, S., and Merlin, T. (2005). The LIUM speech transcription system: a CMU Sphinx III-based system for French broadcast news. In *Interspeech*.

[35] Demsar, J. (2006). Statistical comparisons of classifiersover multiple data sets. *Journal of Machine Learning Research*, 7:1–30.

[36] Deselaers, T., Pimenidis, L., and Ney, H. (2008). Bag of visual words for adult image classification and filtering. In *Proceedings of the International Conference on Pattern Recognition*. IEEE.

[37] Djuric, N., Lan, L., Vucetic, S., and Wang, Z. (2013). Budgetedsvm: A toolbox for scalable svm approximations. *Journal of Machine Learning Research*, 14:3813–3817.

[38] Doliotis, P., Stefan, A., McMurrough, C., Eckhard, D., and Athitsos, V. (2011). Comparing gesture recognition accuracy using color and depth information. In *Proceedings of the 4th International Conference on PErvasive Technologies Related to Assistive Environments*, PETRA '11, pages 20:1–20:7.

[39] Dovidio, J. and Ellyson, S. (1982). Decoding Visual Dominance: Attributions of Power Based on Relative Percentages of Looking While Speaking and Looking While Listening. In *SPQ*, pages 106–113. American Sociological Association.

[40] Escalante, H. J., Garcia, M., Morales, A., Graff, M., Montes, M., Morales, E. F., and Martinez, J. (2015a). Term-weighting learning via genetic programming for text classification. *Knowledge-based Systems*, Online.

[41] Escalante, H. J., Guyon, I., Vassilis, A., Jangyodsuk, P., and Wan, J. (2015b). Principal motion components for one-shot gesture recognition. *Pattern Analysis and Applications*, pages 1–16.

[42] Escalante, H. J., Martinez-Carranza, J., Escalera, S., Ponce-López, V., and Baró, X. (2015c). Improving bag of visual words representations with genetic programming. In *Proceedings of the 2015 International Joint Conference on Neural Networks, IJCNN2015*, pages 3674–3681. IEEE.

[43] Escalante, H. J., Ponce-López, V., Escalera, S., Baró, X., Morales-Reyes, A., and Martinez-Carranza, J. (2015d). Evolving weighting schemes for the bag of visual words. *Submitted to Neural Computing and Applications*.

[44] Escalera, S., Baró, X., Gonzalez, J., Bautista, M. A., Madadi, M., Reyes, M., Ponce-López, V., Escalante, H. J., Shotton, J., and Guyon, I. (2014). ChaLearn looking at people challenge 2014: Dataset and results. In *Proc. of European Conference on Computer Vision - Chalearn workshop*.

[45] Escalera, S., Baró, X., Vitrià, J., Radeva, P., and Raducanu, B. (2012). Social Network Extraction and Analysis Based on Multimodal Dyadic Interaction. *Sensors*, 12(2):1702–1719.

[46] Escalera, S., Pujol, O., and Radeva, P. (2010a). Error-Correcting Output Codes Library. *JMLR*, 11:661–664.

[47] Escalera, S., Pujol, O., Radeva, P., Vitrià, J., and Anguera, M. T. (2010b). Automatic Detection of Dominance and Expected Interest. *EURASIP Advances in Signal Processing, Research Article*.

[48] Everingham, M., Gool, L., Williams, C. K., Winn, J., and Zisserman, A. (2010). The PASCAL Visual Object Classes (VOC) Challenge. *IJCV*, 88(2):303–338.

[49] Eweiwi, A., Cheema, M. S., Bauckhage, C., and Gall, J. (2015). Efficient pose-based action recognition. In *Proceedings of Assian Conference on Computer Vision, LNCS*, volume 9007, pages 428–443.

[50] Fang, G., Gao, W., and Zhao, D. (2007). Large-vocabulary continuous sign language recognition based on transition-movement models. *SMC-A*, 37(1):1–9.

[51] Fei-Fei, L., Fergus, R., and Perona, P. (2004). Learning generative visual models from few training examples: an incremental bayesian approach tested on 101 object categories. In *IEEE Proc. CVPRW*.

[52] Felzenszwalb, P., Girshick, R., McAllester, D., and Ramanan, D. (2010). Object detection with discriminatively trained part based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645.

[Fisher] Fisher, M. Interpreting Sensor Values. http://graphics.stanford.edu/~mdfisher/Kinect.html.

[54] Forman, G. (2003). An extensive empirical study of feature selection metrics for text classification. *J. of Mach. Learn. Res.*, 3:1289–1305.

[55] Freund, Y. and Schapire, R. E. (1996). Experiments with a New Boosting Algorithm. In *ICML*, pages 148–156. Morgan Kaufmann.

[56] Friedman, J., Hastie, T., and Tibshirani, R. (1998). Additive logistic regression: a statistical view of boosting. *Annals of Statistics*, 28:2000.

[57] García-Limón, M., Escalante, H. J., y Gómez, M. M., Morales, A., and Morales, E. (2014). Towards the automated generation of term-weighting schemes for text categorization. In *Proc. of GECCO Comp'14, (Late-breaking abstract)*, pages 1459–1460.

[58] Girshick, R., Shotton, J., Kohli, P., Criminisi, A., and Fitzgibbon, A. (2011). Efficient regression of general-activity human poses from depth images. In *Proceedings of the 2011 International Conference on Computer Vision*, ICCV '11, pages 415–422, Washington, DC, USA. IEEE Computer Society.

[59] Goldberg, D. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1st edition.

[60] Gonzalez-Gurrola, L. C., Moreno, R., Escalante, H. J., Martínez, F., and Carlos, R. (2015). Learning roadway surface disruption patterns using the bag of words representation. *IEEE Transactions on Intelligent Transportation Systems*, Submitted.

[61] Gopikrishnan, M. and Santhanam, T. (2011). Effect of Different Neural Networks on the Accuracy in Iris Patterns Recognition. In *IJRIC*, pages 22–28 vol.7.

[62] Grauman, K. and Leibe, B. (2010). *Visual Object Recognition*. Morgan and Claypool.

[63] Guyon, I., Athitsos, V., Jangyodsuk, P., and Escalante, H. J. (2014). The chalearn gesture dataset (cgd 2011). *Machine Vision and Applications*, 25(8):1929–1951.

[64] Hampapur, A., Brown, L., Connell, J., Ekin, A., Haas, N., Lu, M., Merkl, H., and Pankanti, S. (2005). Smart video surveillance: exploring the concept of multiscale spatiotemporal tracking. *Signal Processing Magazine, IEEE*, 22(2):38–51.

[65] HD. Yang, S. L. (2007). Reconstruction of 3d human body pose from stereo image sequences based on top-down learning. *Pattern Recognition*, 40(11):3120–3131.

[66] Hernandez-Vela, A., Bautista, M., Perez-Sala, X., Ponce, V., Baro, X., Pujol, O., Angulo, C., and Escalera, S. (2012). BoVDW: Bag-of-visual-and-depth-words for gesture recognition. *21st International Conference on Pattern Recognition (ICPR)*, pages 449–452.

[67] Hernández-Vela, A., Zlateva, N., Marinov, A., Reyes, M., Radeva, P., Dimov, D., and Escalera, S. (2012). Human limb segmentation in depth maps based on spatio-temporal graph-cuts optimization. *Journal of Ambient Intelligence and Smart Environments*, 4(6):535–546.

[68] Hernández-Vela, A., Bautista, M.-A., Perez-Sala, X., Ponce-López, V., Escalera, S., Baró, X., Pujol, O., and Angulo, C. (2014). Probability-based Dynamic Time Warping and Bag-of-Visual-and-Depth-Words for Human Gesture Recognition in RGB-D. *Pattern Recognition Letters*, 50:112 – 121. Depth Image Analysis.

[69] Hernández-Vela, A., Reyes, M., Ponce, V., and Escalera, S. (2012). Grabcut-based human segmentation in video sequences. *Sensors*, 12(11):15376.

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

[71] Hoai, M., Lan, Z.-Z., and De la Torre, F. (2011). Joint segmentation and classification of human actions in video. In *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3265–3272.

[72] Huang, D., Yao, S., Wang, Y., and De La Torre, F. (2014). Sequential max-margin event detectors. In *Computer Vision – ECCV 2014*, volume 8691 of *Lecture Notes in Computer Science*, pages 410–424. Springer International Publishing.

[73] Jain, H. and Subramanian, A. (2010). Real-time upper-body human pose estimation using a depth camera. *HP Technical Reports*, 1(190).

[74] Jayagopi, D. B. and Gatica-Perez, D. (2010). Mining Group Nonverbal Conversational Patterns Using Probabilistic Topic Models. *IEEE Trans. on Multimedia*, 12(8):790–802.

[75] Jones, M. and Rehg, J. (1999). Statistical color models with application to skin detection. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 1, page 280 Vol. 1.

[76] Keskin, C., Kıraç, F., Kara, Y. E., and Akarun, L. (2013). Real time hand pose estimation using depth sensors. In *Consumer Depth Cameras for Computer Vision*, pages 119–137. Springer.

[77] Kim, D., Song, J., and Kim, D. (2007). Simultaneous gesture segmentation and recognition based on forward spotting accumulative hmms. *Pattern Recognition*, 40(11):3012–3026.

[78] Kläser, A., Marszałek, M., and Schmid, C. (2008). A spatio-temporal descriptor based on 3d-gradients. In *British Machine Vision Conference*, pages 995–1004.

[79] Knapp, M. and Hall, J. (1997). *Nonverbal Communication in Human Interaction*. Harcourt Brace College Publishers.

[80] Lai, K., Bo, L., Ren, X., and Fox, D. (2011). Sparse distance learning for object recognition combining rgb and depth information. In *ICRA*, pages 4007 –4013.

[81] Lan, M., Tan, C. L., Su, J., and Lu, Y. (2009). Supervised and traditional term-weighting methods for automatic text categorization. *Trans. PAMI*, 31(4):721–735.

[82] Langdon, W. B. and Poli, R. (2001). *Foundations of Genetic Programming*. Springer.

[83] Laptev, I. (2005). On space-time interest points. *International Journal of Computer Vision*, 64(2-3):107–123.

[84] Laptev, I., Marszalek, M., Schmid, C., and Rozenfeld, B. (2008). Learning realistic human actions from movies. In *Computer Vision and Pattern Recognition*, pages 1–8.

[85] Lazebnik, S., Schmid, C., and Ponce, J. (2004). Semi-local affine parts for object recognition. In *British Machine Vision Conference*, pages 779–788.

[86] Lazebnik, S., Schmid, C., and Ponce, J. (2006). Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Proceedings of the Computer Vision and Image Processing Conference*, pages 2169–2178. IEEE.

[87] Lazebnik, S., Schmid, C., and Ponce, J. (2015). Maximum entropy framework for part-based texture and object recognition. In *IEEE International Conference on Computer Vision*, pages 832–838.

[88] Li, K., Hu, J., and Fu, Y. (2012). Modeling complex temporal composition of actionlets for activity prediction. In Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., and Schmid, C., editors, *European Conference on Computer Vision*, volume 7572 of *Lecture Notes in Computer Science*, pages 286–299. Springer Berlin Heidelberg.

[89] Li, W., Zhang, Z., and Liu, Z. (2010). Action recognition based on a bag of 3D points. In *Computer Vision and Pattern Recognition Workshops*, pages 9–14.

[90] Li, Y. (2012). Hand gesture recognition using kinect. In *Software Engineering and Service Science (ICSESS), 2012 IEEE 3rd International Conference on*, pages 196–199.

[91] Liu, L. and Shao, L. (2013). Learning discriminative representations from rgb-d data. In *International Joint Conference on Artificial Intelligence*.

[92] Liu, L., Shao, L., and Rockett, P. (2012). Genetic programming-evolved spatio-temporal descriptor for human action recognition. In *Proceedings of the British Machine Vision Conference*, pages 18.1–18.12. BMVA Press.

[93] Liu, L., Shao, L., and Rockett, P. (2013). Boosted key-frame selection and correlated pyramidal motion-feature representation for human action recognition. *Pattern recognition*, 46(7):1810–1818.

[94] Liu Y., Stoll C., G. J. S. H.-P. and C., T. (2011). Markerless motion capture of interacting characters using multi-view image segmentation. *CVPR*, 14(1):1249–1256.

[95] Lopes, O., Reyes, M., Escalera, S., and Gonzalez, J. (2014). Spherical Blurred Shape Model for 3-D Object and Pose Recognition: Quantitative Analysis and HCI Applications in Smart Environments. *SMC-B*, PP(99):1.

[96] Lopez-Monroy, A. P., y Gomez, M. M., Escalante, H. J., Cruz-Roa, A., and Gonzalez, F. A. (2015). Improving the bovw with discriminative n-grams and mkl. *Neurocomputing*, Accepted.

[97] Luke, S. and Panait, L. (2002). Lexicographic parsimony pressure. In *Proceedings of GECCO*, pages 829–836.

[98] Lv, F. and Nevatia, R. (2006). Recognition and segmentation of 3-d human action using hmm and multi-class adaboost. In *European Conference on Computer Vision*, pages 359–372.

[99] Malgireddy, M. R., Corso, J. J., Setlur, S., Govindaraju, V., and Mandalapu, D. (2010). A framework for hand gesture recognition and spotting using sub-gesture modeling. In *International Conference on Pattern Recognition*.

[100] Malgireddy, M. R., Nwogu, I., Ghosh, S., and Govindaraju, V. (2011). A shared parameter model for gesture and sub-gesture analysis. In *Combinatorial Image Analysis*, volume 6636, pages 483–493.

[101] Manchala, S., Prasad, V. K., and Janaki, V. (2014). Gmm based language identification system using robust features. *International Journal of Speech Technology*, 17:99–105.

[102] Marcos-Ramiro, A., Pizarro-Perez, D., Marron-Romera, M., Nguyen, L. S., and Gatica-Perez, D. (2013). Body communicative cue extraction for conversational analysis. *Proc. of IEEE International Conference on Automatic Face and Gesture Recognition*.

[103] Marin-Jimenez, M., Zisserman, A., Eichner, M., and Ferrari, V. (2014). Detecting People Looking at Each Other in Videos. *IJCV*, 106(3):282–296.

[104] McNeill, D. (1992). *Hand and Mind: What Gestures Reveal about Thought*. University of Chicago Press, Chicago.

[105] McNeill, D. (2005). *Gesture and Thought*. University of Chicago Press, Chicago.

[106] Mehrabian, A. (1972). *Nonverbal communication*. Aldine-Atherton.

[107] Mirza-Mohammadi, M., S.Escalera, and Radeva, P. (2009). Contextual-guided bag-of-visual-words model for multi-class object categorization. In *Proc. of CAIP*, pages 748–756. Springer.

[108] Mitra, S. and Acharya, T. (2007). Gesture recognition: A survey. *Trans. on Systems, Man, and Cybernetics, C*, 37(3):311–324.

[109] Moeslund, T., Hilton, T., Kruger, A., and Sigal, V. (2011). *Visual Analysis of Humans, Looking at People*. Springer.

[110] Mohammadi, G., Park, S., Sagae, K., Vinciarelli, A., and Morency, L.-P. (2013). Who is Persuasive?: The Role of Perceived Personality and Communication Modality in Social Multimedia. In *International Conference on Multimodal Interaction*, ICMI '13, pages 19–26, New York, NY, USA. ACM.

[111] Mortensen, E. N., Deng, H., and Shapiro, L. (2005). A sift descriptor with global context. *CVPR*, 1:184–190 vol. 1.

[112] Muller, M. and Roder, T. (2006). Motion templates for automatic classification and retrieval of motion capture data. In *ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 137–146.

[113] Neverova, N., Wolf, C., Taylor, G. W., and Nebout, F. (2014). Multi-scale deep learning for gesture detection and localization. In *Proc. of ECCV ChaLearn Workshop on Looking at People*.

[114] Niebles, J. C., Wang, H., and Fei-Fei, L. (2008). Unsupervised learning of human action categories using spatial-temporal words. *IJCV*, 79(3):299–318.

[115] Padilla-López, J. R., Chaaraoui, A. A., and Flórez-Revuelta, F. (2014). A discussion on the validation tests employed to compare human action recognition methods using the MSR action3d dataset. *CoRR*, abs/1407.7390.

[116] Parizeau, M. and Plamondon, R. (1990). A comparative analysis of regional correlation, dynamic time warping, and skeletal tree matching for signature verification. *IEEE TPAMI*, 12(7).

[117] Pedersoli, F., Adami, N., Benini, S., and Leonardi, R. (2012). Xkin -: extendable hand pose and gesture recognition library for kinect. In *ACM Multimedia*, pages 1465–1468.

[118] Pentland, A. S. (2005). Socially Aware Computation and Communication. *Computer*, 38(3):33–40.

[119] Pentland, A. S. (2008). *Honest Signals: How They Shape Our World*. The MIT Press, Massachusetts.

[120] Ponce, V., Escalera, S., and Baró, X. (2012a). Social signal analysis in criminal mediation processes. *Advances in Theory and Applications of Computer Vision, CVCR&D*.

[121] Ponce, V., Escalera, S., Baró, X., and Radeva, P. (2012b). Automatic analysis of non-verbal communication. *Achievements and New Opportunities in Computer Vision, CVCR&D Workshop 2010*, pages 105–108.

[122] Ponce, V., Gorga, M., Baro, X., and Escalera, S. (2011a). Human behavior analysis from video data using bag-of-gestures. In *International Joint Conference on Artificial Intelligence*, pages 2836–2837. AAAI Press.

[123] Ponce, V., Gorga, M., Baró, X., Radeva, P., and Escalera, S. (2011b). Análisis de la expresión oral y gestual en proyectos fin de carrera vía un sistema de visión artificial. *ReVisión*, 4(1).

[124] Ponce, V., Reyes, M., Baró, X., Gorga, M., and Escalera, S. (2011c). Two-level gmm clustering of human poses for automatic human behavior analysis. *State of the Art of Research and Development in Computer Vision, CVCR&D Workshop*, pages 47–50.

[125] Ponce-López, V., Escalera, S., and Baró, X. (2013). Multi-modal social signal analysis for predicting agreement in conversation settings. In *Proceedings of the 15th ACM on International Conference on Multimodal Interaction*, pages 495–502, New York, NY, USA. ACM.

[126] Ponce-López, V., Escalera, S., Pérez, M., Janés, O., and Baró, X. (2014). Non-verbal communication analysis in victim-offender mediations. *CoRR*, abs/1412.2122.

[127] Ponce-López, V., Escalante, H. J., Escalera, S., and Baró, X. (2015a). Gesture and action recognition by evolved dynamic subgestures. In Xianghua Xie, M. W. J. and Tam, G. K. L., editors, *Proceedings of the British Machine Vision Conference (BMVC)*, pages 129.1–129.13. BMVA Press.

[128] Ponce-López, V., Escalera, S., Pérez, M., Janés, O., and Baró, X. (2015b). Non-verbal communication analysis in victim-offender mediations. *Pattern Recognition Letters*, 67, Part 1:19 – 27. Cognitive Systems for Knowledge Discovery.

[129] Rabiner, L. and Juang, B.-H. (1993). *Fundamentals of Speech Recognition*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

[130] Rahmani, H., Mahmood, A., Huynh, D., and Mian, A. (2014). HOPC: Histogram of oriented principal components of 3d pointclouds for action recognition. In *European Conference on Computer Vision, LNCS*, volume 8690, pages 742–757.

[131] Raptis, M., Kokkinos, I., and Soatto, S. (2012). Discovering discriminative action parts from mid-level video representations. In *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1242–1249.

[132] Raptis, M. and Sigal, L. (2013). Poselet key-framing: A model for human activity recognition. In *Computer Vision and Pattern Recognition*, pages 2650–2657, Washington, DC, USA. IEEE Computer Society.

[133] Reyes, M., Dominguez, G., and Escalera, S. (2011). Feature weighting in dynamic time warping for gesture recognition in depth data. In *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pages 1182–1188.

[134] Rodgers, J., Anguelov, D., Hoi-Cheung, P., and D., K. (2006). Object pose detection in range scan data. *CVPR*, pages 2445–2452.

[135] Rudovic, O., Pantic, M., and Patras, I. (2013). Coupled Gaussian Processes for Pose-Invariant Facial Expression Recognition. *IEEE TPAMI*, 35(6):1357–1369.

[136] Rusu, R., Bradski, G., Thibaux, R., and Hsu, J. (2010). Fast 3d recognition and pose using the viewpoint feature histogram. In *Proceedings of the 23rd IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2155 –2162, Taipei, Taiwan.

[137] Rusu, R. B., Blodow, N., and Beetz, M. (2009). Fast point feature histograms (fpfh) for 3d registration. In *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, pages 3212 –3217, Kobe, Japan.

[138] Rusu, R. B. and Cousins, S. (2011). 3D is here: Point Cloud Library (PCL). In *IEEE ICRA*, Shanghai, China.

[139] Sabata, B., Arman, F., and Aggarwal, J. (1993). Segmentation of 3d range images using pyramidal data structures,. *CVGIP: Image Understanding*, 57(3):373–387.

[140] Saffari, A. and Guyon, I. (2006). Quick start guide for clop. Technical report, TU Graz - CLOPINET.

[141] Salton, G. and Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Inform. Process. Manag.*, pages 513–523.

[142] Sanchez-Cortes, D., Aran, O., Jayagopi, D., Schmid Mast, M., and Gatica-Perez, D. (2013). Emergent leaders through looking and speaking: from audio-visual data to multimodal recognition. *JMUI*, 7(1-2):39–53.

[143] Sanchez-Cortes, D., Aran, O., Schmid Mast, M., and Gatica-Perez, D. (2012). A Nonverbal Behavior Approach to Identify Emergent Leaders in Small Groups. *IEEE Trans. on Multimedia*, 14(3):816–832.

[144] Sebastiani, F. (2008). Machine learning in automated text categorization. *ACM Computer Surveys*, 34(1):1–47.

[145] Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., and Blake, A. (2011). Real-time human pose recognition in parts from single depth images. In *CVPR*, pages 1297–1304.

[146] Silva, S. and Almeida, J. (2003). Gplab-a genetic programming toolbox for matlab. In *Proc. Nordic MATLAB conf.*, pages 273–278.

[147] Sivic, J. and Zisserman, A. (2003). Video google: A text retrieval approach to object matching in videos. In *International Conference on Computer Vision*, volume 2, pages 1470–1477.

[148] Sminchisescu, C., Kanaujia, A., and Metaxas, D. (2006). Conditional models for contextual human motion recognition. *CVIU*, 104(2-3):210–220.

[149] Specht, D. F. (1988). Probabilistic neural networks for classification, mapping, or associative memory. In *IEEE IJCNN*, pages 525–532 vol.1.

[150] Starner, T. and Pentland, A. (1995a). Real-time american sign language recognition from video using hidden markov models. In *Computer Vision, 1995. Proceedings., International Symposium on*, pages 265–270, Perceptual Comput. Sect., MIT, Cambridge.

[151] Starner, T. and Pentland, A. (1995b). Visual recognition of american sign language using hidden markov models. In *Proc. Int. Workshop Automatic Face and Gesture Recognition*.

[152] Stefan, A., Athitsos, V., Alon, J., and Sclaroff, S. (2008). Translation and Scale-Invariant Gesture Recognition in Complex Scenes. In *PETRA*, pages 7:1–7:8.

[153] Takahashi, T. and Kishino, F. (1991). Hand gesture coding based on experiments using a hand gesture interface device. *SIGCHI Bull.*, 23(2):67–74.

[154] Tirilly, P., Claveau, V., and Gros, P. (2009). A review of weigthing schemes for bag of visual words image retrieval. Technical report, IRISA.

[155] Trujillo, L. and Olague, G. (2006). Synthesis of interest point detectors through genetic programming. In *Conference on Genetic and Evolutionary Computation*, GECCO, pages 887–894, New York, NY, USA. ACM.

[156] Turney, P. and Pantel, P. (2010). From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research*, 37:141–188.

[157] Umbreit, M. S. (2002). *The Handbook of Victim Offender Mediation: An Essential Guide to Practice and Research*. John Wiley & Sons.

[158] V. Ganapathi, V., Plagemann, C., Koller, D., and Thrun, S. (2010). Real time motion capture using a single time-of-flight camera. *CVPR*, pages 755–762.

[159] Vedaldi, A. and Fulkerson, B. (2008). VLFeat: An open and portable library of computer vision algorithms.

[160] Vinciarelli, A., Pantic, M., and Bourlard, H. (2009). Social signal processing: Survey of an emerging domain. *Image and Vision Computing*, 27(12):1743–1759.

[161] Vinciarelli, A., Salamin, H., and Pantic, M. (2009). Social Signal Processing: Understanding Social Interactions through Nonverbal Behaviour Analysis. In *CVPR*, volume 3, pages 42–49.

[162] Viola, P. and Jones, M. J. (2004). Robust real-time face detection. *International Journal of Computer Vision*, 57(2):137–154.

[163] W. Schwartz, A. Kembhavi, D. H. L. D. (2009). Human detection using partial least squares analysis. ICCV.

[164] Wagner, P. K., Peres, S. M., Madeo, R. C. B., Lima, C. A. M., and Freitas, F. A. (2014). Gesture unit segmentation using spatial-temporal information and machine learning. In *Florida Artificial Intelligence Research Society Conference*.

[165] Wan, T., Wang, Y., and Li, J. (2012). Hand gesture recognition system using depth data. In *Consumer Electronics, Communications and Networks (CECNet), 2012 2nd International Conference on*, pages 1063–1066.

[166] Wang, J., Liu, P., M., F. H. S., Nahavandi, S., and Kouzani, A. (2013). Bag-of-words representation for biomedical time series classification. *Biomed Signal Process Control*, 8(6):634–644.

[167] Wang, J., Liu, Z., Wu, Y., and Yuan, J. (2012). Mining actionlet ensemble for action recognition with depth cameras. In *Computer Vision and Pattern Recognition*, pages 1290–1297.

[168] Wang, J., Liu, Z., Wu, Y., and Yuan, J. (2014a). Learning actionlet ensemble for 3d human action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(5):914–927.

[169] Wang, J. and Wu, Y. (2013). Learning maximum margin temporal warping for action recognition. In *Proceedings of International Conference on Computer Vision*, pages 2688–2695.

[170] Wang, L., Qiao, Y., , and Tang, X. (2014b). Video action detection with relational dynamic-poselets. In *European Conference on Computer Vision*.

[171] wei Chen, Y. (2005). Combining SVMs with Various Feature Selection Strategies. In *Taiwan University*. Springer-Verlag.

[172] Weitekamp, E. (1993). Reparative justice. *European Journal on Criminal Policy and Research*, 1(1):70–93.

[173] Wilson, A. and Bobick, A. (1999). Parametric hidden markov models for gesture recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(9):884–900.

[174] Wren, C., Azarbayejani, A., Darrell, T., and Pentland, A. (1997). Pfinder: Real-Time Tracking of the Human Body. *PAMI*, 19(7):780–785. IEEE TPAMI.

[175] Xia, L. and Aggarwal, J. K. (2013). Spatio-temporal depth cuboid similarity feature for activity recognition using depth camera. In *Computer Vision and Pattern Recognition*, pages 2834–2841.

[176] Y. Zhu, B. D. and Fujimura, K. (2008). Controlled human pose estimation from depth image streams. *CVPR Workshop on TOF Computer Vision*.

[177] Yang, H.-D., Sclaroff, S., and Lee, S.-W. (2009). Sign Language Spotting with a Threshold Model Based on Conditional Random Fields. *IEEE TPAMI*, 31(7):1264–1277.

[178] Yoo, S. J. (2004). Intelligent multimedia information retrieval for identifying and rating adult images. In *Proceedings of the International Conference KES*, volume 3213 of *LNAI*, pages 164–170. Springer.

[179] Zhang, J., Marszablek, M., Lazebnik, S., and Schmid, C. (2007). Local features and kernels for classification of texture and object categories: A comprehensive study. *International Journal of Computer Vision*, 73(2):213–238.

[180] Zhang, K., Lan, L., Wang, Z., and Moerchen, F. (2012). Scaling up kernel svm on limited resources: A low-rank linearization approach. In *Proc. of AISTATS 2012*.

[181] Zhao, Z. and Elgammal, A. M. (2008). Information theoretic key frame selection for action recognition. In *British Machine Vision Conference*.

[182] Zhou, F., De la Torre, F., and Hodgins, J. K. (2008). Aligned cluster analysis for temporal segmentation of human motion. In *IEEE Conference on Automatic Face and Gestures Recognition (FG)*.

[183] Zhou, F., De la Torre Frade, F., and Hodgins , J. K. (2013). Hierarchical aligned cluster analysis for temporal clustering of human motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(3):582–596.

[184] Zhu, X. and Ramanan, D. (2012). Face detection, pose estimation, and landmark localization in the wild. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2879–2886.

Appendix A

Publications

This Appendix provides the list of publications of the author related to the work presented in this thesis.

A.1 Journal papers

[43] H.J. Escalante, V. Ponce-López, S. Escalera, X. Baró, A. Morales-Reyes and J. Martinez-Carranza. *Evolving weighting schemes for the Bag of Visual Words*. Under Revision for Neural Computing and Applications, Special Issue on Computational Intelligence for Vision and Robotics.

[68] A. Hernandez-Vela, M. A. Bautista, X. Perez-Sala, V. Ponce-López, X. Baro, O. Pujol, C. Angulo, and S. Escalera. *Probability-based Dynamic Time Warping and Bag-of-Visual-and-Depth-Words for Human Gesture Recognition in RGB-D*. Pattern Recognition Letters, 2013, ISSN 0167-8655.

[128] V. Ponce-López, S. Escalera, M. Perez, O. Janés, and X. Baró. *Non-Verbal Communication Analysis in Victim-Offender Mediations*. Pattern Recognition Letters, Special Issue on Cognitive Systems for Knowledge Discovery, ISSN 0167-8655, vol. 67, Part 1, pp. 19-27, 2015.

[69] A. Hernández-Vela, M. Reyes, V. Ponce, S. Escalera. *GrabCut-Based Human Segmentation in Video Sequences*. Sensors 2012, ISSN 1424-8220, vol. 12, num. 11, pp. 15376-15393.

A.2 Proceedings in international Conferences and Workshops

[127] V. Ponce-López, H.J. Escalante, S. Escalera, X. Baró. *Gesture and Action Recognition by Evolved Dynamic Subgestures*. Proceedings of the British Machine Vision Conference (BMVC), pp. 129.1-129.13, 2015.

[42] H.J. Escalante, J. Martinez, S. Escalera, V. Ponce-López and X. Baró. *Improving Bag of Visual Words Representations with Genetic Programming*. IJCNN 2015.

[44] S. Escalera, X. Baró. J. Gonzàlez, M. Bautista, M. Madadi, M. Reyes, V. Ponce-López, H.-J Escalante, J. Shotton, I. Guyon. *ChaLearn Looking at People Challenge 2014: Dataset and Results*. ECCV ChaLearn LAP Workshop, vol. 1, pp. 459-473, 2014.

[125] V. Ponce-López, S. Escalera, and X. Baró. *Multi-modal Social Signal Analysis for Predicting Agreement in Conversation Settings*. International Conference on Multimodal Interaction, 2013, Sydney, ISBN 978-1-4503-2129-7, pp. 495-502.

[66] A. Hernández, M. A. Bautista, X. Pérez, V. Ponce, X. Baró, O. Pujol, C. Angulo, and S. Escalera. *BoVDW: Bag-of-Visual-and-Depth-Words for Gesture Recognition*. 21st International Conference on

Pattern Recognition, 2012, Japan, ISSN 1051-4651, pp. 449-452.

[11] M. A. Bautista, A. Hernández, V. Ponce, X. Pérez, X. Baró, O. Pujol, C. Angulo, and S. Escalera. *Probability-based Dynamic Time Warping for Gesture Recognition*. International Workshop on Depth Image Analysis, ICPR 2012, Japan, ISSN 0302-9743, vol. 7854, pp. 126-135.

[122] V. Ponce, M. Gorga, X. Baró, S. Escalera. *Human Behavior Analysis from Video Data using Bag-of-Gestures*. International Joint Conference on Artificial Intelligence, 2011, ISBN 978-1-57735-515-1 vol. 3, pp. 2836-2837.

A.3 Non-indexed publications

[126] V. Ponce-López, S. Escalera, M. Perez, O. Janés, and X. Baró. *Non-Verbal Communication Analysis in Victim-Offender Mediations*. arXiv CoRR, vol. abs/1412.2122, 2014.

[120] V. Ponce, S. Escalera, and X. Baro. *Social Signal Analysis in Criminal Mediation Processes*. Advances in Theory and Applications of Computer Vision, CVCR&D Workshop, 2012, vol. 1.

[124] V. Ponce, M. Reyes, X. Baró, M. Gorga and S. Escalera. *Two-level GMM Clustering of Human Poses for Automatic Human Behavior Analysis*. CVCR&D State of the Art of Research and Development in Computer Vision, CVCR&D Workshop, 2011, ISBN 978-84-938351-5-6, pp. 47-50.

[123] V. Ponce, M. Gorga, X. Baró, P. Radeva, and S. Escalera. *Análisis de la expresión oral y gestual en proyectos fin de carrera vía un sistema de visión artificial*. ReVisión, la revista electrónica de la Asociación de Enseñantes Universitarios de la Informática AENUI, 2011, ISSN 1989-1199, vol. 4, num. 1.

[121] V. Ponce, S. Escalera, X. Baró, and P. Radeva. *Automatic Analysis of Non-verbal Communication*. Achievements and New Opportunities in Computer Vision, CVCR&D Workshop 2010, ISBN 978-84-938351-0-1, pp. 105-108.