2013

A Methodology for Extracting Human Bodies from Still Images

Athanasios Tsitsoulis
Wright State University

A Methodology for Extracting Human Bodies from Still Images

A thesis submitted in partial fulfilment
of the requirements for the degree of
Doctor of Philosophy in Computer Engineering

By

ATHANASIOS TSITSOULIS
M.S., University of Patras, Computer Engineering and Informatics Department, 2009

2013
Wright State University
Dayton, Ohio 45435-0001

WRIGHT STATE UNIVERSITY
SCHOOL OF GRADUATE STUDIES

December 20, 2013

I HEREBY RECOMMEND THAT THE THESIS PREPARED UNDER MY SUPERVISION BY `Athanasios Tsitsoulis` ENTITLED `A Methodology for Extracting Human Bodies from Still Images` BE ACCEPTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF `Doctor of Philosophy in Computer Engineering`.

Nikolaos G. Bourbakis, Ph.D.
Thesis Director

Arthur Goshtasby, Ph.D.
Director, Computer Science & Engineering
Ph.D. Program

R. William Ayres, Ph.D.
Interim Dean, Graduate School

Committee on
Final Examination

Nikolaos G. Bourbakis, Ph.D.

Soon Chung, Ph.D.

Yong Pei, Ph.D.

Ioannis Hatziligeroudis, Ph.D.

ABSTRACT

Tsitsoulis, Athanasios. PhD. Department of Computer Science and Engineering, Wright State University, 2013. A Methodology for Extracting Human Bodies from Still Images

Monitoring and surveillance of humans is one of the most prominent applications of today and it is expected to be part of many future aspects of our life, for safety reasons, assisted living and many others. Many efforts have been made towards automatic and robust solutions, but the general problem is very challenging and remains still open. In this PhD dissertation we examine the problem from many perspectives. First, we study the performance of a hardware architecture designed for large-scale surveillance systems. Then, we focus on the general problem of human activity recognition, present an extensive survey of methodologies that deal with this subject and propose a maturity metric to evaluate them.

One of the numerous and most popular algorithms for image processing found in the field is image segmentation and we propose a blind metric to evaluate their results regarding the activity at local regions. Finally, we propose a fully automatic system for segmenting and extracting human bodies from challenging single images, which is the main contribution of the dissertation. Our methodology is a novel bottom-up approach relying mostly on anthropometric constraints and is facilitated by our research in the fields of face, skin and hands detection. Experimental results and comparison with state-of-the-art methodologies demonstrate the success of our approach.

Contents

List of Tables

List of Figures

ACKNOWLEDGEMENTS

I would like to express my gratitude to my advisor Dr. Nikolaos Bourbakis for giving me the opportunity to perform research work in a very interesting scientific area. In addition to that, his constant supervision and criticism have helped me become more efficient, responsible and creative. His help and support throughout this whole research will always be appreciated and have definitely helped me become a better professional.

I would also like to thank the members of my doctoral committee Dr. Yong Pei, Dr. Soon Chung and Dr. Ioannis Hatziligeroudis. Their tutoring and suggestions have greatly contributed in this work and I am very thankful to them for doing so.

Finally, I would like to thank my family and friends in Greece, who although far away, are always in my heart and thoughts. I would also like to thank my colleagues and friends Alexandros Pantelopoulos, Alexandros Karargyris, Michael Tsakalakis, Adamandia Psarologou, Kostas Michalopoulos, Robert Keefer, Ryan Patrick and Raghudeep Kannavara for our fruitful discussions and the good times we had together. Special thanks go to Anna Trikalinou for giving me strength to go through this difficult but exciting journey.

To my mother and sister,
for their unconditional love and support.

To Anna,
for being always by my side.

You don't understand anything until you learn it more than one way.

Marvin Minsky

1

Introduction

1.1 Motivation

One of the main needs for the betterment of modern societies is that of systems that can efficiently and effectively be used for monitoring and surveillance. Cameras are becoming a prevalent median because of the high amount of information they can convey. Additionally, technological advances have made possible for camera sensors to become a commodity over the last decades and gain a wide range of capabilities (high quality imaging, portability, processing power). There are numerous applications they can facilitate, such as observation of traffic flow and recording of accidents, recording of transactions in ATMs and incidents in banks, surveillance of security sensitive areas and monitoring people at risk in care centers, to name a few and thus, they are becoming part of our everyday life.

However, monitoring systems still rely heavily on human operators, which is a costly and inefficient solution, let alone in large-scale systems, as the ones needed for example for coverage of large areas. It is clear that in order for this field to reach its full potential solutions have to be given that allow computers to "see", so that they can process the raw stream of visual data and extract higher level knowledge from it. High-level processes such as video summarization and categorization, object recognition, action recognition and event analysis can aid the management of the vast amounts of imaging data that keep growing. The next step is to move to systems and methods that can use these data to successfully predict events and not just report them or keep evidence of their occurrence.

Another issue that rises when monitoring systems increase in scale and perform complex operations and reasoning is the computational demand they require. Transmittance, storage and processing require vast amounts of resources and processing power, which is not only expensive but in some cases might be even infeasible. Thus, besides efficient methodologies there should be pow-

erful architectures too to cope with the problem. Architectures adapted to the needs of surveillance systems should be able to handle heavy loads of data and incorporate units to process them in an intelligent manner.

Many of the aforementioned applications consider human subjects as targets and propel the main motivation in this thesis. In order to monitor human behaviors, characteristics, etc., humans have first to be detected and/or informative cues about them have to be extracted. This is a daunting task in real-world environments, which are frequently uncontrolled, dynamic and complex and also because humans appear in many different ways, both to due to variations of color and articulation. Towards the alleviation of the problem, image processing techniques such as image segmentation can facilitate human-centric approaches for garnering information about observed humans.

1.2 Summary of Contributions

Novel and original work presented in this dissertation includes:

- Representation of DIAS architecture's main multiprocessor unit using Stochastic Petri-Nets and evaluation of its simulated operation.
- Classification and presentation of the literature in human activity recognition and proposal of a maturity metric for methodology evaluation.
- Formulation of a blind image segmentation metric that aims in providing a more objective perspective to evaluation.
- Development of a scale and rotation invariant face detection methodology that can produce results for both frontal and profile views of faces in images.
- Development of a novel bottom-up framework for automatic segmentation of human bodies in single images. The main contributions are the algorithm for combination of multiple levels of segmentation, skin color modeling and construction of anthropometric model using the information of extracted face regions and searching algorithm for the most salient body regions guided by soft anthropometric constraints.

1.3 Dissertation Outline

In this thesis, the problem of monitoring and surveillance is studied from many perspectives and focus is given to methods that aim in observing human subjects. The structure of this thesis can

be seen as an hierarchy, starting with the study of a generic architecture of hardware components designed to cope with heavy loads of data, continue with an extensive survey of literature [1–217] evolving around human activity recognition methodologies and then we focus on the main part of the thesis, which is human body segmentation from single images. In [218] we present a study about several issues occuring in surveillance system for assisting the elderly in smart homes, an application that can greatly motivates this research.

- In **Chapter 2**, we revisit the DIAS architecture proposed in [219,220], a multiprocessor system with many components for performing distributed tasks needed in generic large-scale surveillance and monitoring systems. We focus on its main processing unit, simulate its operation with Stochastic Petri Nets and perform experimental results to demonstrate its capabilities. This work is published in [221].

- In **Chapter 3**, we present an extensive survey [222] that attempts to comprehensively review the current research and development on vision-based human activity recognition. Synopses from various methodologies are presented in an effort to garner the advantages and shortcomings of the most recent state-of-the-art technologies. Also, a first-level self-evaluation of methodologies is also proposed, which incorporates a set of significant features that best describe the most important aspects of each methodology in terms of operation, performance and others and weighted by their importance. The purpose of this study is to serve as a reference for further research and evaluation and raise thoughts and discussions for future improvements of each methodology towards maturity and usefulness.

- In **Chapter 4**, we deviate slightly from the main course of the thesis to present blind reference evaluation scheme based on Regional LocalGlobal (RLG) graphs [223], which follows our early work in [224], which aims at measuring the amount and distribution of detail in images produced by segmentation algorithms. Image segmentation is one of the first important parts of image analysis and understanding. Evaluation of image segmentation, however, is a very difficult task, mainly because it requires human intervention and interpretation.

- In **Chapter 5**, we present a face detection method [225] for detection of human faces in images based on skin detection, image segmentation and graph matching. One of the major merits of this approach is that it can cope with both profile and frontal views of the face, while being scale and rotation invariant.

- In **Chapter 6**, we propose a methodology for human body segmentation from images, extending significantly the works in [226] and [227], motivated by the work in [228]. In this method, which

is the key contribution of the dissertation, the face region is used for a rough localization of the human body, adaptive skin modeling and construction of an anthropometric model. By combining multiple levels of image segmentation and soft anthropometric constraints, we are able to probabilistically locate the regions where existence of human body is high using color similarities and finally extract the human body, even in challenging cases.

- In **Chapter 7**, we summarize our approach and our key results, and provide a discussion of the advantages and limitations of the work. We also provide some suggested directions for future research in this area.

2

Representation of the Flow of Images on a Multiprocessor Surveillance System

2.1 Introduction

The proliferation of surveillance and monitoring systems in everyday life has given rise to increasing research interest regarding the implementation and improvement of their software and hardware architecture [229–232]. Their vast majority uses visual information garnered by camera sensors, because of its rich informational content, which can be easily understood by human operators. However, relying complete to human operators for video processing and decision making has been proven to be extremely inefficient, even in small-scale systems. Current trends move towards the automation of these processes [233] and human operators are viewed more like coordinators of the system and while their intervention is still necessary, their labor will be significantly less and more productive. Design and implementation of large-scale surveillance and monitoring systems for demanding applications is very difficult. First of all, processing and transmission of video content requires huge amounts of resources and the resulting infrastructures cannot scale adequately along with the application's demands. Second, in order for the system to be able to replace a human operator and be effective it has to be able to meet real-time constraints and be consistent with specific deadlines in its operation. Significant work has been conducted in this direction, as it can be seen in [234–238]. In this chapter we study the operation of an architecture for generic surveillance and monitoring applications, named DIAS, extending the work previously presented in [219, 220]. DIAS' design includes the data flow from cameras to the rest of the system, where

special units coordinate the communication among the system's components and refine it as they glean information over long periods of operation. Here, we focus on the Image Analysis (IA) section, where most of the processing takes place. Its design and operation is coupled with a high-level grouping of machine vision tasks that demonstrates the flow of information from low-level image processing to high-level understanding, as it is expected to be seen in a complete surveillance system capable of making complex decisions and executing multiple tasks. The emerging communication schemes in the IA section are modeled using Stochastic Petri-Net (SPN) models, which in turn guide the modeling of the whole IA section. The final model is used to demonstrate the section's operation and performance under strenuous conditions.

2.2 General Configuration

The overall organization of the DIAS system architecture is illustrated in Figure 2.1. The DIAS system receives image either through a set of 2-D photoarrays (PAs) from the direct environment, or from a storage area. The values of the image pixels are then fed to a set of preprocessors (Cs), which process these values extracting critical information (parameters), such as average intensities of various picture regions, the number of pixels per region, the locations of informative areas, etc. [234]. Since the preprocessors must function very quickly, they are implemented in hardware. The image parameters are carried over the Master Planner processor (MP), in the PC section, for further evaluation and formulation of the processing plans. A second processor, called Service Controller (SC), receives the abstract plans from the MP processor and schedules the synchronization and implementation of these plans on the Multiprocessor-Array in the IA section. The values of the image pixels are sent to IA under the SC command. The IA section is composed of several special purpose processors in an efficient parallel/pipelined scheme, in order to perform the tasks deemed too difficult for the Cs preprocessors. Throughout the operation, the SC processor monitors the status of the MA array by interrupt-driven accesses to the interface buffer (B), and updates the MP processor accordingly. A unique feature of the DIAS structure is its Back-End processor (BE). The BE processor performs the output operations and accumulates statistical and experimental data about the performed image analysis/processing tasks. These data are then sequentially supplied to the MP processor for future quick decision making and adaptive planning with learning capabilities.

The determination of performance in a system such as DIAS involves many considerations. Each of the DIAS sections will contribute to the overall efficiency in its own unique way. The performance of the Planning and Control section will be constrained by its utilization. The C preprocessors and their associated photoarrays are implemented in hardware to provide an extremely

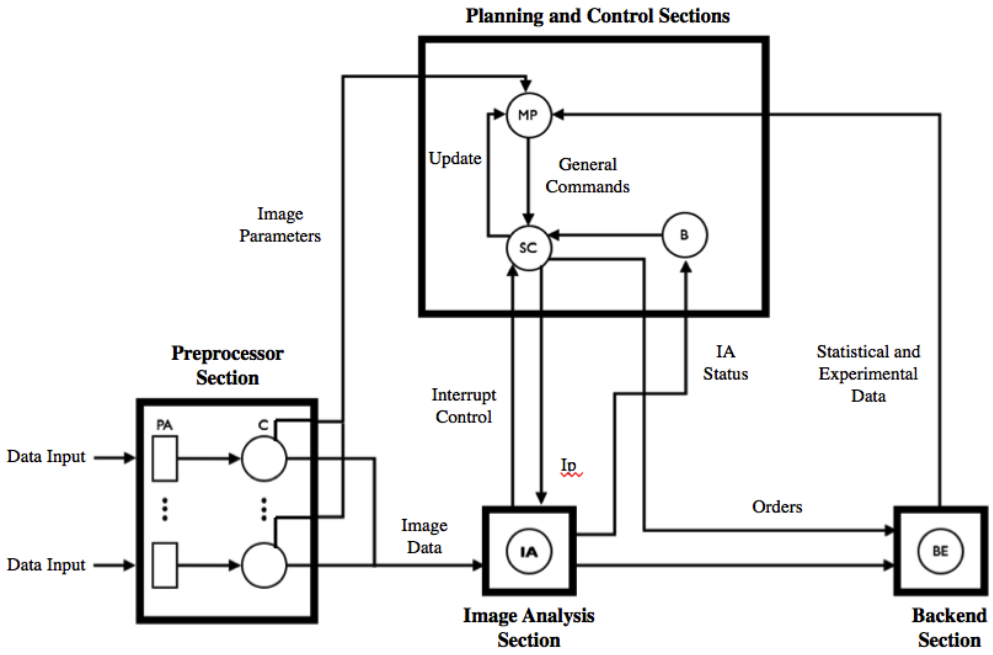


Figure 2.1: General configuration of DIAS.

fast operation. Although high data output rates are possible from the DIAS system, the BE section will handle presently only minor data flow problems by operating on a limited number of statistical and experimental data. In future expansion, the BE section will play a more significant role on the overall performance. Under the assumptions above, the majority of the processing tasks is handled in the IA section. Tasks may be assigned to one or more processors simultaneously, or tasks may remain in a queue for a single processor. Since the IA section carries the burden of processing, the DIAS performance will depend directly on IA array structure's performance.

2.3 Image Analysis Section

In this section se focus on the Image Analysis (IA) section, the system's "heart" of operations. As aforementioned, this section is responsible for the heavy processing of the images that travel through the system. IA can be deemed as the implementation of the Machine Vision Processing Tasks, as seen in Figure 2.4, where each row is dedicated to a specific group of tasks and the processors that comprise it are specifically designed to perform these tasks. The Grouping of the Machine Vision Processing Tasks has been proposed by Bourbakis in [239]. The rationale behind this design is that in an image processing and understanding system, one would expect the flow of information to follow the one depicted in the aforementioned scheme. In other words, it is more probable for similar

types of image processing tasks to be performed on the data packets that move through the system. Thus, we would like a communication scheme that favors communication among the elements that implement similar tasks.

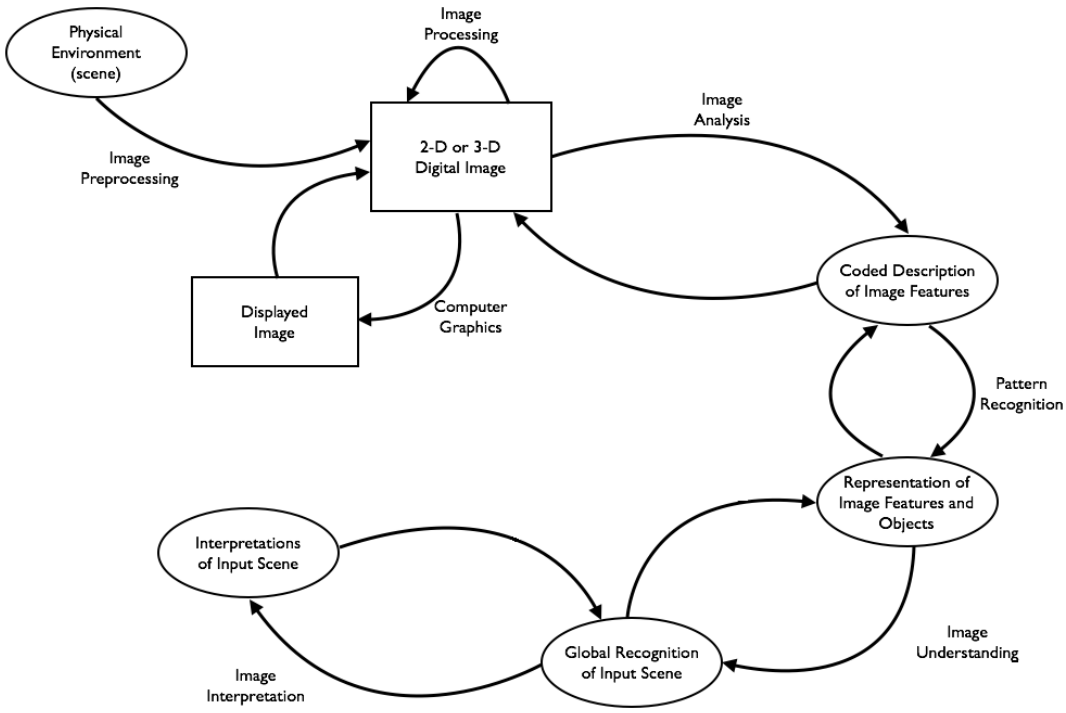


Figure 2.2: Bourbakis' grouping of the machine vision tasks.

The general architectural characteristics of the IA array structure are shown in Figure 2.3. In particular, this called "Common Buses with Switches" (CBS) architecture. It allows the Service Controller to route the data stream from the input bus to any processor in the array by "opening", "closing" appropriately the input switches (a(1) to a(n)). In addition, when a particular processor has finished with its own task, the new data stream may be routed to the output bus or to any other processor by "opening", "closing" the appropriate output switches (b(1) to b(n)). More specifically, the dashed lines, in Figure 2.3, illustrate this routing method from the input bus to processor 2 in the first row to processor "i" in the nth row and to the output bus. In order to ease the design effort and to speed up processing time, each processor in the array will perform one specific task. In this way each machine may be customized to system requirements and each processor may be optimized for the task that is called for.

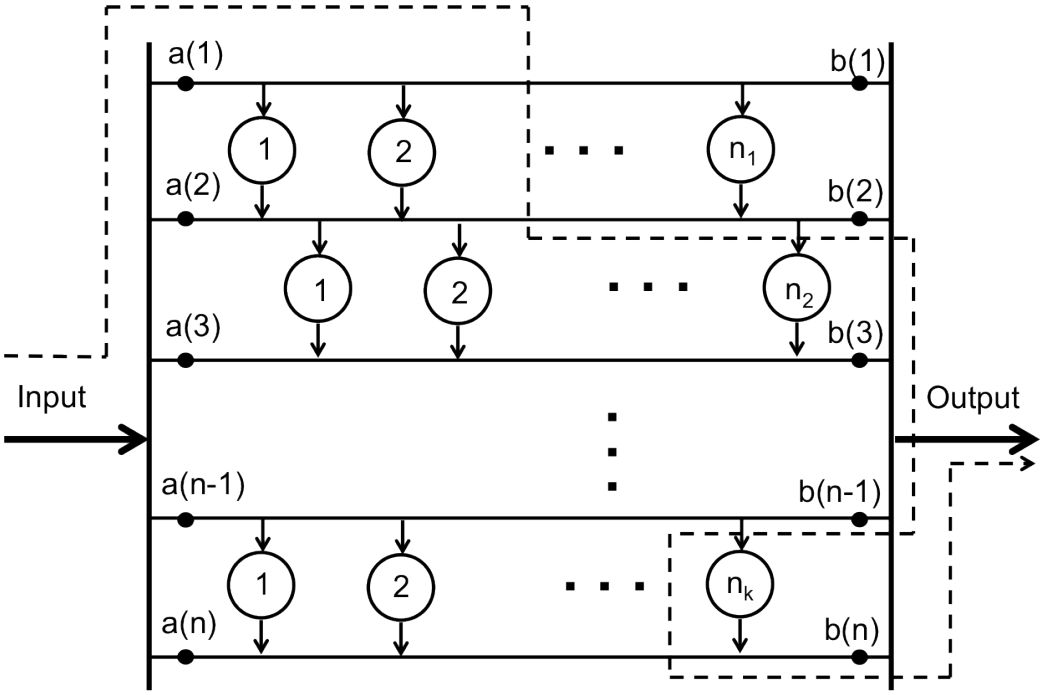


Figure 2.3: Common Buses with Switches (CBS) architecture.

2.4 General Scheme

A trivial scenario for a multiprocessor system that can serve the needs of a complex automated surveillance system is depicted in Figure 2.4. In this case, the PEs form a row and process data in a sequential manner. In order for the system to represent the flow seen in Figure 2.4, PEs are dedicated to specific tasks and form clusters that correspond to the groups of tasks of Figure 2.4. Without going into details about its specific implementation, it can be clearly seen that the resources of this design are limited and the congestion of the buses is expected to be preventatively high for normal operation in the case of heavy processing load. Thus, this design needs to be modified in order to be more efficient and scalable.

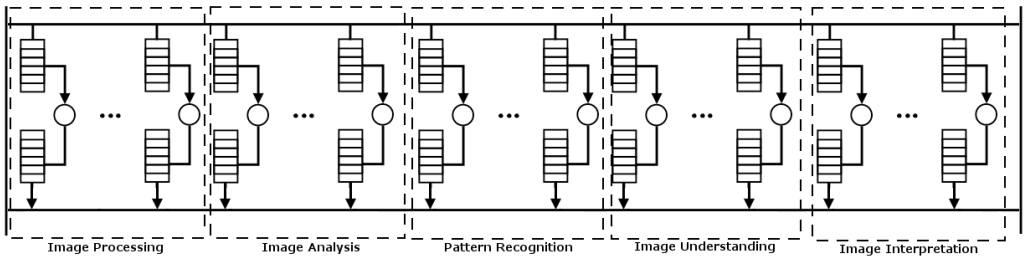


Figure 2.4: Serial PE communication scenario.

In order to achieve the goal mentioned above, we propose designed the IA subsection as an array, as seen in Figure 2.5. Each row is comprised of processing elements (PEs), specifically designed to perform tasks of the same group (i.e. image processing tasks). Subsequent rows then are responsible for conducting subsequent groups of tasks, as described in the scheme of Figure . As we will see below, this structure allows us to design communication schemes among the processors of the same row and subsequent rows that reduce the contention for shared resources, specifically the input and output bus, whose role is to allow communication among IA section's rows and transport data in and out of the subsection.

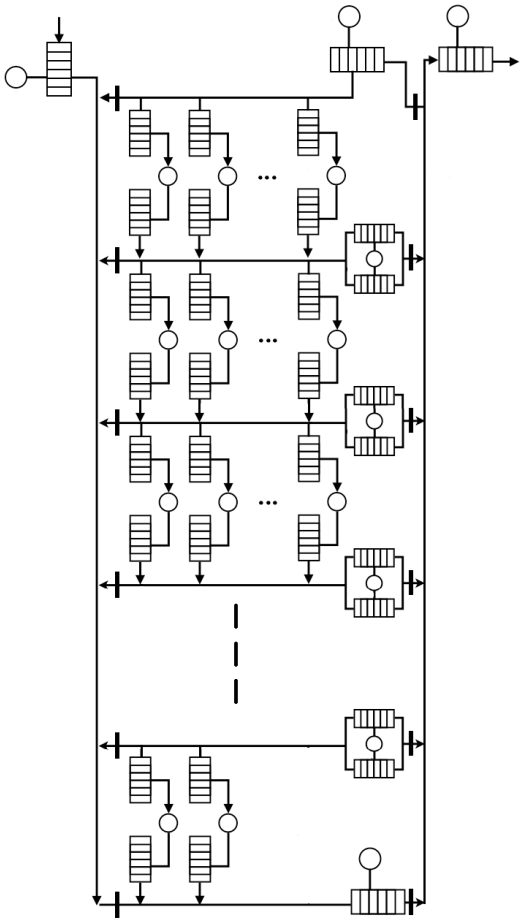


Figure 2.5: Image Analysis (IA) section.

We can better understand the coupling of the IA section and the grouping of machine vision tasks with two example scenarios. First, let us consider a case where the operator of the surveillance system wants to send encrypted information to an external source, similarly to the application described in [240]. The main subtasks that can accomplish this request can be image compression, for fast

transmission, image encryption, since data will be transmitted over non-secure means and hiding, in order to embed additional information into the encrypted data. These subtasks can be mainly characterized as low-level image processing tasks and belong to the first group of the scheme in Figure , or else the corresponding row of the IA for their processing is the first row. Similarly, in a more complex scenario where the goal is object recognition different levels of abstraction are required and more different types of tasks are expected to participate in the overall process. Specifically, image processing, analysis and pattern recognition are intermediate steps for a generic object recognition application and image understanding might be included too in more complex methods. In this case, data would stay longer in the system and move from the first row to the third or fourth row before exiting the system as results of the object recognition algorithm.

2.5 Communication Schemes

As noted earlier, the processing for a single image involves routing of the data around different paths in the IA. The study of IA section can be greatly simplified if the paths can be classified as logical subsystems of the section. From an analysis of the IA, eight different communication schemes, as seen in Figure 2.6, are identified. These schemes are independent of each other, although the order in which the image may be processed through the system will be governed by a fixed set of rules with respect to the ordering of these schemes. Table 2.1 indicates the source and destination processors involved in each of the communication schemes and information regarding the use of other system resources, like the Input and Output bus and the Common (local) buses.

Table 2.1: Communication Schemes Resource Allocation (X indicates that resource is used)

CS	Source	Target	Input Bus	Local Bus	Output Bus
1	IBP	$LQIP_i$	X	$a_{i-1}b_{i-1}$	
2	$LQIP_i$	$LQOP_i$			
3	$LQOP_{i-1}$	$LQIP_i$		$a_{i-1}b_{i-1}$	
4	$LQOP_i$	$LQIP_k$	X	a_ib_i, $a_{k-1}b_{k-1}$	
5	$LQOP_i$	CBP_{i+1}		a_kb_k	
6	CBP_k	CBP_i, $k > i$		a_kb_k, a_ib_i	X
7	CBP_i	$LQIP_i$		$a_{i-1}b_{i-1}$	
8	CBP_i	OBP		$a_{i-1}b_{i-1}$	X

In more detail, six distinct types of flows can be defined, combinations of which lead to the

generation of all possible paths that packets of data can take in the IA section. In general, the communications schemes can be divided to those that move data down-wards (upper to lower rows) and those that move them upwards (lower to upper rows). First, the packets arrive to the Input Bus Processor (IBP), which sends them to one of IA section's rows. This is implemented with CS1. CS3 is designed for communicating data between two subsequent rows in a downward direction. As it can be seen in Figure 2.7(a), this scheme does not transmit data over the input or output bus, thus leading to less contention for these shared resources. In a case where the flow of information follows the logic of a scheme similar to the grouping of machine vision tasks, illustrated in Figure 2.4, meaning a flow that follows an almost serial sequence of steps, this communication scheme would lead to improvement in performance. CS4 is similar to CS3, but is designed for the communication of not adjacent rows and partially uses the input bus for the transfer. In order to move the data between two rows, but upward, a combination of communication schemes has to be employed, namely CS5, CS6 and CS7. Finally, data leave the system using the output bus, following the flow defined by communication schemes CS5 and CS8. The communication schemes mentioned until now are responsible for moving the data from output buffers to input buffers. When data arrive to an input buffer, meaning that they are waiting to be processed, communication scheme of type CS2 guides them to the respective PE and then to its output buffer, where they wait for transmission until the appropriate buses are available. Finally, it should be noted that whenever communication over the buses is required (all communication schemes except CS2), Communication Bus Processors (CBPs) are involved. CBPs are processors specifically designed for gathering requests, opening and closing of switches and conflict resolution. Generation of all possible sequences of communication schemes can be conducted using a Context Free Grammar. It uses the communication schemes as letters of its alphabet and is defined as $G = (\{S\}, \{CS1, CS2, CS3, CS4, CS5, CS6, CS7, CS8\}, P, S)$, with production rules:

$S \rightarrow CS1\, CS2\, P\, CS5\, CS8$

$P \rightarrow CS3\, CS2\, P \mid CS4\, CS2\, P \mid CS5\, CS6\, CS7\, P \mid \epsilon$

2.6 Modeling of Communication Schemes

The operation of the communication schemes is formally described using Stochastic Petri Nets, which model the interactions and coordination of the resources involved in each scheme, according to its communication protocol. Petri Nets are a powerful tool for modeling and analyzing systems where events take place concurrently and in parallel [241–243]. Stochastic Petri Nets can be formally defined as a quintuple:

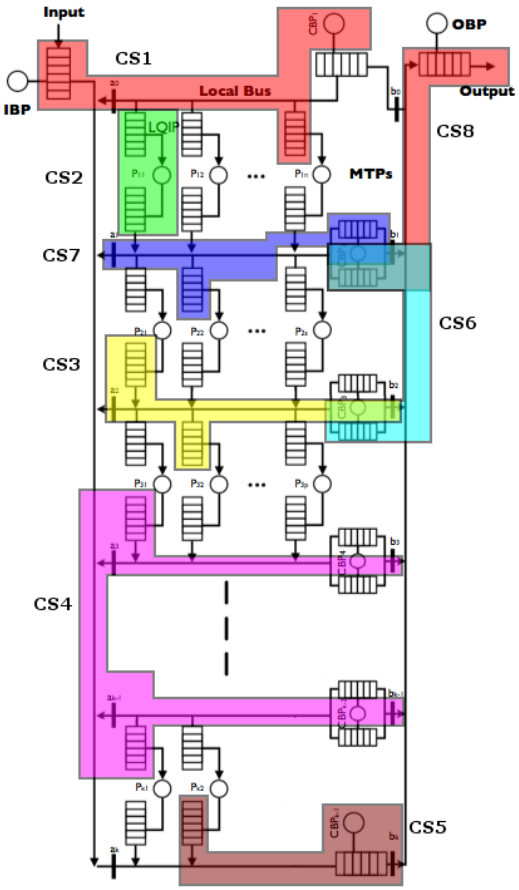


Figure 2.6: Image Analysis section and Communication Schemes (CS).

$SPN = (P, T, A, M_0, L)$

where

$P = \{p_0, p_1, \ldots, p_n\}$

$T = \{t_0, t_1, \ldots, t_n\}$

$A_i \subset (P \times T)$

$A_o \subset (T \times P)$

$A = (A_i \cap A_o)$

$M_0 = \{m_{00}, m_{01}, \ldots, m_{0n}\}$

$L = \{l_0, l_1, \ldots, l_n\}$

where P is a set of places, T is a set of transitions, Ai is a set of input arcs, Ao is a set of output arcs, M0 is a set of initial markings for the Petri Net and L is the set of firing rates associated with the transition. Petri Nets may contain tokens graphically drawn as black dots. A transition fires

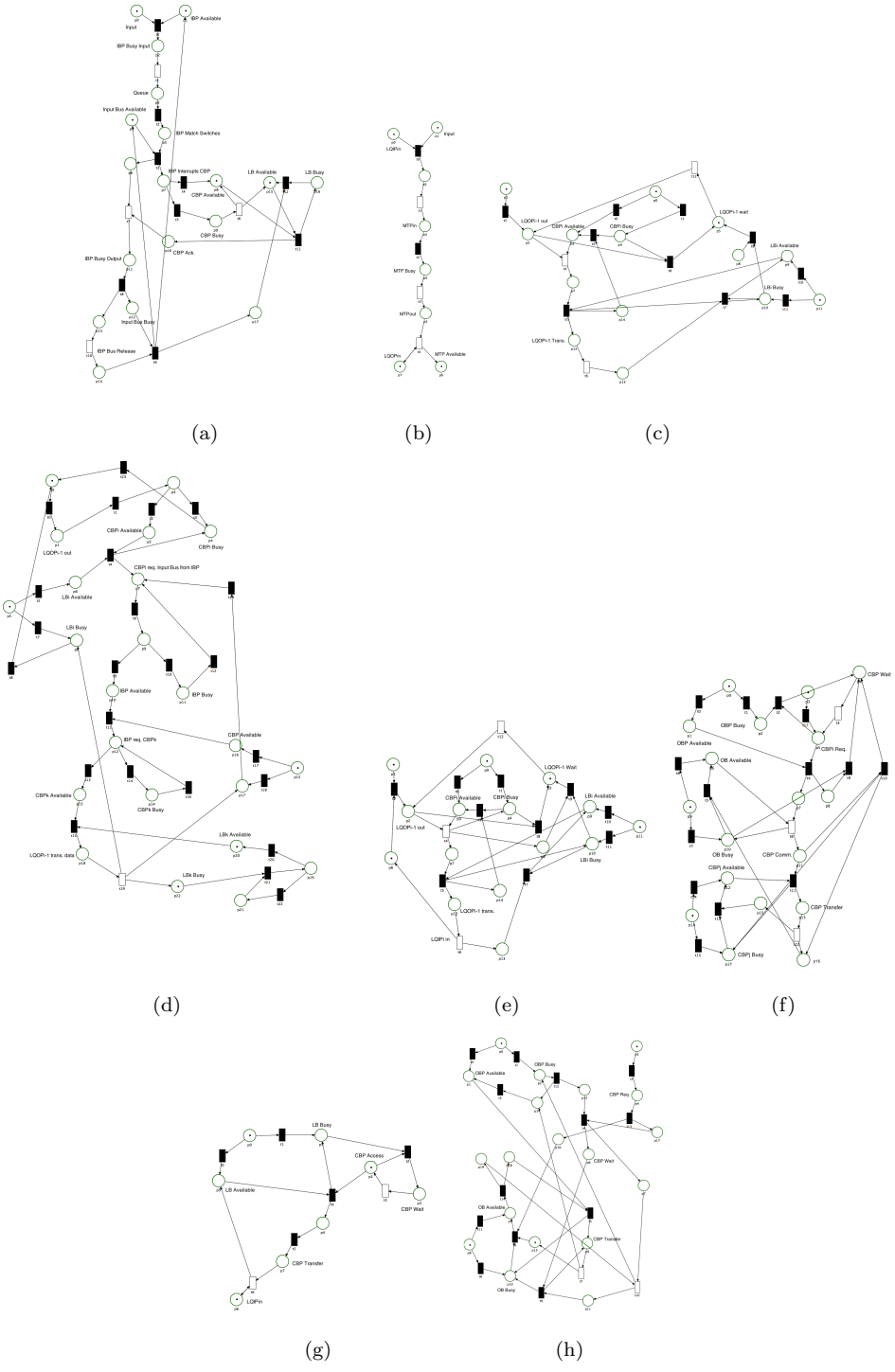
when all of its input places contain a token. When a transition fires, a token is put out in places connected to each of its outputs arcs. In case of conflict in a transition, in the case for example where a place has two output arcs, the firing rule is based on assigning probabilities to each of the output arcs. This notation is useful in representing certain activities that may or may not occur. For example, if place p1 represents such a place, then it could represent the fact that a certain resource is requested. If there are two output arcs on this place, then the places connected to each of the arcs could represent the states indicating that the particular resource is available or not. The Petri Nets for CS1 through CS8 are illustrated in Figure 2.7.

2.7 Modeling of IA unit

After having formally defined the communication schemes and the CFG that connects them, we can construct the model of the whole IA unit, using the SPN model shown in Figure 2.8. This model allows the performance analysis of the complex system we have described, in various conditions. Some simplification assumptions have been made in order to reduce the complexity of the model and make the results more interpretable. First, the queuing time and delays imposed by the Communication Bus Processors (CBPs), which regulate the flow of information and open/close the appropriate bus switches to realize it, are considered negligible. This is not a strong assumption, because these special processors only deal with requests and not data, processing and scheduling of which is fast. Second, the time to read and write from memory and from and to the local input and out queues, respectively, is neglected too. This is also reasonable, considering that there is no contention for the local memories and they act as simple, fast buffers.

2.8 Experimental Results

In order to remain closer to the rationale behind the grouping of the machine vision tasks, the model of the IA unit was modified as follows. First, the number of rows is fixed to five, which corresponds to five groups of tasks, like the ones seen in Figure 2.4. Of course, this number can change according to one's needs, but since the aforementioned grouping aims in covering the whole spectrum of machine vision tasks, from low to high level, it is argued that the selected number of rows is appropriate for a generic architecture, such as DIAS. Second, for the experiments we assigned different probabilities to the transitions that better adhere to the expected from of information, balance the processing load better and indirectly bound a packet's time in the system. Packets are allowed to visit every component of the system in all possible ways, but they are biased to do so in a "waterfall" fashion, beginning their flow from the upper rows and moving sequentially to the lower rows of the IA unit.



(a) (b) (c)

(d) (e) (f)

(g) (h)

Figure 2.7: Petri-Net models of the Communication Schemes.

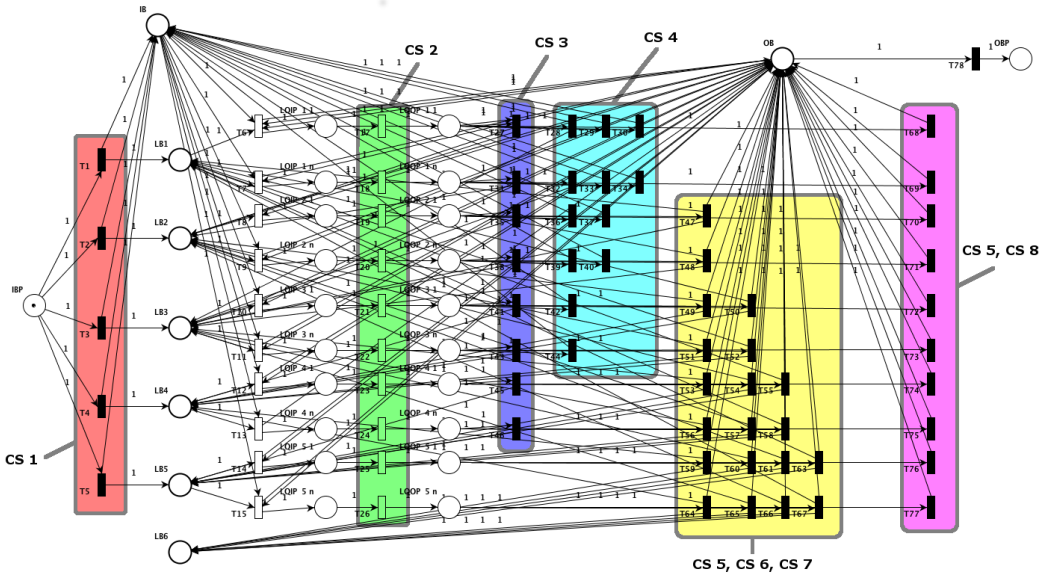


Figure 2.8: Petri-Net model of the IA section.

This is realized by the assignment of high probability in firing the transition that moves newly arrived packets to the first row. This probability gradually diminishes for the transitions to lower rows. Also, the sequential flow is favored, meaning that packages move from one row to the one directly below using CS3, which reserves only one local bus and no I/O buses. Finally, the packets are more probable to exit the system from the lowest rows.

After having described the general organization and communication patterns of the IA unit, we define the characteristics of its resources, namely the processors and buses. The rates used in the experiments derive from the bus' transfer rate, which is fixed to 5 GBytes/sec. This choice corresponds to transfer rates of modern architectures. The service rate of the PE's is assumed to be ten times slower, or 500 Mbytes/sec in this case. This is reasonable in many cases, where processors can operate synchronously with the bus, but since the complexity of the tasks they are assigned to varies and is almost always high, their speed is notably lower than the bus' speed. Our choice is arbitrary, however not optimistic. One of the goals of DIAS architecture is to encompass specifically designed processors that can carry on tasks with hardware. Thus, their speed in many cases could match that of the bus, but we relaxed this assumption to make the conditions of operations more strenuous. The packets that come and travel through the system are considered to be of 1 MByte size, which again is a restrictive choice. A packet this size corresponds to an uncompressed image of approximately 591x591 pixels, which is not commonly the case. Additionally, even if a packet's size was big as it entered the IA unit in the form of an image (or video frame), it would be expected

to rapidly be reduced as it is being processed and transformed into feature vectors, interpretations, etc.

During the experiments the model's performance was tested using two metrics, the Processing Time (PT) and Processing Element Utilization (PEU), for different input rates and number of PEs. Processing Time is defined as the average time that packets spend in the PEs over the total time they stay in the system, which includes the times spent in queues and buses. Processing Element Utilization, as the name suggests, measures the average number (percentage) of active PEs during the system's operation. The ideal value for both metrics is 100%, which in the case of PT implies limited contention and high throughput and in the case of PEU that there is no waste in resources, specifically in PEs. However, increase in one metric means decrease in the other, so another interesting measurement that derives from the curves of the two aforementioned metrics, as seen in Figure 2.9, is the point of their intersection, which shows how many processors are needed so that the system achieves a balanced trade off between the two metrics. In the case where this balance is desired, the number of PEs that achieves it is considered optimal. Figure 2.10 depicts the optimal number of processors for different input rates, as well as the number of PEs the system requires in order to achieve its maximum PT value.

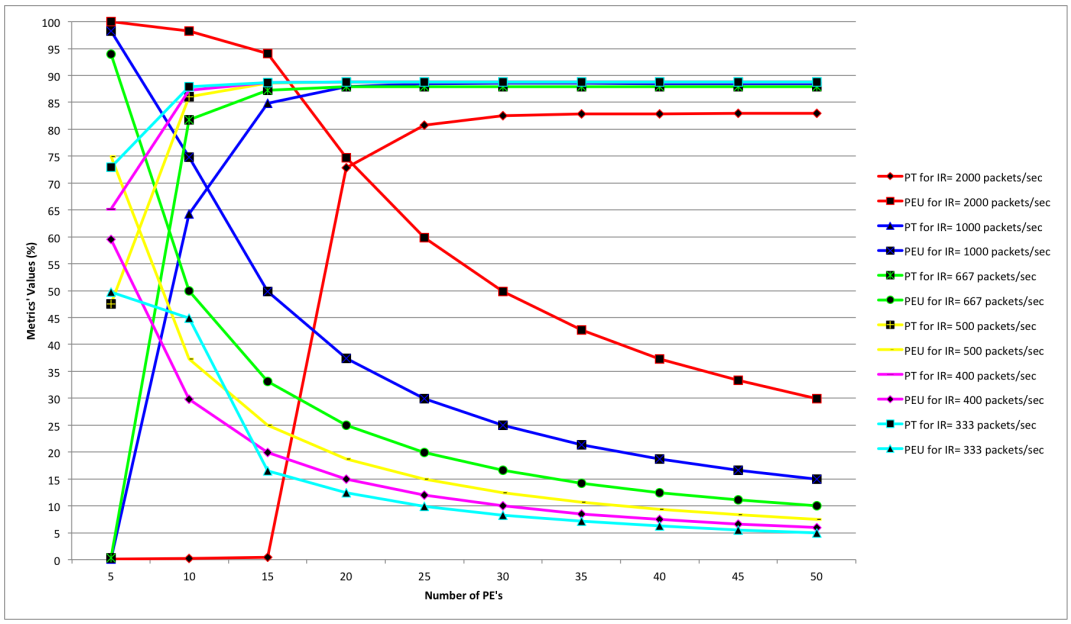


Figure 2.9: IA unit Processing Time (PT) and PE Utilization (PEU) for different input rates and number of PEs.

Figure 2.9 shows that the maximum value for PT achieved was approximately 89%, meaning that only 11% of the overall time packets spend in the system is spent in queues and buses. For

input rates lower than 333 packets/sec, PT is close to its maximum value was achieved for five PEs in total, one in each of the five rows, while at the same time PEU value is low because the PEs are underutilized. What is interesting is to observe how the system operates for greater input rates, where more than five PEs are needed and the proposed array becomes more useful. When the input rate becomes greater than 400 packets/sec PEU is 60% for five PEs, which is acceptable given that we also desire to not have excess of resources. The most significant observations of Figure 2.9 are summarized in Figure 2.10. For input rates from 400 to 1000 packets/sec, which corresponds approximately to 20 to 50 cameras, respectively, transmitting high quality images at 20 frames/sec, the system's PT can converge to its maximum value with almost linear increase of the number of PEs. Also, optimal number of PEs increases also in a linear fashion, which shows the effectiveness of the proposed system for high input rates, as it maintains reasonable resource requirements and is able to guarantee strict real-time constraints. However, the system's performance is eventually limited by the performance of its components and more specifically the bus speed. For input rates greater than 2000 packets/sec the contention for the I/O buses becomes too great and does not permit convergence to the maximum PT. As shown in Figure 2.10, the increase in the number of PEs is no longer linear and does not lead to proportional improvement to the system's performance.

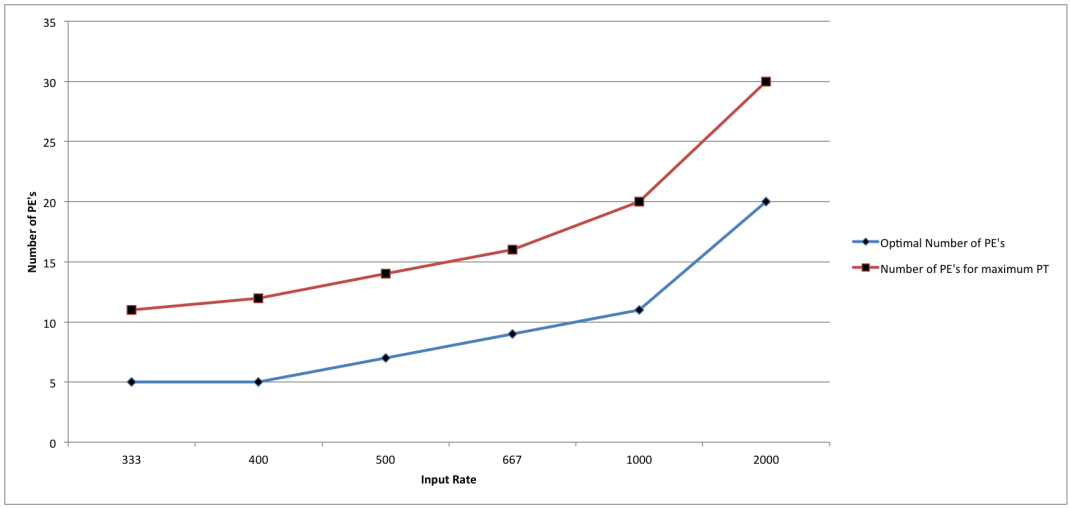


Figure 2.10: Optimal number of PEs for different input rates.

3

Survey on Vision-Based Human Activity Recognition Methodologies

3.1 Introduction

Vision-based human activity analysis systems have attracted the attention of the research community and industry, especially during the last two decades. Recent technological advances are fueling the interest, which is increasing yearly at a very high rate, as indicated by the recent number of publications. There have already been immense investments in surveillance and monitoring systems and installations, which however stumble upon a common obstacle: the need for human operators. Most of the traditional systems have been designed to solely acquire images and are enhanced only with some simple intelligent capabilities, leaving all the high-level reasoning and inference to human operators. Manual monitoring is a very tedious task and has been proven to be very expensive and ineffective for large scale and everyday applications. Thus, new solutions are emerging and heading towards automatic, autonomous, ubiquitous surveillance and monitoring systems.

Such systems can be applied in human-centric applications, where the subject of interest is a human or group(s) of humans in an observed scene, whose actions and interactions among themselves and the environment are to be recognized. Some of the most prominent types of human-centric applications are summarized as follows:

Security surveillance in special areas: A common application is the surveillance of areas such as travel sites and public areas, military bases etc., to reinforce their security. Usually, these environments contain sensitive areas where trespassing is not allowed. Human activities and interactions

are monitored and recognized in a more general manner and usually alarms are raised whenever suspicious activities are detected.

Crowd flux analysis: These applications are mostly applied to outdoor environments or large public areas, where there are flows of many individuals, who in turn form crowds. The details of individuals fade as crowds grow, so the point of interest here is the analysis of the crowd as a unit, where both local and global fluctuations in the flow exist and indicate different phenomena (i.e. panic).

Behavior analysis and activity recognition: Smart homes, medical and care centers for the elderly, people in need of attention etc., can greatly benefit from systems that have the ability to recognize efficiently (normal or abnormal) actions and activities of individuals, interactions among humans or humans and the environment. This information can be of vital importance to doctors and physicians, for prevention of dangerous situations and to enhance assistive living. Moreover, helpful cues can be derived for the higher-level task of emotion recognition and enrich well-established traditional methods, such as those based on facial expressions.

In this work, we focus on the third category and we are specifically interested in methodologies that consider the motions of the whole human body. Regardless of the specific application, the main characteristics of the reviewed the methodologies are the use of visual data as input, which are manipulated with the use of tools and methods that spring from the fields of image processing and artificial intelligence, so as to recognize human activities.

The words events, actions and activities are usually used interchangeably in the literature and oftentimes the distinction among them is not clear. However, a common agreement among researchers over, at least a vague, discrimination among them should be made, as it would make evaluation and exchange of information easier and more meaningful. One way to distinguish each type would be via a semantic hierarchy that connects them, as shown in Figure 3.1. We define six layers of abstraction where components of each layer are generally a combination of components of the previous layers.

At the bottom level, the first contains single static poses of body parts, so knowledge at this level is restricted to specific parts of the human body and there is only spatial information. In the second layer temporal information is incorporated and sequences of poses form gestures, primitive actions of body parts with semantic meaning. One or more gestures of body parts combined sequentially, concurrently and/or in parallel lead to the actions of the third layer. Thus, actions can be recognized by observing the spatio-temporal changes of the whole human body. Examples of actions are “walking”, “skipping”, “jumping”, etc. and most of the widely used datasets for training and testing human action/activity recognition methodologies contain usually video segments in which actor(s) perform a certain action. However, actions themselves can be building blocks for more

complex actions or activities, as seen in the fourth layer of the hierarchy. This layer of abstraction bonds naturally with the hierarchical way human understanding is organized. It is also necessary because simple actions alone do not possess enough semantic content to describe concepts such as "playing a specific game" or "building something", which can be seen as combinations of distinct simple actions. Up to now, only activities of one person are considered. In the fifth layer group activities are defined as interactions among different individuals, actions of whom are intertwined in space and time and acquire a different meaning. Finally, in the sixth layer events are defined as the most semantically abstract concepts that take into account the observed scene as a whole. One or many individuals and/or groups of individuals can collectively lead to specific interpretations of events that occur in the scene, for instance "shopping in super market", where there might be people browsing, pushing shopping carts and waiting in line to be serviced by the store's clerks. Explicit or implicit knowledge about the environment, like where or when events occur and what objects are involved can provide valuable cues that dissolve ambiguities in recognition. Of course, this knowledge can be helpful in the lower levels too, especially when activities and group activities are to be recognized, because usually people behaviors are often correlated with the environment they are in. However, since there is not a clear and an established definition of terms for the aforementioned motions and appearances of the human body, in the subsequent summaries of the methodologies the original terms of the authors will be used.

3.1.1 Generic Architecture

Figure 3.2(a) shows a generic architecture for a complete system that performs the overall task, extending the one proposed by Ko in [2] and Figure 3.2(b) shows a compact version with its crucial components, which will be used later for the classification of the methodologies. A brief description of the layers of the general architecture follows, so as to get a better insight to the problem at hand.

3.1.1.1 Sensor Level

Since we are interested in methods that process visual data, the sensors used in the bottom layer for data gathering are mainly cameras. The most common types of cameras are the typical grayscale or color surveillance or even typical web cameras, since they have become a commodity a long time before and today can capture high quality video at a low cost. However, many new types of cameras have already been designed and are gaining ground, such as cameras with extra sensors built in, like infrared and depth sensors. Installation and calibration of cameras are the first issues that require attention and in case of multicamera environments more challenging issues arise, such as communication protocols and distribution of processing and information, to name a few.

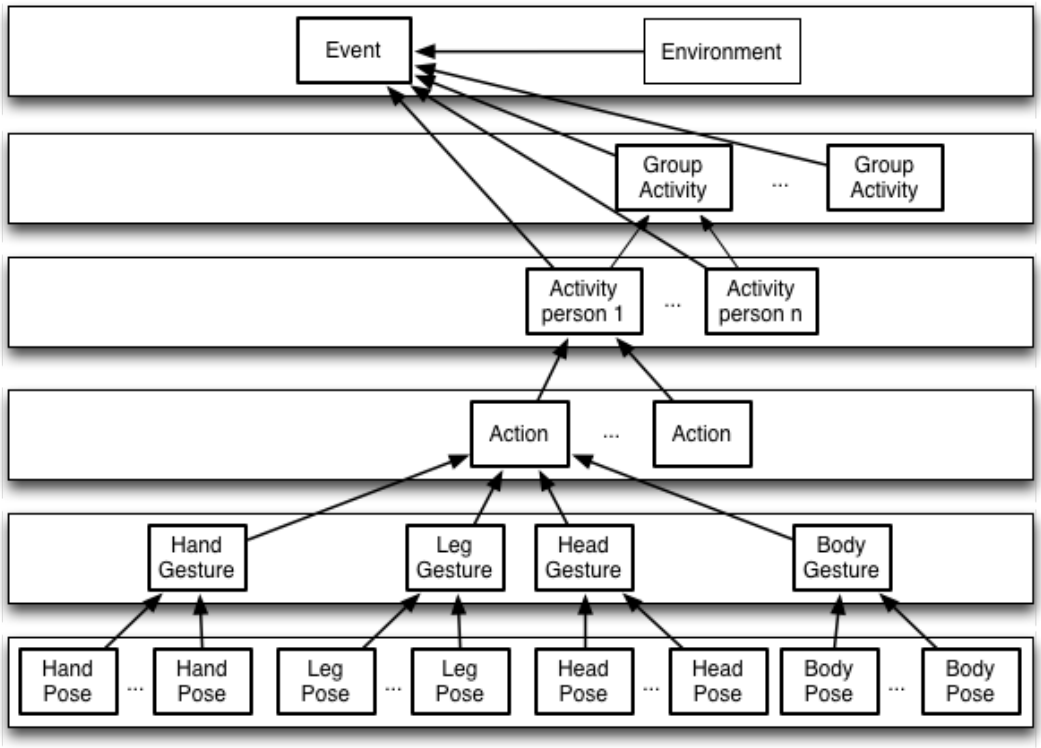


Figure 3.1: Semantic hierarchy of action primitives, actions, activities and events

3.1.1.2 Pre-processing

In many methodologies, the visual data gathered by the sensors require a pre-processing stage before they become ready for further processing. Some of the most common pre-processing steps are background modeling, foreground extraction and tracking. Background modeling is a crucial step in this type of techniques, indicating the regions that belong to the environment and should be removed. In real applications and world conditions, backgrounds are dynamic and impose huge difficulties in their successful extraction, like several moving objects of no interest, clutter, shadows and illumination changes. Many methods have been proposed to mitigate their impact, such as Gaussian mixture models and pixel parameters, panorama graphs and motion compensation and are still under investigation, especially concerning the 3D domain. Foreground extraction is the natural course of action after the background has been modeled and aims at segmenting the moving regions of interest. The most common approaches, offering different trade-off between performance and complexity, are background subtraction, where the static background is first captured and then subtracted from consequent frames, temporal differencing, makes use of the pixel-wise differences between two or three consecutive frames and optical flow which employ the notion of flow vectors

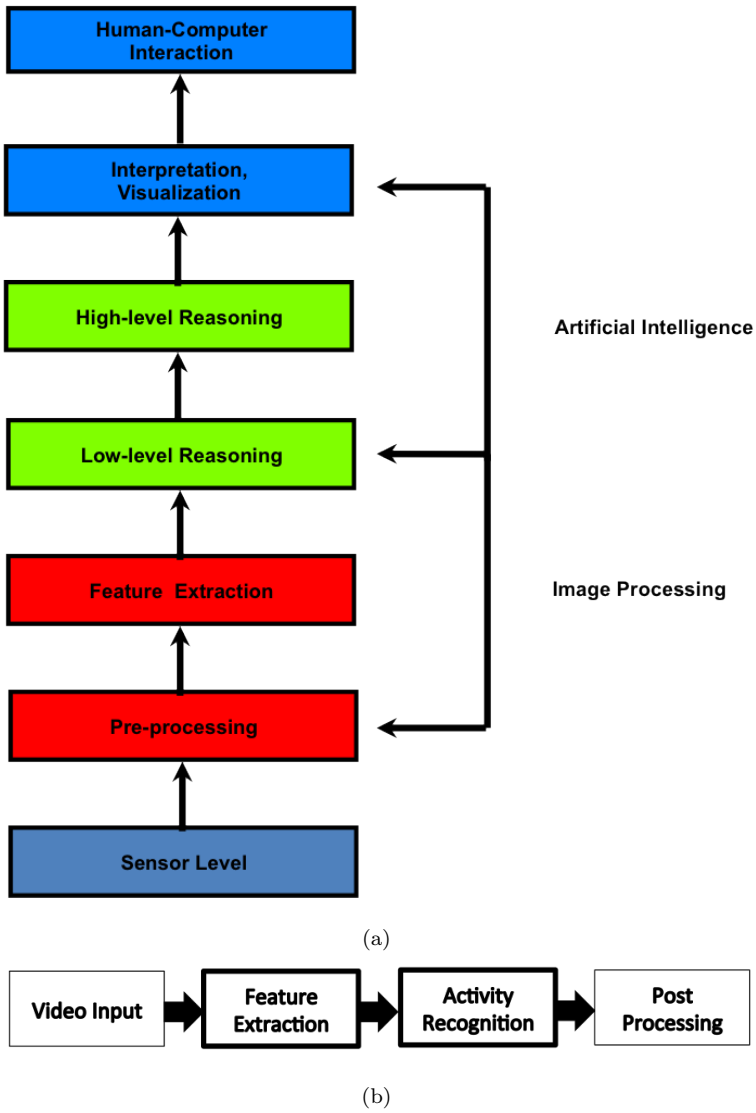


Figure 3.2: Generic architecture of a vision-based human activity recognition system (extending the one proposed by Ko in [2]) a) Full view, b) Compact view

for moving objects. Foreground refinement can be seen as a separate process that aims at cleaning the extracted foreground from false positive regions or regions of no interest and further enhance the segmentation. It usually takes into account shape and/or motion characteristics of the extracted silhouettes that form the background, for instance the aspect ration of a blobs height/width and whether its pixels exhibit a uniform and probably periodic motion. After the region of interest (ROI) has been extracted, it is often desirable to track its location in the subsequent frames. This means to follow its movement and acquire its 2D or 3D coordinates in the observed scene. This information

provides very useful information allowing the comprehension of temporal evolution and identification of objects throughout the video stream. This knowledge in turn could enhance the communication of multiple-camera systems, giving them the means to exchange information about the objects they observe, distinguish them and proceed to camera handoff and data fusion. Occlusions and cluttered background are factors that can hinder the tracking process, making it a challenging task, especially if we desire it to operate in real-time.

3.1.1.3 Feature Extraction

This is the first crucial step in every methodology because here is where the sensed image/video data are represented in a compact, meaningful, machine understandable manner and converted to perceptually significant numeric or symbolic forms. Features should be able to balance the trade-off between specificity and generality, where specificity here means their ability to accurately capture the motions, actions, regions etc. of interest, whereas generality referrers to their ability to accurately describe many of the aforementioned elements. Reliability and robustness in this level are of the utmost importance, since any errors occurring here will propagate to the rest of the systems components and degrade its performance significantly. However, care should be taken at the same time so as to keep the computational requirements low. Manipulating video content is very demanding in resources and this factor alone is enough to determine the effectiveness of the whole system and influence its design. In this survey we classify the features into three broad categories, namely global, local and hybrid, with latter being the combination of the first two. Briefly, global features treat the human body or body parts as a unified entity and usually are combined with pre-processing methods, whereas local features extract and describe only points and patches of interest without explicitly localizing the human body/parts. This section will be further analyzed later.

3.1.1.4 Low-Level Reasoning

The biggest portion of the literature considers it of being part of the feature extraction process, where the notion of semantics is firstly introduced into the methodology. However, in many complex, hierarchical systems low-level reasoning is performed on top of feature extraction, mainly in order to organize the features into primitive actions that will become the structural blocks for high-level reasoning and further organized to more complex activities of behaviors. One example is the organization of a hierarchy that moves from simple gestures and postures to primitive actions and higher levels of human activities, as pointed out by Kojima et al. in [147]. A typical approach for learning primitive actions is by learning statistical models of the temporal sequencing of motion descriptors. Popular ways to discover primitive actions are motion templates, proposed in the

seminal work of Bobick and Davis in [148] and their extensions, such as the motion history volumes, proposed by Weinland et al. in [150]. Hierarchical approaches that capture the notion of low-level inference can be found in the methodologies of Ivanov and Bobick in [149], who employ stochastic grammars, Natarajan and Nevatia in [151], who employ hierarchical Hidden Markov Models (HMM) and Ryoo and Aggarwal in [187], who perform cuboid spatio-temporal feature matching.

3.1.1.5 High-Level Reasoning

At this point, the methodology examines all the information gleaned in the previous steps and formulates the final decisions, or in our case activity recognition, by assigning semantic labels to the observed data. If we consider the feature extraction as the heart of the methodology, then this is its mind. In general, it is achieved following two broad approaches, namely knowledge-driven and data-driven approaches. Knowledge-driven approaches require expert knowledge to construct the worlds model, which describes the way activities are formed and can be extended to different levels of granularity. Examples of this type of approach are the works of Ryoo and Aggarwal in [183], who use a context free grammar (CFG) for composite action representation, Minnen et al. in [184], who use an extension of stochastic grammars and the VERL ontology framework of Francois et al. in [185]. On the other hand, data-driven approaches attempt to learn the model of the world or distinctive separation lines or planes that classify different events, using samples of data for training. Methods employed in this approach emanate usually from the fields of machine learning and data mining. Examples of this approach are the works of Robertson and Reid in [186], who present a general method for human activity recognition based on Hidden Markov Models (HMM) and Li et al. in [188], who base recognition upon action graphs. This section is the second parameter of our methodology classification scheme and it will be analyzed further later.

3.1.1.6 Interpretation, Visualization

One of the missing components in the majority of the current works is the part that represents the final decision in more human understandable manner, like 3D visualizations and interpretation of events in natural language. There are still other issues to be resolved in the problem of recognition alone and this step will probably be ignored until some real breakthroughs have ben achieved and recognition systems are ready to become a commodity. When the methodologies manage to become the backbone of real systems, this part will play significant role in the qualification of the product, since it will need to address the needs of not just researchers, but also simple users who do not have an understanding of the methodologys underlying infrastructure.

3.1.1.7 Human-Computer Interaction

This step is related to the previous one and it is one of the future needs the methodologies will need to address, when they reach the level to be used in real systems. This step is not necessarily related to activity recognition, since it deals with the methods the data are stored and presented, and the interfaces and platforms that allow effective communication with end-users. However, since the applications of activity recognition and thus the group of end-users will be more specific, this layer should be able to use existing and future technologies to make the human-computer interaction (HCI) more pleasant and customizable to the individuals demands.

3.1.2 Challenges

Even today, besides the numerous methodologies proposed and the collaborative effort of the researchers to produce robust and accurate systems and frameworks for activity recognition, several challenges still exist that have not been yet fully addressed. These challenges can hinder the performance of the methodologies significantly and render their applicability impossible to real-world scenarios. Even from the sensor level, camera effects and distortions can introduce significant amounts of noise. Even in high quality video streams, quick motions can cause motion blurring and camera motion can alter the perception of the localized motions. Moreover, dynamic illumination and shadows, especially in appearance-based methods, are still a major challenge. It is commonly assumed that illumination conditions are controlled, however this is not true in real-world scenarios. Another problem that has to be dealt with are occlusions, either by the environment or moving objects, of self-occlusions by body parts, which limit the observations of potentially informative cues. Also, cameras position plays a significant role in recognizing activities, because activities look different from different views ad the feature that are to be extracted have to be view-invariant so as to dissolve the imposed ambiguities. Finally, activity itself is inherently ambiguous to define. Different people perform the same action differently and each activity is so complex that can be decomposed to many primitive parts. As a result, the domain of application has to be limited, since human experts cannot possibly conceive and design every possible combination and machine-based methods are not expressive or powerful enough to capture this spectrum either, even with a huge amount of training data. Although many sophisticated methodologies have been developed and significant technological advancements have been made towards the direction of the general problem of surveillance and activity recognition, truly successful results cannot be claimed in real world conditions.

3.1.3 Related Work

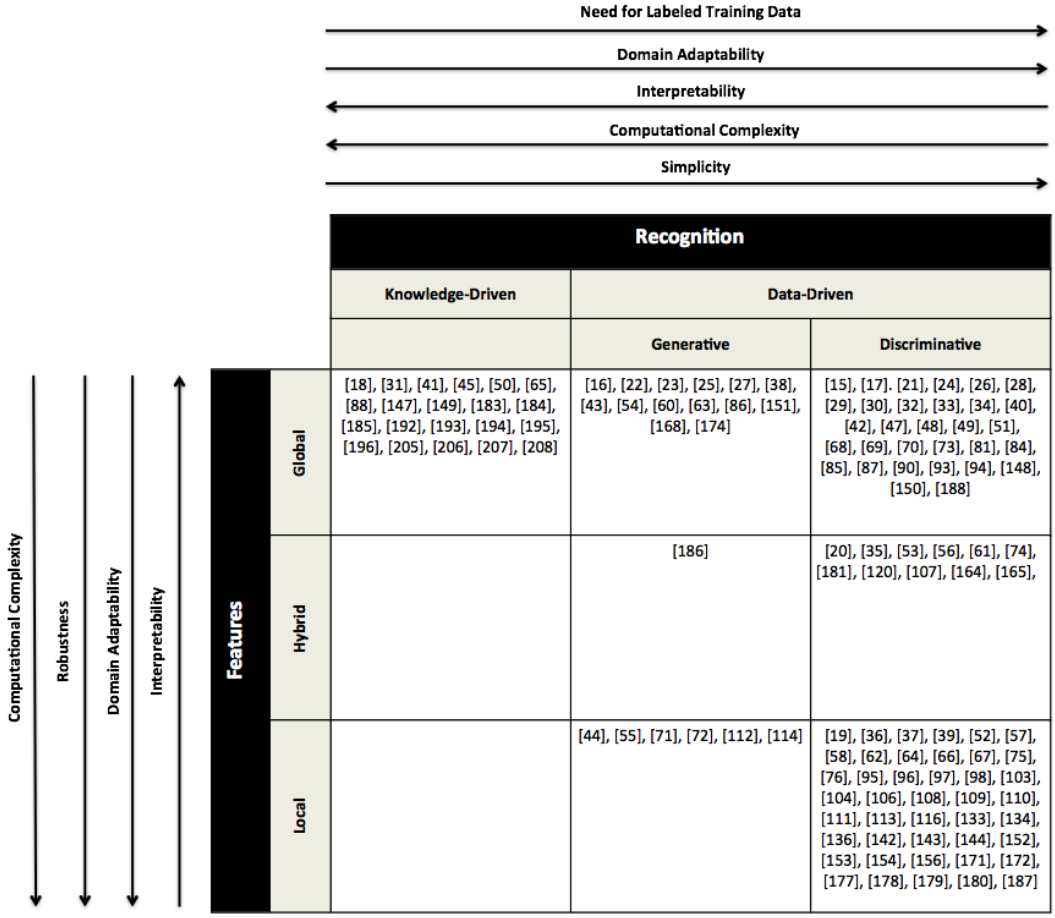
The problem of vision-based human activity recognition has garnered a lot of attention and there many helpful surveys have been proposed in the recent years, which review and organize the related literature and aid researchers to achieve a better understanding of the problem. Candamo et al. [1], Ko in [2] and Hu et al. in [3], offer a comprehensive survey of image processing human behavior recognition algorithms. Moeslund et al. in [4], Aggarwal and Ryoo in [12], Weinland et al. in [13] and Poppe in [14], present different interesting taxonomies and discussions of relevant papers. Turaga et al. in [5], focuses mostly in high-level recognition and not the low-level image processing. Radke in [6], Taj and Cavallaro in [7] and Rinner and Wolf in [8], focus mostly on distributed smart camera architectures and distributed computer vision algorithms.

In this survey, the current state in research and development of vision-based human activity recognition systems is reviewed and the attributes of the most promising current achievements of several worldwide projects are summarized and compared. Our main contribution is the proposal of a simple formula for a first level methodology evaluation, which aims in embodying a holistic assessment from the points of view of those who are essentially interested in the development or use of a methodology. As opposed to the most relevant of the surveys, we focus deeply into the description and evaluation of the most recent and prominent methodologies. A taxonomy based on the most important components of the methodologies for human activity recognition is proposed, namely the feature extraction and activity recognition. Except for better organization of the categories the methodologies fall into, this taxonomy acts also as a first step of the final evaluation and gives a quick insight about the main attributes of these categories.

3.2 Classification

In the previous section, the general architecture of a vision-based human activity recognition system was discussed. Figure 3.2(b) shows a compact version with its most basic components, namely the feature extraction and activity recognition steps. This flowchart can be also viewed as a generic representation of a typical machine learning or data mining process, adapted for video input and the task of human activity recognition. Feature extraction and activity recognition techniques are the common denominator of the methodologies that tackle the problem of human activity recognition. In this survey, we further analyze these two distinct and essential steps into categories with distinct characteristics and classify the reviewed methodologies according to which combination they follow. Features are decomposed to local, global and hybrid features and recognition techniques to knowledge-driven and data-driven, which are further decomposed to generative and discriminative

methods, since they have significant differences. Brief descriptions of the aforementioned categories are provided in the following sections and Table 3.1 shows the classification of the methodologies according to the aforementioned categories. The proposed classification method is generic enough to divide the categories into an easy to view table, while each category is meaningful and retains specific characteristics. The arrows above on the top and on the left depict a general qualification of important aspects of the feature extraction and recognition methods and are described in the following paragraphs.



Features	Recognition		
	Knowledge-Driven	Data-Driven	
		Generative	Discriminative
Global	[18], [31], [41], [45], [50], [65], [88], [147], [149], [183], [184], [185], [192], [193], [194], [195], [196], [205], [206], [207], [208]	[16], [22], [23], [25], [27], [38], [43], [54], [60], [63], [86], [151], [168], [174]	[15], [17]. [21], [24], [26], [28], [29], [30], [32], [33], [34], [40], [42], [47], [48], [49], [51], [68], [69], [70], [73], [81], [84], [85], [87], [90], [93], [94], [148], [150], [188]
Hybrid		[186]	[20], [35], [53], [56], [61], [74], [181], [120], [107], [164], [165],
Local		[44], [55], [71], [72], [112], [114]	[19], [36], [37], [39], [52], [57], [58], [62], [64], [66], [67], [75], [76], [95], [96], [97], [98], [103], [104], [106], [108], [109], [110], [111], [113], [116], [133], [134], [136], [142], [143], [144], [152], [153], [154], [156], [171], [172], [177], [178], [179], [180], [187]

Table 3.1: Classification of the methodologies

3.2.1 Features

As mentioned earlier, features can be classified into three broad categories, namely global, local and hybrid, with latter being the combination of the first two.

3.2.1.1 Global Features

Global features are object-centered and in our case they are mostly human-centered. They are usually applied on segmented regions depicting humans, so usually pre-processing steps are needed, such as background modeling, foreground extraction and tracking, in order to detect and localize the regions of interest. Traditional approaches describe the shape models of the human body or its parts. A very common approach is to extract general statistics from the silhouette or blob that represents the human in the scene, using its aspect ratio of height and width as seen in [47] and color histograms, express the contour using chain coding. A more detailed description can be also achieved through skeletonization of the human area, which provides more understanding of the postures and pose evolution. Finally, these models can be expanded to three dimensions where the building blocks become voxels, volumetric descriptions of the model parts. Except for the shape, global features over describe the objects motion too, for example by estimating the trajectory of the blobs centroid using optical flow, as seen in [24]. These models are mostly used in multiview systems, in order to provide more robustness against view changes. Global features are chosen in general because they offer an adequate level of information for a low computational cost.

3.2.1.2 Local Features

On the other hand, recent methodologies are based on local features, which are argued to be more robust to noise. The main shortcoming of the global features are its dependence to the preprocessing steps, which are open research problems themselves and can produce noise and ambiguities that will propagate to the feature extraction process and hinder its performance. Another limitation is that they are often too restrictive, as they are inherently connected with the defined human model and have to comply with its definition. These limitations are mitigated in the case of local features, which basically describe points of interest and their local neighborhoods through the whole scene. The main idea is that during the evolution of the image sequence, sets of local features will acquire a semantic meaning and manage to implicitly describe the humans and their motions, while irrelevant features will eventually be discarded. Feature detectors are used to search for corners, regions or other structure types and feature descriptors describe the neighborhoods of these features. Typical feature detectors are the Harris corner and edge detector proposed by Harris and Stephens in [100], or the Hough transform initially proposed by Hough in [101] and typical feature descriptors are the Scale-invariant feature transform (SIFT), developed by Lowe in [102] and cross-correlation. In the past few years, significant research has been conducted in order to augment these features and descriptors with time information, evolving them into spatio-temporal features. The most notable works concerning feature detectors have been proposed by Dollar et al. in [103], Jhuang et al. in

[57], Laptev and Lindberg in [104], Oikonomopoulos et al. in [116], Willems et al. in [106], Wong and Cipola in [107] and works concerning feature descriptors have been proposed by Klaser et al. in [108], Laptev and Lindberg in [109], Laptev et al. in [110], Scovanner et al. in [111]. Local features have been successfully used along with the Bag of Words (BoW) or Bag of Features (BoF) approach. One example is the framework proposed by Fei-Fei and Perona in [112]. This approach is adopted from the fields of natural language processing (NLP) and information retrieval and is based on the construction of codewords that adequately describe the scene. The main shortcomings of these features are the increase they often impose to computational complexity and the fact that their nature can be too generic, leading in production of features that are of no use.

3.2.2 Recognition

By recognition here, we mean the final methods employed to reach the final decisions and labeling of the activities observed in the scene. These methods, as aforementioned, are broadly categorized as knowledge-driven and data-driven.

3.2.2.1 Knowledge-Driven Approaches

Knowledge-driven approaches attempt to explicitly capture and engineer domain knowledge, through human experts. In this approach, formal models represent activities and sensor data and activity recognition is mapped to inference and reasoning methods, e.g. induction, deduction and abduction. Thus, a prior rule base is mandatory to react to the incoming information. A formal theory of plan recognition has been presented in the PhD thesis of Kautz in [197], where plan recognition is defined as a logical inference process of circumscription. Abstraction, decomposition and functional relationships between types of events are encoded in an event hierarchy, represented in first-order logic (FOL). One category of knowledge-driven approaches is the process-based activity modeling, which makes use of logical knowledge representation formalisms. Works that follow this path are these of Kautz in [198], who follows the aforementioned approach of event hierarchies and its extension by Wobke in [199], who uses situation theory to address the different probabilities of inferred plans. Plan recognition in the work of Bouchard and Giroux in [200] is carried out by a combination of action Description Logic (DL) and lattice theory, while Augusto and Nugent in [201] use temporal reasoning and Chen et al. in [202] use event calculus. Another approach of knowledge-driven approaches is the state based activity modeling, with major representatives the works of Tapia et al. in [203], who mine textual descriptions from the Wordnet ontology and Chen and Nugent in [204], who use ontological modeling and reasoning. Formalization of action models is also achieved through syntactic approaches, as seen in the works of Moore and Essa in [205], who use context-

free grammars (CFG) for recognition of multi-object events and Joo and Chellapa in [206], who use attribute grammars for the same task. Finally, a very powerful graphical tool to encompass domain knowledge is the Petri Net and its variations, e.g. stochastic, probabilistic. Examples of works that use Petri Nets for activity recognition are the framework proposed by Albanese et al. in [207] and Castel et al. in [208]. In overall, knowledge-driven approaches tend to build powerful and expressive models that are accurate for a specific domain and allow high level inference, while at the same time follow a tractable and interpretable process that allows better fault diagnosis. Especially after the improvements on description logic and the emergence of ontologies, reusability of and conveyance of knowledge has become easier. However, since they are heavily dependent on the experts knowledge, they usually tend to limit the domains scope and require notable manual effort for the knowledge representation, which hinders their scalability. Another shortcoming often attributed to knowledge-driven models is their inherent inability to cope with probabilities, since inference is normally deterministic.

3.2.2.2 Data-Driven Approaches

The second broad category of recognition approaches is the data-driven approach, which is where the majority of the recent methodologies fall into. As mentioned before, these approaches model the domain knowledge implicitly by learning descriptive relations that emanate from training data, which represent a fraction of the world and focus in capturing the details of specific domains. Many examples of this category are summarized in this survey. Data-driven approaches can be further divided into two subcategories, namely generative and discriminative approaches.

Generative model-based approaches are more traditional approaches that employ statistical models and perform random generation of observable data, based on some hidden parameters. Their basic operation is the estimation of the joint probability distribution over observation and label sequences and form full probabilistic models, by explicitly modeling the relations between the observations and the class labels. However, dependency assumptions have to be made. Discriminative models, on the other hand, avoid making independence assumptions among the observations. Their inference lies on a statistical framework that models the conditional distribution over observation to label sequences and attempt to directly model the discriminative boundary between the different class labels. Typical examples of generative models are Gaussian mixture models and other types of mixture models, Hidden Markov models, Naive Bayes, Latent Dirichlet allocation, Bayesian Networks and Markov Random Fields. Typical examples of discriminative models are Logistic and Linear Regression, Linear Discriminant Analysis, Support Vector Machines, Boosting techniques, Conditional Random Fields and Neural Networks.

In general, generative models capture well the intrinsic parameters that describe the world and can be accurate in a specific domain. However, these models make independence assumptions among the variables they model, which one hand can greatly reduce models parameters [10, 11], but restrict the models performance in the case they violate real-worlds conditions [105]. Since as aforementioned generative models give a holistic approximation of a domain, their functionality is easier to understand, which leads to better fault diagnosis. When the domain is complicated though, they have to follow its complexity, so they need a well defined and specific application domain to remain tractable. Discriminative models have a different perspective of how to relate observed data to label classes and in a sense compensate the functionality of the generative models. Due to the fact that they avoid making independency assumptions and are more robust to real-world and model mismatch. They are in general simple to construct and fast, but they do not offer a good insight to their inner workings, so they are not easy to interpret. Their major disadvantage is that they require a large amount of training data in order to capture adequately the domains informative cues, whereas generative models can cope with unlabeled data and be applicable to unsupervised learning.

3.3 Methodologies

3.3.1 Data-Driven Approaches

Data-driven approaches can be further divided into two distinct categories, namely generative and discriminative approaches.

3.3.1.1 Discriminative Approaches

a) Using Global Features

Activity recognition is achieved by Yi and Krim in [15] via monitoring paths of silhouettes and mapping them to known paths using homotopy transformation. The methodologys main feature is the path delineated by human silhouettes movement and it is matched probabilistically with stored paths of known behaviors. Special care has been taken to mitigate the ambiguity and discontinuity in homotopy paths, which are sampled during processing with Niquist sampling rate. The main focus of this system lies on the dimensionality reduction and high compression of information, although the description of the steps is not thorough. Moreover, the system does not include image processing, but rather uses Iranis group dataset [53] where human silhouettes are already segmented and clear.

The goal of Nater et al. in [17] is to design a system that primarily aims in assisting living of elderly or people at risk in general. Here, normality of an action is based on the frequency it occurs. The methodology introduces an unsupervised approach that employs two hierarchical structures,

inspired by the work of Fidler et al. [9]. The first one is about clustering the feature vectors that describe the appearance of the monitored human (blob) using the k-means approach, in a top-down hierarchy. The second hierarchy contains formations according to a bottom-up approach, where as the levels develop, more complex actions are considered. Blobs are extracted using background subtraction, rescaled and distance transformed and tracked using the mode estimating tracker of Bradski in [77]. The temporal relations of the symbol sequences, representing the postures, define the actions. Classification of normal or abnormal events is done according to frequency of occurrence, meaning that the larger the length of the symbol sequences, the more usual the action.

The surveillance system proposed by Lao et al. in [21] aims in covering the average consumers needs and experiments include simulations of shoplifting. The background is subtracted using the Gaussian Mixture Models technique of Zivkovic and van der Heijden in [78] and multiple persons are identified with the k-Nearest Neighbor classification method. A mean-shift tracker is also employed using color histograms. The trajectories are calculated using the Double Exponential Smoothing operator proposed by Han et al. in [79], which is argued to be faster than the widely used Kalman filter. Body-based analysis is performed to classify the different postures in the form of silhouettes, described using the HV-PCA shape descriptor, proposed by the authors. Final classification is performed via a Continuous Hidden Markov Model. The interactions between two persons are recognized as the combinations of single events that satisfy some spatial-temporal constraints, defined by the temporal logic. The system's cost is low and operates in near real-time, deals with interactions among multiple subjects in the scene, which are visualized in 3D so as to be increase user friendliness. However, the 3D visualization is still done offline and requires four white lines to exist in the scene so as to calibrate its coordinates.

Recognizing complex activities in video sequences based on motion information in a simple and computationally efficient manner is the objective of Ahad et al. in [24], continuing their work appeared in Ahad et al. [81]. The subjects motion is captured using Motion History Images (MHI) and Motion Energy Images (MEI). The extracted features are the seven Hu moments, which are invariant to rotation and the central moment, which is invariant under translation. Optical flow is computed with the gradients method proposed by Christmas in [80] and applied together with median filtering so as to overcome the motion-overwriting problem due to self-occlusion. Finally, k-Nearest Neighbors algorithm for classification of the actions is employed. The methodology is intended to be easy to implement and maintain a good trade-off between performance and complexity. However, the outdoor and illumination changes scenes still pose challenges and the k-NN method does not lead to a very sophisticated classification scheme.

Fast, robust and efficient recognition of single actions in videos captured from a monocular

camera from different views is the subject of interest of Cherla et al. in [26]. Initially background subtraction [82, 83] and refinement with morphological operations are performed and the basic features are extracted, namely the contours width (normalized with scaling), displacements of the contour centroids and standard deviations. Width features for gait representation have been used by Kale et al. in [84] and by Liu et al. in [85] and are preferred compared to MHIs, which can only deal with linear changes in speed create confusions with similar motions. Principal Component Analysis (PCA) is used to select the five most discriminating vectors and Dynamic Time Warping (DTW) to calculate the optimal average template for each action and again using multiple features. One of the authors ideas is to weigh the second half of an activitys evolution in time more, to increase separability. Experimental results show that fusing two datasets from orthogonal views during training increases view-invariance. A good level of view invariance is expected without the corresponding increase in computational complexity, however the methodology is not yet concrete and does not achieve high recognition rates.

Another system for fall detection in smart homes is introduces by Zweng in [28]. Information gathered from four cameras is fused and fall is detected after a voting procedure. The goal is to increase robustness by using information from multiple sources. First, background modeling is conducted using the Color Mean and Variance approach, initially proposed by Wren et al. in [86] and shadows are removed after combining the RGB and NRGB color space. Simple tracking of the blob with the appropriate size takes place after these steps. The features used in the methodology are the length/width ratio, orientation and axis ratio and velocity of the blob. The authors use the statistical model (accumulation hitmap) of Ermis et al. in [87] to recognize falling after recognizing three postures, namely standing, in between and lying, inspired by Anderson et al. in [88]. The system has also been designed to learn zones of usual activity (e.g. bed). Conclusively, it is a simple and computationally inexpensive approach, offering robustness against occlusions and clutter. However the features used and the postures recognized are simplistic and so is the reasoning, while it could be argued that there is a redundancy in information.

Human action recognition performed by Kim et al. in [29], is based on an ordinal measure of accumulated motion. Event retrieval in videos captured with a static camera is performed, where a video segment containing an action is treated as template, the characteristics of which are queried in target videos. As a descriptor of the humans in the scene, the Accumulated Motion Image (AMI) feature is proposed. This feature indicates the areas where motion is most frequently observed and is combined with the proposed ordinal measure for matching similar actions and it is argued to lead to more efficient computations as opposed to 3D spatiotemporal features. AMI is first horizontally and vertically projected to reveal the energy histograms in the two directions. The energy histograms

are normalized and then the similarity is measured between those of the candidates and of the query action video. The methods advantages are its simplicity, real-time operability, robustness to occlusions, clutter, similar actions and appearances and elimination of the need for preprocessing of the data, such as segmentation and learning. Its disadvantages are the lack of support for dynamic scenes and the potential limitations to actions of small length.

Foroughi et al. in [30] describe a methodology for video surveillance in smart homes, using a single camera to recognize daily activities of a person, unusual events and mostly fall incidents. Effort has been made to able to recognize different types of falls and complex actions reliably, while at the same time keeping the false positives rate limited. The human's silhouette is extracted using the fuzzy background subtraction method of Shakeri et al. in [89]. The chosen features have been proposed by Wang in [90]. They derive from the fitted ellipse, horizontally and vertically projected histograms of the foreground images (normalized with Discrete Fourier transform) and temporal changes of the head position (top-most point of silhouette). Significant change in the standard deviation of the orientation or ratio of length/width of the eclipse, most probably indicate a fall incident. The recognition takes place in a supervised four-layered MLP Neural Network with back-propagation learning schema. The system is argued to be robust to shadows, illumination change and variation of postures and achieves high image compression. However, simple features are used and assumptions have been made to simplify the situations, as training the Neural Network is already a difficult task.

Hazelhoff et al. in [32] propose a methodology for detecting fall incidents occurring inside a room, captured with two cameras with perpendicular views. The goal is the creation of a fast and computationally inexpensive system that can tolerate occlusions. Extraction of the foreground pixels is done using Gaussian Mixture Model (GMM) for the background modeling. Moving objects that have certain size and no head region attached to them are marked as non-human. The head is defined as the highest skin region and it is marked and tracked (position and motion). Then, Principal Component Analysis (PCA) is adopted as the basic component of the methodology. PCA produces the ratio of the variances in horizontal and vertical direction, as well as the angle between the main axis of the human body and the vertical camera axis (two features gathered from each camera). Multi-frame Gaussian classifier is finally trained to distinguish fall and non-fall events. Recognition process requires lying posture to be detected and inactivity zones to be manually marked in the scene. Its main merits are it tolerance to occlusion and robustness against cluttered background and illumination changes. However, the crucial to systems performance head detection and tracking is still simple and manual intervention is required for the correct operation.

A distributed camera network architecture is proposed by Song et al. in [34], designed to per-

form the tasks of tracking humans and recognizing their activities in large outdoor areas. In this approach, each camera is deemed as an agent and cameras communicate with neighboring cameras with the same FOV, based on the framework previously proposed by Soto et al. in [92]. A modified version of the Consensus Kalman Filters algorithm, initially proposed by Olfati-Saber in [91], is used for refinement and handoff of locations of humans. Consensus-based posture recognition follows, which estimates probabilistic similarities of shapes to a predefined dictionary. Shape-based activity recognition, similarly to the one described Veeraraghavan et al. in [93] is the final component of the methodology. Both the architecture and the chosen algorithms support the expected systems robustness, reliability and scalability, but it is still a proof of concept and the authors make assumptions for the network communication. In future work tracking and recognition will be fused.

Meng et al. in [40] use Motion History Image (MHI), Modified Motion History Image (MMHI) and Motion Gradient orientation (MGO) as their features and perform Principal Component Analysis (PCA) to reduce their high dimensionality. Six binary SVMs are trained, each one corresponding to an activity, using the one-versus-all method and the winner-takes-all strategy. The experimental results showed that the simple MHI had the best performance, due to its ability to handle better the noise of cluttered scenes, while the MMHI feature approached had a slightly increased performance in the case where 4x4 down-sampling was used, comparable on average to MHIs in the previous case. The methodology is simple and computationally inexpensive and there is no need for parameter setting, but these come to the cost of limiting recognition to simple actions in a homogenous scene.

Huang et al. in [42] base the modeling of human behavior on Discriminative Random Fields (DRF), inspired by the work of Wang et al. in [94], who introduced the Hidden Conditional Random Field (HCRF) for gesture recognition. The motion observations are mapped to category label variables and an undirected graph is constructed, where the nodes are the behavior features and the edges the relationships among them. The conditional probabilities for the model are then computed from the feature functions between the labels and the hidden states. A skeleton model is proposed as the humans appearance model and the extracted features describe the general bodys posture and the more discrete gestures of the limbs. The parameters for the model are learnt from the training dataset using the Quasi-Newton method and the inference is carried out via belief propagation. The methodology has a good computational complexity, can model long term contextual dependencies among observations and achieve better recognition rate than the classic HMM and CRF. The authors, however, are concerned only with the recognition problem and not the image processing.

Fleck et al. in [47], describe the architecture of a network of smart cameras, installed in the corridors of a building to aid assisted living. The smart camera network is considered to be distributed

in the sense that every camera has processing power and performs the tracking and the activity recognition autonomously and only processed results are sent to the central server wirelessly. The tracking is based in adaptive background segmentation and multiple particle filter estimation for each moving object that belongs to the foreground. Since the cameras are static and calibrated, the results can easily be mapped to the 3D world. The features that describe the silhouettes are a combination Aspect Ratio, Aspect Ratio Differences and Histogram Differences and are fed to SVM classifiers running in each of the systems cameras to perform fall detection. Great effort has been made in visualizing the recognized events, which combined with its holistic architecture and modularity constitute a suitable solution for industry.

The work presented by Khalid in [48] incorporates novel techniques to achieve accurate unsupervised learning of patterns in the presence of anomalies in training data. Emphasis has also been given on keeping the space and time complexities low. The features to be processed derive from the trajectories of humans, which are treated as motion time series. More specifically, trajectories are represented using a modified version of the Discrete Fourier Transform (DFT-MOD), which augments the DFT coefficients-based feature vector with information regarding the length and starting location of the trajectory. The extracted features are clustered with a cooperative learning algorithm that combines learning vector quantization (LVQ) with Hierarchical Semi-Agglomerative ClusTering (HSACT), which enables it to discover the actual clusters and avoid being stuck in local minima. Each learning step is iterative and each iteration filters out anomalous trajectories from the training data, which are trajectories that belong to the learned clusters with fewer cluster memberships. Another contribution of this work is the proposal of a mechanism for modeling various patterns that are present in the motion data set, called m-Medoids. The classification of trajectories is performed by the k Nearest Medoids (k-NM) technique, which follows the same principles as the well-known k-Nearest Neighbors approach. In overall, the clustering method offers improved results with low computational cost, although it is sensitive to the presence of very high number of anomalies.

Zhang and Gong in [49] propose a method for silhouette-based action categorization, by treating action images as a whole, rather than identifying the detailed body configurations. To this end, the authors propose a combination of Conditional Random Fields and Hidden Markov Models, referred as modified HCRF (mHCRF) based on HMM pathing. The systems input are clearly segmented binary silhouettes from a static background, described by the Chain Coding approach. In fact, the Fourier Transform of the latter is computed, so as to augment the resulting spectrum features with rotation invariance. To overcome ambiguities caused due to subtle body shape changes, motion moment features are also computed with frame differencing. The core of this publication is the mHCRF algorithm, which eliminates a major drawback of the HCRF technique, namely the sensitivity to

the initial parameter selection. In mHCRF this problem is solved by making the hidden variables observable, which in turn ensures the convexity of the objective function. Conclusively, the proposed methodology introduces an improved way to model actions.

Wu et al. in [51] introduce a discriminative model capable of online updating for action recognition, enhanced with semi-supervised learning so as to handle partially labeled training sets. The main technique used is an enhancement of Discriminant-analysis of Canonical Correlations (DCC), described by Kim et al. in [95], referred to as Incremental Discriminant-analysis of Canonical Correlations (IDCC), which is able to update the discriminant transformation matrix for classification when new training data is being added. IDCC is fused with a semi-supervised learning method, namely the linear neighborhood propagation (LNP), in order to cope with both labeled and unlabeled training data and augment it by allowing it to simultaneously label new training sets. The methodology was also specifically tested for its robustness against occlusions, different styles and conditions in walking sequences and showed that the online re-training improves the recognition over time and manages to cope with them. Recognition is expected to be more accurate if temporal information and non-linear learning will also be taken into account.

An interesting approach for unsupervised learning of activities based on a linear dynamical framework is proposed by Turaga et al. in [68]. The focus of this method is the automatic temporal segmentation of complex actions, which are modeled as cascades of linear time invariant dynamical models. The resulting dynamical systems are clustered into action prototypes, which in turn form the cascade using the n-grams learning technique. This method can be viewed as a grammar based approach, where the production rules construct actions from sequences of action prototypes. One of the appealing aspects of this work is the incorporation of view and rate invariance into the models, but achieved in the learning (clustering) stage rather than the feature extraction stage.

Another methodology for unsupervised learning of human action classes has proposed earlier by Wang et al. in [69]. This approach differs significantly from the previous one, since here information is extracted solely from static images and thus no temporal relations can be known. The basic idea lies on grouping images depicting humans with similar body poses and matching them to manually labeled action prototypes. Poses are formed from the responses of Canny edge detection and matched using a deformable matching method that is based upon a linear programming relaxation technique. However, this method is computationally expensive and the authors propose pruning the search space using shape contexts. Action classes are finally produced by spectral clustering of the pairwise distances and are the prototypes that represent the clusters are manually labeled, for further classification of newly seen images.

Wang and Mori in [70], where they expand their work appearing in [96], base their action recogni-

tion methodology upon the bag of words approach, with the difference that each frame is represented by a word. One of their main arguments is that their global motion descriptors are more effective than the local patches of information usually used with the aforementioned approach. The models used for visual recognition are the Semi-Latent Dirichlet Allocation (S-LDA) and Semi-latent Correlated Topic Model (S-CTM), which augment the traditional LDA and CTM models of Blei et al. in [97] and Lafferty and Blei in [98], respectively, by observing some of the latent variables during training. Temporal relations are recovered via the co-occurrence statistics amongst visual words and whole video sequences are used during the classification process, instead of single words. Their work shows high recognition rates, but in not complex datasets by assuming tracking and stability of the human figures. In a more recent work from the same group, Lan et al. in [99] overcome the latter assumption and introduce a discriminative model that couples activity recognition with person detection and tracking.

Gong and Medioni in [73], use labeled motion capture (Mocap) data to learn view invariant 3D models of human motion in the joint trajectory space. During the offline training phase the submanifolds in the video data are represented by a latent variable model (LVM), extended to a spatio-temporal manifold (STM) model that additionally includes the temporal dimension. The learnt manifold structures are used for recognition of actions, using method called Dynamic Manifold Warping (DMW) to align multivariate time series and matches are found based on a similarity score. The main features of the method, the joint trajectories and 3D skeleton figures are extracted from partially occluded tracks. However, this step is sensitive to noise and restricts the methodologys full potential.

Zhou et al. in [33] on the other hand use a fish-eye camera mounted on the ceiling of a living room, in real environment conditions. An intelligent background subtraction method is initially applied, based on brightness and chromaticity distortion setting and decision rules. For the background modeling Gaussian mixtures are used and thresholding based on maximum likelihood. The extracted humans are tracked with an algorithm that employs a speed up method called diamond search and the silhouettes location and speed are estimated. At this point, the authors use their adaptive fuzzy inference system (ANFIS) to refine the results and a Hierarchical Action Decision Tree (HADT), which provides different granularities of recognition for multiple-level features. The system achieves robustness against illumination, shadows, lens distortion and occlusions, while maintaining real-time functionality.

b) Using Local Features

Zweng and Kampel in [19], study the problem of unexpected behavior recognition, focusing on high-density crowd scene. Additionally, they test the applicability of the method in fall detection

for a single human. The input comes from a single camera and Average of Gaussian algorithm is used as the background model, enhanced with shadow removal. The extracted features emerge from the accumulated hitmap, which shows the areas in the sequence where motion takes place the most. Crowd pace and density are then estimated, along with the areas of low walking frequency.

Hofmann and Gavrilla in [37] present an introduction to a framework for unconstrained 3D human upper body pose estimation from multiple camera views in complex environment, extending previous work of Hofmann and Gavrilla in [113]. In the beginning, a rough ROI is determined via foreground segmentation and volume carving, so as to exclude moving regions due to clutter in the next steps. In the hypothesis generation step, the resulting blobs are matched in a hierarchical manner to pre-computed 2D pose exemplars containing silhouette data in the individual camera views and are mapped to corresponding 3D poses. The observations and an action model are combined in a Viterbi-like maximum likelihood approach to compute the K-best trajectories, which are used for pose selection. The selected poses in turn are used to generate and adapt a texture model that enriches the shape likelihood measure used for pose recovery. At this point the pose prediction is performed, by using the multiple hypotheses and as a result, the 3D pose candidates generated by single frame pose recovery at the next step are augmented. Great effort has been made in maintaining the 3D pose search space low, by ruling out unfavorable solutions. The resulting system can cope with complex and dynamic environment, uses coupled tracking and pose estimation and does not require initialization pre-operation process.

Lavee et al. in [39] describe a framework for detecting user-defined suspicious events in a vast amount of video data. The features come from the RGB color space and the system also supports four levels of temporal granularity, by setting four different sampling values. For each temporal scale three-dimensional matrices are calculated, containing the intensity gradients values in the three directions, representing the distribution of change of the observed motion. Four classifiers are used (Nearest Neighbor Algorithm using Histogram Distance Function, proposed by Zelnik-Manor and Irani in [114], Euclidean distance, Neural Networks and Decision tree) and the results are evaluated. Due to the high dimensionality of the input data, Nearest Neighbor classification algorithm using the histogram distance measure shows the best performance, coping better with them and Decision trees the worst performance. Bayesian Networks and Support Vector Machines are not chosen because they require an extra step to set their parameters. The system is extended further with the addition of storing high-level interpretations in XML files can be visualized and processed.

A set of novel visual descriptors based on B-splines forms the foundation of the human activity recognition methodology thoroughly described by Oikonomopoulos et al. in [52]. The optical flow method designed by Black and Anandan in [115] is used to detect spatiotemporal interest points

with the algorithm of Oikonomopoulos et al. in [116], and median filtering is then applied to deal with general camera motion. A low degree B-spline polynomial is then fitted to these points and the sequences of images are represented as collections of B-spline surfaces. The final spatiotemporal feature descriptor vectors are single histograms of their partial derivatives, which are clustered using the K-means method to form a codebook. The codebook in turn is refined using the GentleBoost algorithm of Friedman et al. in [118] and given as input to a Relevance Vector Machine, proposed by Tipping in [117], for the final classification. This approach does not use or require image preprocessing or background subtraction prior to feature extraction. Although many of the methodologys steps are computationally expensive, measures have been taken to reduce the needed computational load. The main advantage of the methodology is its robustness to several factors and good handling of unknown cases. Further future research should include more extensive experiments on complex cases, where the background is dynamic and several actors are in the scene, as the authors report.

An interesting approach for human activity recognition comes from biologically inspired systems, propelled by research on how the human visual system works. Juang et al. in [57] propose a hierarchical feed-forward architecture, previously used in [142, 143, 144], where features are formed hierarchically and in each level they increase in complexity and invariance to scale and position. Their model only considers motion, which is initially captured by Gabor filters of multiple orientations, applied to flow vectors and a local max operation increases their tolerance to position. The next level of features is the result of the previous filter responses and template matching with stored prototypes. Frame-based classification is done at this level. A similar process, template matching and global max operation, granting shift-in-time invariance, compose the final video-based classification, where feature selection with zero-norm SVM is performed. Although the model is able to represent smooth actions, it suffers from high computational complexity, especially in the first stages of the feature selection. To mitigate this problem as much as possible, the authors also perform background subtraction using GMMs as a preprocessing step and more importantly, they propose the use of sparse features in the intermediate steps of the overall process.

An extension of the work of Jhuang et al. [57] is presented by Escobar et al. in [58], where the initial steps of the methodology are more directly connected to the human biology. Specifically, they use a feed-forward spiking network of Perkel and Bullock in [145], emulating the responses and communication of the two brain areas dedicated to motion. First, directional-selectivity filters are applied over each frame of the input sequence in a log-polar distribution grid obtaining spike trains as V1 output and these spike trains are processed in the MT spiking model, which provides the spatio-temporal relations. The motion information is obtained through the mean firing rates of MT spike trains or a synchrony map of the spikes trains generated by MT cells and encapsulated

in motion maps. The latter are used as input for trained CRFs, which perform the final action classification. One addition over the work of Jhuang et al. in [57] is that interactions between cells are also considered and more complex actions can be represented.

Action recognition from unconstrained videos is a very difficult problem, due to the amount of noise and appearance variations this case presents. Liu et al. in [61], propose using dense static (Harris-Laplacian (HAR), Hessian-Laplacian (HES), and MSER detectors [146]) and motion (spatiotemporal interest point detector proposed by Dollar et al. [103]) to provide an accurate spatio-temporal representation of regions of interest. Noisy features are pruned using motion statistics and the PageRank technique. Discriminative semantic visual vocabularies are then learnt using an information-theoretic divisive algorithm, which encompass the compressed useful information. Action in videos is represented by the histogram of bag of visual words and Adaboost is used for the final action recognition.

One of the most prominent problems in learning of activity patterns is the preparation of the training set. Marszalek et al. in [62] suggest using movie scripts for automatic video annotation, following the works of [133, 134, 110]. Additionally, they aim in exploiting the correlations between actions and scenes and employ a combination of script-to-video alignment and text search using Wordnet [135], to recover the co-occurrences of actions and scenes. Action is represented with static, dynamic and motion descriptors (2D [137] and 3D Harris detector [136], HoF and HoG descriptors [110], SIFT descriptor [138]), used to form visual vocabularies [139]. Finally, video samples are represented with bag-of-features (BoF) method [140] and SVMs [141] are used for the final action recognition.

Rapantzikos et al. in [64] propose using cuboid features, as those seen in Liu et al. in [61] and Bregonzio et al. in [56] and argue that intensities alone, as seen in the works of Dollar et al. in [103] and Laptev in [136] are not descriptive enough, so they propose augmenting them with color and motion information. Visual input is represented by a volume, which is decomposed into conspicuity features, which in turn are decomposed into multiple scales. Saliency is used for the final feature point detection, which is calculated by a global minimization process, constrained by proximity, scale and feature similarity. This representation is argued to offer a good balance between information and complexity. Recognition is performed using the k-NN classifier in bag of words fashion.

Seo and Milanfar in [66] and [152], treat activity detection as a matching query of short videos containing actions of interest to longer target videos. The operation of this approach deviates from the standard human activity recognition framework, but has the advantage of being non-parametric training-free. One of the main points of this work is the use of space-time local steering kernels (3D LSK), as also seen in [153, 154, 155], which are in general robust to noise and expressive. Here,

Principal Component Analysis (PCA) has been chosen to reduce their complexity. Detection of actions is based in Matrix Cosine Similarity (MCS) measure, between the query and target videos.

Sun et al. in [67] propose a methodology based on SIFT descriptors [138] to extract trajectories of salient points and model the spatio-temporal context information encoded in unconstrained videos. The low level features are processed through a three level hierarchy, where the first level deals with point-level context, using SIFT average descriptors, the second level with intra-trajectory context, using trajectory transition descriptors and the third level with inter-trajectory context, using trajectory proximity descriptors. The two latter levels are encoded with the bag of words method into the transition matrix of Markov process. The final fusion of features is carried out with multi-channel nonlinear SVMs, similarly to Laptev et al. in [110], while the MKL technique is applied to mitigate the computational load this method imposes.

Tackling the very difficult problems that emerge from camera motion is the main objective of Wu et al. in [75] and their methodology revolves around robust trajectory extraction, without relying on standard preprocessing steps, such as motion compensation. Thus, they propose a Lagrangian particle trajectory acquisition approach, inspired by the particle trajectories used by Wu et al. in [156] for crowd flow analysis. This method eliminates the need for pre-definition of interest points and point correspondence across frames and at the same time enables the extraction of independent object motion using rank optimization. The final recognition of actions is based on Support Vector Machines (SVM).

Chakraborty et al. in [76], propose another methodology based upon the Bag of Visual Words (BoV) model aiming primarily in robustness to several factors that are prominent in videos of unconstrained conditions. In order to do so, they detect selectively spatio-temporal interest points (STIP) and apply surround suppression combined with local and temporal constraints, following the ideas of Grigorescu et al. in [157] and Lindeberg in [158], so as to weed out the less informative features. The features are described by local N-jets, developed by Koenderink and Doorn in [159]. The BoV model used to learn the visual vocabularies along with the method to compress them are borrowed from Liu et al. in [61] and extended with pyramidal levels in the feature space. Finally, recognition is carried out with Support Vector Machine (SVM) classifiers.

Another important issue, that of recognition human actions under different views, is addressed by Junejo et al. in [36] with the use of an interesting action descriptor based on similarity matrices. The main motivation comes from the observation that self-similarities of action sequences over time demonstrate notable stability across different views. The action recognition follows the Bag of Features (BoF) approach where each video is represented as a set of quantized local SSM descriptors with their temporal positioning in the sequence being discarded. Each image sequence is described

by a normalized histogram of visual words and two types of classifiers, a Nearest Neighbor classifier and a Support Vector Machine perform the recognition, using these sequences as an input. The methodology achieves view independence without the need for structure recovery or multi-view correspondence and focuses in disambiguating similar activities, assuming accurate tracking of the points of interest.

b) Using Hybrid Features

The study of Wu et al. in [35] deals with the problem of subtle human activity recognition in smart homes and investigates the merits and challenges of three types of data fusion. A hierarchical approach and a combination of spatio-temporal features coming from multiple views of a scene is argued to be more efficient in recognizing subtle activities, such as reading, based on the recognition of coarser activities, such as walking. The authors use test bed smart environment a special smart studio, located at Stanford University, called AIR (Ambient Intelligent Research) Lab, equipped with furniture and everyday appliances and six cameras that cover its area. The activity recognition is based in a two-hierarchy, with the first level recognizing coarse activities and the second one extending the recognition to finer activities, by making use of additional features and information. Features emanating from foreground extraction, namely the height of the human (through 3D tracking) and the aspect ratio of the humans bounding box are fed into a temporal conditional random field (CRF) that classifies the coarse actions. Then, based on the result of the coarse level, the activity is further classified at the second level based on spatio-temporal features, the results of a face detector and the location context (i.e. kitchen, living room etc.). The Bag of Features (BoF) approach is employed to represent the data in the form of histograms of spatio-temporal features and Support Vector Machines (SVM) carry out classification of actions. The second part of the methodology is a presentation and comparison of three types of data fusion in multi-camera systems, namely best view, mixed-view and combined-view, the first two performing decision fusion and the latter feature fusion.

Hu et al. in [60] focus their interest on cases of complex scenes where multiple complex actions take place. In their experiments include videos from the CMU dataset and from a real shopping mall surveillance dataset, where additionally they propose a way to detect shopping interest. In their work it is shown that combining appearance (foreground image from background subtraction [160] and HOG [161]) and motion (MHI [162]) features give better results than relying solely in one feature type. Action recognition is performed with their SMILE-SVM (Simulated annealing Multiple Instance Learning Support Vector Machines). They are SVMs inspired from [163] and extended with simulated annealing, so as to ensure convergence to global optimums and avoid model initialization issues. Another important feature of their work is that they require only rough annotation of the

training data, which is admittedly a tedious and time-consuming process.

Fathi and Mori in [120] use of biologically inspired features, built on the mid-level shapelet features proposed by Sabzmeydani and Mori in [121]. The main idea is to focus the feature detection on the area occupied by humans, so as to increase the descriptive power of the local features. To this end, the authors employ well-known algorithms for detection and tracking of the human subjects, which are represented by 3D spatiotemporal volumes. Given these volumes, first the low-level motion features are computed using the optical flow algorithm of Lucas and Kanade in [122] and they are refined into stronger, mid-level features after classification with the Adaboost method of Viola and Jones in [123]. Adaboost is then performed again, now on the mid-level features, in order to merge the local semantics they represent and discover the final, strong classification.

Bregonzio et al. in [56] address many of the limitations imposed by complex actions and dynamic background with their method, which solely relies on the global spatio-temporal distribution of the interest points. The interest points derive from frame differencing fused with the responses of 2D Gabor features of different orientations and foreground extraction is done using Prewitt Edge Detection [119]. Clouds of interest points are then formed for series of frames. The final features encapsulate absolute and relative height/width ratios and speed information of the objects and clouds of points. A feature selection of low computational cost is proposed and is performed on the feature set to refine it and finally the action recognition is achieved using Support Vector Machines (SVM) and Nearest Neighbor Classifiers (NNC).

A combination of appearance and dictionary-based is proposed by Qiu et al. in [74], where action attributes are learnt via information maximization. The features extracted are both local and global, borrowed from the works in [161, 164, 110, 165] so as to cope with static and dynamic backgrounds and camera motions and increase robustness to noise, occlusions and viewpoint changes. These features are encoded in a dictionary via the K-SVD [166] algorithm and optimized after maximization of the mutual information for appearance information and class distributions via a Gaussian Process (GP) model, both suitable for sparse representations. The sparse coding property leads to efficient dictionary learning, through selection of sets of compact and discriminative action attributes.

3.3.1.2 Generative Approaches

a) Using Global Features

A system for behavior recognition of multiple humans in a scene, using multiple fish-eye cameras, is introduced by Uson et al. in [16]. Utilization of multiple cameras and 3-D reconstruction aim in the elimination of ambiguities in observations. Each camera provides information about a partial model of the scene and they are merged to create a volumetric description of the scene. Spatio-

temporal features are extracted (i.e. center of gravity, axis-aligned minimal bounding box, etc.) and volumetric ROIs are treated as agents and their probabilistic behavior is modeled with a Hidden Markov Model dynamic Bayesian Network (HMM-DBN) that interprets the observations. This methodology is scalable, reliable and it can handle occlusions and multiple-person tracking, but it requires a large training dataset and the segmentation algorithm cannot cope with fast illumination changes.

In another approach by Huang et al. in [20], ambiguities in human behaviors due of the angle of observation are resolved differently. Three types of features, namely shape, trajectory and optical flow vectors, are fused in a Bayesian Network and the best ones are chosen. The background is modeled with Gaussian Mixture Models, using a minimum number of frames. Each detected object is represented by its center of mass and tracked using the nearest neighbor criterion and Kalman filters predict the location and size of the moving objects. A linear support vector machine is used for the classification of the scenes, which is trained with the SIFT features extracted from pedestrian blobs. It is in overall a computationally effective approach to achieve view independence, but it is still restrained to simple activities, while the data fusion might sometimes be imposing unnecessary load.

Silhouette extraction and modeling of actions with Hidden Markov Model are the main components the methodology of Martinez-Contreras et al. in [22], which addresses the case where the number of training samples is small. Motion History Images (MHI) are used for the representation of the human bodys motion evolution and Principal Component Analysis (PCA) and Kohonen Self Organizing feature Map (SOM) [167] for the classification. Hidden Markov Models (HMM) are used to model behavior on the temporal sequences of MHI. The lack of many action samples is compensated with a technique called Sampling Importance Resampling (SIR), which generates more samples.

A similar approach is followed by Hu et al. in [60], who use information of many cameras to perform fall detection. Background subtraction is performed in each camera to extract and track the monitored humans silhouette. A variant of the mixture of Gaussians modeling [83] is used for this purpose and the tracking algorithm is based on an appearance model. Two basic postures are recognized according to the angle between the main axis of the fitted ellipse and the vertical direction. The information of all cameras is taken into account and when there are significant differences among the estimated silhouettes, the most appropriate one is estimated through geometrical reasoning. Metric rectification of perspective images assures that the postures will look the same in both the image and real-world domain. The decisions of the cameras are fused with logical ORs between lengthened pose detectors and the Bayesian probabilities of the detected standing or lying pose are

calculated, which are treated as observations of a Layered Hidden Markov Model.

A more complex model based on variations of HMMs is introduced by Liu et al. in [23] that goes beyond human behavior recognition, considering also group interactions. The input dataset has been collected from a nursing home environment and poses the following challenges: maintaining robustness in real scene and situations, while managing complex interactions at low computational cost. In the image processing part background subtraction and trajectory tracking of humans with Kalman filters are performed. The image understanding part comprises a Switch Control (SC) module that extracts the atomic behavior units, an Individual Duration Hidden Markov Model (IDHMM) module used to recognize autonomous actions and an Interaction-Coupled Duration Hidden Markov Model (ICDHMM) module to recognize units that interact with each other. Interaction between two people is measured by how distance between them changes over time. A similar approach using layered HMMs has been proposed earlier by Zhang et al. in [168].

Trajectories of humans and HMMs are again used by Suzuki et al. in [27]. More specifically, the authors focus at learning of complex trajectory patterns, observed in a real environment for which they possess no prior knowledge, in an unsupervised manner. Hidden Markov models are used for modeling of the spatio-temporal features of the trajectories, which are projected to a space of lower dimensions, using Multi-Dimensional Scaling and the Young-Householder transform [169]. Finally, the motion patterns are categorized with K-Means clustering, using random sampling and Calinskis function value. In overall it is a straightforward and automatic approach, which is however restricted in distinguishing only normal and abnormal behaviors, arguing that distance among trajectories suffices for this cause and the features are restrictive for recognition of complex activities.

Chen et al. in [38] propose a method based on skeletonized silhouettes, described by the star-figure model, which outlines the positions of the limbs with respect to the silhouettes centroid. In order to overcome problems due to self-occlusions and ambiguities in silhouette, the unsupervised classifier ISODATA is employed, which automatically defines the appropriate number of points in the star-skeletons by merging, splitting and dropping clusters through an iterative procedure. The length and orientation of this representation model are the two dimensions of the corresponding Gaussian distributions. The maximum likelihood parameters of a mixture of k Gaussians in this feature space are calculated using the well-known Expectation-Maximization (EM) algorithm and the K-means algorithm for the initial estimation of the parameters. Recognition of actions is conducted via a similarity measure and matching with the stored models in the database.

Bruckner et al. in [43] present the SENSE (Smart Embedded Network of Sensing Entities) project, which is an 8-level hierarchy for human behavior recognition. It processes and passes the information gathered from visual and audio sensors through its levels, extracting incrementally more

semantic information about behaviors and situations. Features called Low-Level Symbols (LLS) are gathered in the first level and refined in the second using a mixture of Gaussians and fuzzy logic. Then, the third level deals with the tracking of those objects, using particle filtering and a motion model based on Markov Random Fields (MRF) to compensate for the first methods instability. These uni-modal symbols are fused in the fourth level with respect to their time correlation and the resulting multi-modal symbols are processed with an online version of Expectation Minimization (EM) algorithm, which reveals the parameters of the models behind them. In the fifth layer the mapping of their local trajectories is created in the form of local transition matrices. The sixth layer conducts the inter-communication of the SENSE networks nodes to define the global trajectories of the objects and which of the neighboring nodes their trajectories affect and also prepare information that might raise alarm events in the last layer. Loopy-Belief Propagation (LBP) algorithm is used to convey the messages among neighboring nodes. The last level of the hierarchy gives the high-level semantic interpretation of the observed events.

Xiang and Gong in [54] present a unified bottom-up and top-down approach to model complex activities of multiple objects in cluttered scenes. The foreground is extracted using the Pixel Energy History (PCH), an extension to Motion History Image (MHI) and object-independent events are clustered using the unsupervised Gaussian Mixture Models (GMM) method and classified using automatic model selection based on Schwarzs Bayesian Information Criterion (BIC) [170]. Formulating Dynamically Multi-Linked Hidden Markov Model (DML-HMM) conducts the behavior understanding in the scene level, in order to model the temporal and causal correlations among discrete events. This methodology, besides increased robustness, is able to model more complex activities than the ones usually considered by allowing a relaxation of the linear temporal order of the states.

Local patch methods, as seen in [171, 110, 172, 55], exhibit successful action description and recognition; however the complexity of motion they can represent is restricted by the locality of the features. In order to overcome these limitations, Messing et al. in [63] propose using more global spatial and temporal information and exploit the motion of tracked points on a body, inspired by the works of Johansson in [173] and Madabhushi and Aggarwal in [174]. Thus, they extract feature trajectories from the video sequence, using Birchfields implementation [175] of the KLT tracker [122] and call their basic feature velocity history. Activities are then modeled as distributions over mixtures of velocity history Markov chains. Although this features yields good results for high-resolution videos, a significant improvement can be achieved when augmenting extra information, namely absolute position (initial and final feature position), relative position with respect to face (using the Viola-Jones face detector [123]), local appearance (using PCA-SIFT [176]) and color histograms.

Natarajan et al. in [25] base the recognition of complex actions upon lifting 2D images from video streams to 3D action models. The authors intention is to eliminate the limitations of dataset dependence and sensitivity to view and scale variations. The methodology is based on the decomposition of a composite action into a sequence of primitive actions. Representation of actions, inspired by Lamport in [190], is in form of parametric functions and their parameters are learned by lifting 2D pose annotations in a few key frames to 3D, similarly to Taylor in [191] and interpolating between them. The event models are mapped to a Dynamic Bayesian Network (DBN) and the relative weights of the different features are learned using a novel Latent State Perceptron Algorithm. The combination of these learning methods significantly reduces training requirements. Finally, a novel inference algorithm estimates the action sequence by sampling the action models and accumulating the feature weights.

a) Using Local Features

Gupta et al. in [44] use edge and skin detection and background subtraction, so as to detect the multiple moving objects and the human in the scene (GrabCut is used for the human segmentation). The type and location of objects is estimated using a variant of the histogram of oriented gradient (HOG) approach and a cascade of Adaboost classifiers. Actions are perceived through modeling of the sequence of three classes of sub-activities, mainly reaching, grasping and manipulation of an object by the human. The observation of the velocity of the hand trajectories defines the reach and manipulation motions and the likelihood of the recognized action is calculated by HMMs, specifically trained for this purpose. A Bayesian model is employed to indicate the relations between the scene objects, the manipulable objects and the human in action. The probability of the scene is decided by a SVM classifier and the likelihood of the scene object by an Adaboost classifier. In order to define the humans actions, an upper-body pose estimation algorithm is employed, which segments and tracks the torso and the hands in each frame, while the pose is classified with the K-Nearest Neighbors rule.

Savarese et al. in [55] achieve very high classification accuracy on the KTH dataset, under an unsupervised learning scheme. Their work extends previous bag of video words approaches [103, 177, 154], by using the idea of correlograms, initially proposed by Savarese et al. in [178] to describe co-occurrences of codewords within spatio-temporal neighborhoods. First, the vocabulary of video words is formed, based on K-means clustering over descriptors emanating from the application of separable linear filters on the video sequences. Then, for each pair of video word labels the corresponding ST-correlogram element is computed by using a given set of kernels. The optimal size and size of the kernels needs further examination. These descriptors are clustered using again the K-means clustering method, to form the ST-correlatons. These features can encode flexible long-

range temporal information into the spatial-temporal motion features, resulting in rich description of human actions. The different classes of human actions are learnt by an unsupervised generative model in accordance with the probabilistic Latent Semantic Analysis (pLSA).

Brendel and Todorovich in [71] propose unified framework for inference and learning of a structural activity model, based on weighted least-squares optimization. Similarly to volumetric approaches [53, 165], the video is regarded as a volume in 2D space and time and via a multiscale, spatiotemporal segmentation, homogeneous subvolumes (tubes) are extracted. Descriptors of the tubes form the nodes of weighted directed graphs, the edges of which encode their hierarchical and spatiotemporal relationships. During training, the weighted least-squares graph models of activity classes are learnt and when a new video sequence is observed, its spatiotemporal graph is extracted and matched to the closest one in the training set. Although this methodology offers fast training and efficient feature extraction, it fails to filter out unimportant repeating actions, occurring primarily in the background.

Raptis et al. in [72], propose a methodology, based on discovery of discriminative action parts, which addresses better the above shortcoming and is inspired by the part-based models appearing in [179, 180, 181]. Initially, moving points are detected and clustered based on their spatial and dynamic similarity. The groups represent parts of more complex activities and are described with their intensity, motion and appearance statistics, by employing a regular grid for efficiency. Then, the learning mechanism treats the part descriptors as latent variables and selects the most informative according to discrete optimization [182] of the energy of a Markov Random Fields (MRF). This results into localization of informative parts, additionally to the overall classification process. The main weakness of the proposed methodology is lies on the low-level processing, which is sensitive to video data with extreme intensities or small lengths of sequences.

3.3.2 Knowledge-Driven Approaches

This is the second broad category, consisting of knowledge-driven approaches that are closer to human reasoning and interpretation, permeating by logical rules and ontological hierarchies. Their number is significantly smaller than the number of those that are data-driven and they only make use of global features.

a) Using Global Features

Zhang et al. in [45] employ Stochastic Context-Free Grammar (SCFG) extended by Allens temporal logic [189] to model the complex events that include the interactions among multiple moving objects and/or the interactions between moving objects and static in scenes. These segments are considered as basic motion patterns of agents and are translated as terminal symbols in the

grammar and are detected by a HMM, trained for each one of them and a Minimum Description Length (MDL) based rule induction algorithm is performed to select the best ones, according to their bit-length representation. A manual agglomerative hierarchical clustering is then used to cluster similar rules into meaningful classes. After the rules have been learnt they are parsed using a Multi- thread Parsing (MTP) algorithm. Parallel sub-events are handled by taking into account the temporal relations between ID sets of sub-events that indicate overlapping. Once the parsing tree has been created, a Viterbi-like backtracking determines the maximum possible errors that may have occurred, usually insertion and deletion errors. The redundant states are pruned by employing two more constraints, the beam-width and the maximum errors constraint, so as to reduce the cost of the computationally expensive MTP algorithm.

The goal of Wu and Aghajan in [18] is designing a system that primarily aims in assisting living of elderly or people at risk in general. In their work, normality of an action is based on the frequency it occurs. The methodology introduces an unsupervised approach that employs two hierarchical structures. The first one deals with the clustering of the feature vectors that describe the appearance of the monitored human (blob) using the k-means approach, in a top-down hierarchy. The second hierarchy contains formations according to a bottom-up approach, where as the levels develop more complex actions are considered. Blobs are extracted using background subtraction and are rescaled and distance transformed. The temporal relations of the symbol sequences, representing the postures, define the actions and classification of normal or abnormal events according to frequency of occurrence, meaning that the larger the length of the symbol sequences, the more usual the action.

Liu at al. in [31] propose a simple and computationally efficient methodology for fall detection using a cheap web camera. Foreground blob extraction is initially carried out with frame differencing, mean filtering and thresholding. Height and width aspect ratios define the three types of postures, namely the standing, temporary and lying-down postures. The silhouettes are then refined with vertical projection of their histograms and thresholded by the mean and standard deviation and the features are finally clustered with the k-nearest neighbor approach. Events of falling and intentionally lying-down are discriminated via a simple finite transition model, which considers the time the silhouette is in the temporary posture.

Perse et al. in [50], aiming in the automatic evaluation of complex, multi-agent activities, describe a methodology based on Petri Nets (PN). As opposed to existing systems that employ PNs for the action modeling, like the ones proposed by Lavee et al. in [192] and Ghanem et al. in [193], this approach allows automatic construction of the PNs from the activity templates and the expert knowledge encoded in them. This work studies the recognition of the individual and collaborative activities that occur in basketball games and possess an underlying structure. Naturally, large

activities consist of simpler action primitives. Thus, the PN construction is composed of two stages. First, the action chains that are the basic building blocks and represent individual actions are modeled with three-node chains. The second step then deals with the connection of the action chains so that they encode the complex temporal relations between the actions. Then, the temporal parameters of the model, namely the durations of the actions, are learnt from a small amount of training data. An important part of the proposed methodology is the evaluation of how well actions were performed. Trajectory-based action detectors were applied to each transition that represented an action and the information about the activity score is propagated via the tokens, following the approach of colored PNs. The overall system demonstrated good results on tests done using real-world basketball games video data and more importantly it was shown to be robust in case when the ending of an activity varied.

Even now, recognition of high level, complex activities remains a very hard task. One of the reasons is that they incorporate concurrent temporal relations, which are difficult to be modeled effectively. Another difficulty that permeates all related works, is coping with noise introduced by low-level computer vision processes.

Ryoo and Aggarwal [65] aim to overcome both these obstacles by proposing a probabilistic extension of their recognition framework [187, 194, 195, 183, 196]. The low-level layer of their new system comprises of body part extraction, posture estimation (using Bayesian networks) and gesture recognition (using Hidden Markov Models). The high-level layer, the system distinguishes between atomic actions, composite actions and interactions (up to two persons), using the information from the low-level layer. Then, by exploiting the knowledge of an expert, which describes the underlying temporal, spatial and logical relations of actions encoded in a CFG, they manage to recognize complex actions in a hierarchical manner. Time is represented with respect to Allens temporal logic [189] and ambiguities are resolved using the concept of hallucinations of Minnen et al. in [184].

3.3.3 Depth Sensors

Recently, methodologies based on depth information and specifically Microsofts Kinect sensor, have caught the attention of the research community. Although it was initially designed for controller-free gaming applications for the Microsoft Xbox 360 console, many developers and researchers have used it for various applications, the most prevalent of which are gesture and human activity recognition. The main advantage of Kinect over regular visual sensors is the depth information provided by its integrated infrared sensor, which can be used for 3D scene reconstructions where illumination is unstable, while it can capture useful information in near no-light conditions. In combination with its low cost and commercial availability, Microsoft Kinect is certainly a sensor that will drive

future research. However, since we focus only in methodologies based on visual sensors without the augmentation of other modalities, we review only a few representative methodologies that use depth information for the sake of completeness. We choose also not to include them in the evaluation section because their additional sensing capabilities give them a significant advantage over the rest of the methodologies reviewed here.

Gill et al. in [209] explore the new directions that open with the use of Kinect and provide a comparison with their previous work [210] where they achieve depth perception using stereo camera pairs. Their main algorithm begins with calibration of multiple Kinect camera pairs that collaboratively estimate the global world space. Next, an adaptive, voxel-based modeling of the background follows and human detection based on weak skin and head classifiers, using a combination of color and depth information. Special care is taken for the disambiguation of human and non-human objects and occlusions that occur during everyday activities.

Li et al. in [211] proposed a methodology that uses sequences of only depth images as input. Action modeling is based on trained action graphs [212], which are suitable for silhouette-based action recognition and propose using bags of 3D points to characterize a set of salient postures, which can cope with cases of occlusion. Xia et al. in [213] outperform the aforementioned methodology, by employing a compact posture representation based on histograms of the 3D joint locations estimated from Kinect depth maps using the method in [214]. These histograms are reprojected using LDA and then clustered into k posture visual words, the temporal evolutions of which are modeled by discrete hidden Markov models (HMMs). Emphasis is given in view invariance, achieved using a spherical coordinate system and the outputs are produced in real-time.

Sung et al. in [215] use the Kinect sensor for robotic vision. Skeleton based features, namely the joint positions acquired using PrimeSenses tracking algorithm [216], are used to describe the body postures and also the motion of body and hands. Strong emphasis is given in the learning model. Specifically, the authors adopt maximum-entropy Markov model (MEMM) of Mccallum et al. [217] and extend it to a two-layer model to better capture the hierarchical nature of human activities, which are inferred using a dynamic programming approach. Although occlusions are not treated in this methodology, the model is not falsely triggered by activities that have not been observed before by the robot.

3.4 First Level Evaluation

In this section, we attempt to evaluate the most representative and prevailing systems from the ones discussed in the previous sections. All of them have been published in well-accredited conferences,

journals and transaction papers, have been widely accepted by the research community and promote state-off-the-art methods. One the least developed sections in surveys and reviews on research about behavior analysis and human activity recognition, is that of their evaluation. Evaluation can be a daunting task, because there has not been enough effort yet for the establishment of common ground that would allow the development of objective metrics for methodology assessment. Even if results in forms of confusion matrices and percentages of accuracy are informative, they are not sufficient for the evaluation and comparison of methodologies. Especially recently there exists the phenomenon of researchers reporting very high accuracies and also close to each other. If this information alone were enough for evaluation, it would suggest that the problem is almost solved and not significant progress is being made. However, it is known that the problem in general is far from being solved and the research that is being conducted is of great scientific meaning. In order for a more meaningful evaluation to be able to be formed, the layout of the research papers should evolve and the contents should be more thorough. For instance, researchers should not only suffice in explaining the parameter tuning, but also include the human aspect, especially since the type of methodologies studied here is human-centered. Moreover, standards should be set about quantification of the complexity of activities difficulty of environmental parameters and the requirements for computations and training. Finally, failures and experiences should be described not only without fear but with rather confidence, since not only they allow better overview of the described work, but also help others to avoid dead-ends and promote the general benefit of the research community. The problem of evaluation is one of the issues addressed in the position papers [124 - 132], appeared in the workshop Pervasive 2010, in Helsinki, Finland, which was specifically dedicated to communication of ideas upon improving the way research is carried out in the field of human activity recognition.

The evaluation presented in this section should be considered as a first level evaluation, since we lack the tools and information for an in depth, objective evaluation. Our goal is to acquire an evaluation outcome that gives a general idea about a methodologys overall performance, weak and strong points. Thus, we selected the most important features that permeate all the methodologies and quantified based on knowledge about the categories the individual components used fall into and information provided by the authors. For example, the classification in Table 3.1 gives a general indication, based on theory, about how the categories of features and recognition methods fare with respect to certain aspects (features), like simplicity, complexity, etc. Admittedly, this information is very generic, it is however the only common ground that can be used for evaluation. Information provided by the authors is used of course in conjunction for fine-tuning. For example, in case of a discriminative method being used for recognition, it would be expected that the methodology require

a big amount of labeled training data and thus get a low score for the corresponding feature, but if it is mentioned that effort has been made towards the alleviation of this problem, the penalty will be reduced. We chose direct quantification instead of fuzzy labels for the evaluation, because they give a more accurate perception and allow feature values to be combined into an overall score, for a quicker and more compact evaluation.

Ten features have been chosen for the evaluation of the various selected methodologies, after discussions with former and current members of the ATR Center and its collaborators. These features have been shaped in time during different surveys conducted in our lab and the majority of them can describe general and important aspects of methodologies (i.e. simplicity, computational complexity). More specific features according to the application studied here are also added (i.e. complexity of behavioral patterns recognized) in an effort to make the representation of the main aspects of the selected methodologies more appropriate. In Table 3.2, a list of these features is provided along with a brief description for each one. In addition, we wanted to provide an overall evaluation of the systems based on the perspective of the parties who are involved in the development and the use of the systems, namely the developer and the end-users, after discussions with members of our laboratory, students of Wright State University and external collaborators. For example, a methodology of great simplicity might catch the attention of a developer, since it implies easy implementation, whereas it is not of much interest to the end user, who just cares about its functionality. Each feature has been weighted according to the importance it posed to each group and those weights have been averaged out to show the final weights for the evaluation. The chosen weights are purposefully generic and get values from 1 to 3, corresponding semantically to the categories of low, medium and high importance. These weights are shown in Table 3.3. However, we realize that opinions will probably vary concerning the feature weighting, so have also included the unweighted results of the evaluation, along with the evaluation with respect to the developers and the end-users perspective, as well as their average result. The evaluation scores are shown in Figure 3.3.

It could be argued that these features are not directly quantifiable, however by assigning numbers to them we can convey the advantages and disadvantages of each methodology in a more compact and easy way, while it allows us to assign scores to the overall evaluation. These evaluation scores are not to be treated strictly, they mainly give a general idea about a methodologys performance. The overall score is a normalized weighted summation of the scores of the individual features, which have a range from 1 to 9, except for the case of the last four features, which form pairs and their range is from 1 to 3. We choose these ranges because after long discussions and observation of the responses of the subjects that took part in the evaluation, we concluded that they provide a generic yet discriminative way to differentiate the methodologies. The aforementioned feature pairs are

features that provide more meaningful information when combined together. Specifically, Dataset (F7) is multiplied by Robustness (F8), resulting in a feature that collectively shows the robustness with respect to the difficulties imposed by the dataset. Similarly, Behaviors (F9) is multiplied by Accuracy (F10), resulting in a feature that combines the achieved performance with the complexity of the behaviors recognized, since it is not fair for a methodology to get a high accuracy score when it only considers simple cases and vice versa. In order for them to remain in the same value range with the rest of the features, the weights in each pair are also averaged. Normalization of the scores is done with respect to the ideal score where all features achieve their maximum score. However, this ideal score can be deemed as unrealistic, since many features contradict each other. Its main purpose is for the scores to have stable values and graphical representations and give an idea of a global optimum. The final evaluation formula thus is:

$$Score = \sum_{i=1}^{6}(W_{ji} * F_i) + \frac{(W_{j7} + W_{j8})}{2} * F_7 * F_8 + \frac{(W_{j9} + W_{j10})}{2} * F_9 * F_{10}$$

where F_i is the value of feature i and W_{ji} is the weight for group j (1 for end-users, 2 for developers and 3 for their average) and feature i.

Table 3.2: Evaluation features.

Simplicity (F1) *[1-9]*	The methodology is easy to implement and contains a few single standalone steps.
Complexity (F2) *[1-9]*	Reflects the number of computations/operations required.
Impact (F3) *[1-9]*	The methodology promotes original ideas and affected other research works.
Interpretability (F4) *[1-9]*	Results and operation of the methodology can be easily understood.
Training (F5) *[1-9]*	Number of (labeled) data needed for training.
Scalability (F6) *[1-9]*	Methodology's adaptability to new environments, activities etc.
Dataset (F7) *[1-3*	Difficulty of the dataset used as input.
Robustness (F8) *[1-3]*	The methodology's capability to produce acceptable results under challenging circumstances.
Complexity of behavioral patterns recognized (F9) *[1-3]*	Depicts the methodology's capability of dealing with simple, complex or group behaviors.
Accuracy (F10) *[1-3]*	Reliability of produced results.

Table 3.3: Communication Schemes Resource Allocation (X indicates that resource is used)

Perspective \ Feature	*Simplicity (F1)*	*Complexity (F2)*	*Impact (F3)*	*Interpretability (F4)*
End user (W1)	1	1	1	1
Developer (W2)	3	3	1	3
Average (W3)	2	2	1	2

Perspective \ Feature	*Training (F5)*	*Scalability (F6)*	*Dataset (F7)*	*Accuracy (F8)*
End user (W1)	3	3	1	3
Developer (W2)	2	3	2	3
Average (W3)	1.5	3	2.5	3

Perspective \ Feature	*Behaviors (F9)*	*Robustness (F10)*
End user (W1)	3	3
Developer (W2)	3	3
Average (W3)	3	3

3.5 Discussion

A general phenomenon observed in the recent methodologies is a preference towards discriminative models for activity recognition and the use of local features for action description. The main reasons discriminative models are appealing are that are simple to construct and can lead easier to real-time operability. All this research revolving around methodologies based upon discriminative models has improved significantly their performance. It is not thus a coincidence that all winning systems in Pascal Visual Object Classes Challenge 2009 (VOC2009) were discriminative. Another trend as aforementioned in activity recognition methodologies is the use of local features and descriptors, frequently along with a Bag of Words (BoW) approach for dictionary creation. The main reason for this choice is that local features are inherently robust to noise and they can be used without traditional preprocessing steps, such as background subtraction, which are still open research problems.

However, one of the methodologies that achieve high scores in all categories is the work of Xiang and Gong in [54], which employs more traditional methods (Hidden Markov Models and Motion History Images). Some of the reasons are the focus in robustness and more importantly its ability to cope with complex activities of multiple subjects. The majority of methodologies take only simple

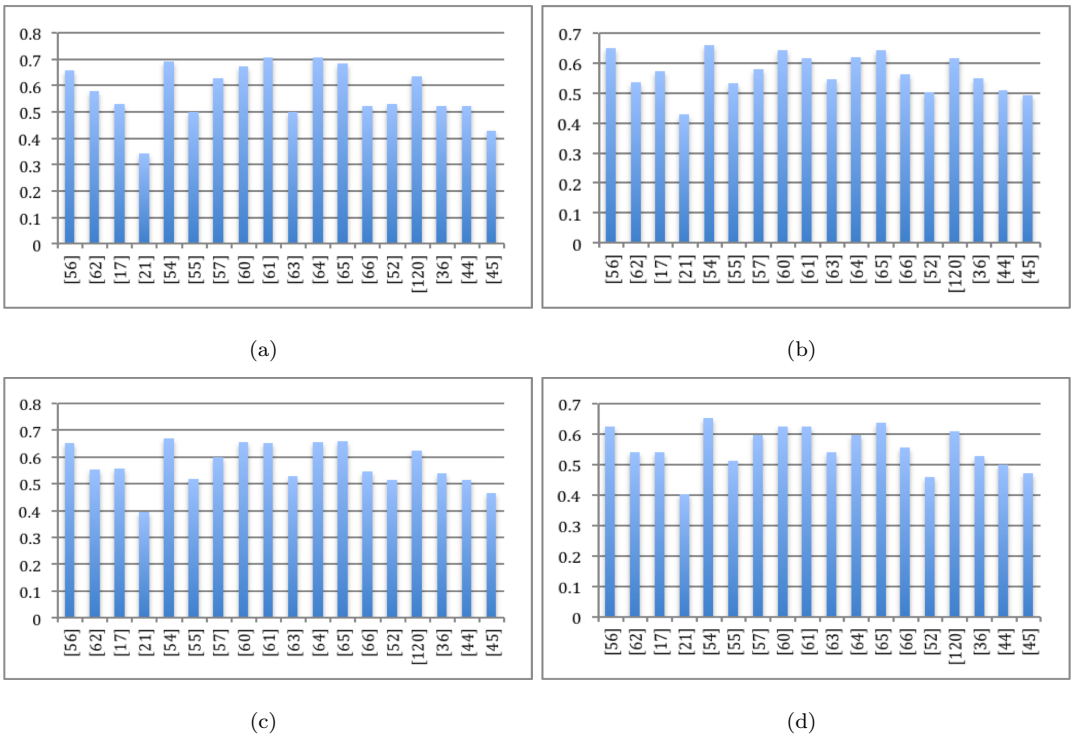


Figure 3.3: Evaluation scores according to a) End-users perspective, b) Developers perspective, c) Average perspective, d) without weights

actions into consideration of usually one subject, which hinders their scalability to more complex, real world situation. Also, robustness is a very important feature that opens up directions for real-world applications, where the many sources of noise and ambiguities would cause many systems to fail. As it can be seen in the feature weights in Table 3.3, the end-user does not care about the implementation and intricate mechanism of the methodology, but is interested instead on user-friendly systems with high performance. For the same reasons, which are crucial from an end-user perspective, Liu et al. in [61] and Rapantzikos et al. in [64] achieve high scores in this category and they both use cuboid features and simple spatiotemporal detectors, following the current trends. From a developers perspective, the methodology of Bregonzio et al. in [56] seems attractive because it also addresses the aforementioned problems, using simple features, fast and popular methods and an algorithm that is easy to implement. Although performance and operability are important from a developers perspective as well as an end-users, developers also take seriously into account implementations issues, reproducibility and in general tend to favor simpler algorithms. In the last categories, which tries to collectively weight the features according to both end-users and developers perspective, we notice that one of the methodologies achieving high-scores is of Hu et al. in [60], which describes

a system tested in real, complex scenes and comprises well-known. Another system achieving high scores is the one proposed by Ryoo and Aggarwal [65], which is attractive because it addresses issues that emanate from the low-level image processing to high-level semantics interpretation in a complete approach, which is a continuation of previous works.

In general, it can be noted that there are some common goals in the methodologies about vision-based human activity recognition. First of all, in order for them to be applicable in real, everyday environments, they must be able to cope with the difficulties of a complex and dynamic environment. This is one of the most known issues and many ways to resolve it have been proposed but even today the claim of a generally applicable methodology is difficult to be made. Another issue found in real environments is occluded body parts, either by ones self or the environment. This problem is usually tackled by either high-level inference methods, compensation of views using multiple cameras or sensors with additional modalities, such as depth sensors. Moreover, it is observed that in few methodologies address the problem of high-level interpretation of complex, real-world activities and proposed solutions are usually not suitable for real-time implementations.

The problem of human activity recognition is open since early 80s and many interesting solutions have been proposed since. The next level would be the creation of a complete system, as depicted in Figure 1, which is not thoroughly addressed in literature. A complete system should include all the mentioned steps, from the initial camera installation and data acquisition up to the final level, which aims in Human and Computer Interaction and often eludes the researchers attention. The design and implementation of a complete system is not an easy task, involving several correlating factors, but is the only way to prove the real applicability of a methodology. Since the methodologies about vision-based human activity recognition deal with huge amounts of information and complicated processing, they are computationally demanding. A complete system has to be carefully designed and implemented, so that its comprising parts can be combined to achieve high performance, without imposing prohibitive computational cost.

4

Image Segmentation Metric

4.1 Introduction

Image segmentation is one of the fundamental low-level processing techniques that partitions a digital image into multiple segments (sets of pixels or superpixels), in order to create a more meaningful and easier-to-analyze representation In the image processing and computer vision fields, there are many image segmentation methodologies [244–248]. In each of these algorithms, different criteria and computational steps are followed in order to produce a segmented image. In addition, almost all of them rely on user-defined parameters that result in variations of segmented images for the same original image.

While measuring the computational complexity, resource consumption and even user friendliness is straightforward, deciding whether a result is good or not depends entirely on human judgment and the specific application, as stated in [249] and [250]. A common practice for the evaluation of the effectiveness of an image segmentation comes either through direct visual inspection, or indirectly, through the final results of a more complex process that relies on image segmentation, as shown in [251] and [252].

However, visual assessment is a tedious and restrictive process and an automatic, mathematically sound and generally accepted way for evaluation is expected to be of great benefit to the research community. In a recent and thorough survey of evaluation methods, Zhang et al. [253] have categorized this type of evaluation methods to analytical and empirical. The difference is that analytical methods assess the segmentation algorithms themselves, whereas empirical methods assess their results. The main advantage of the first is that they can directly contribute to the improvement of the studied algorithms, but they are not easy to generalize their findings ([254]). Thus, the empirical methods have garnered more attention, as they produce generic results, irrespective to the segmentation algorithm.

Empirical methods are further divided into two categories, namely goodness methods and discrepancy methods. Goodness methods, as the name suggests, attempt to measure how "good" a segmentation result is based on metrics deriving from human intuitive rules, almost always adhering to the criteria proposed in [247]. There is a series of works that evolve this idea, the most notable of which are the works of Liu and Yang [255], Borsotti et al. [256] and Zhang et al. [253]. Discrepancy methods measure how similar (or different) a segmented image is compared to an ideally segmented image from human supervisors. One of the most prominent works that initiated the formulation of methodologies employing this idea is the work of Martin et al. [257] that still fuels research works, such as those of Dogra et al. [258] and Goldmann et al. [259].

Both of these types of methods have evolved significantly, especially during the last decade, and can encompass to a great extent human perception in mathematical formulas, in order to quantify the effectiveness of an image segmentation. Their power and weakness, however, lie in the same fact: the evaluation is subjective. This is obvious in the case of the discrepancy methods, if there were cases where the ground truth images differ significantly. Goodness methods claim to overcome the shortcomings ground truth dataset imposes, but are still dictated by human intuition, which varies significantly among individuals.

In this chapter, we propose a graph-based, blind reference evaluation scheme (no need for the original or a human- segmented image) for image segmentation results. To this end, we employ an extended version of graphs, the regional localglobal (RLG) model, which is a weighted undirected graph with attributed edges and vertices. This type of graph is generally known as Local-Global (LG) graph [260] or Attributed Relational Graph (ARG) [261] and can be the cornerstone of very powerful models and adapt to many different scientific fields and applications. The proposed evaluation scheme differs significantly from the existing evaluation schemes, as it does not aim at measuring the goodness or the discrepancy of image segmentation, but rather at describing its underlying structure, by discovering the amount of detail depicted in a segmented image and its spatial distribution. This approach is based on the fact that image segmentation is a strictly perceptual and application-driven process and the existence of numerous segmentation algorithms that produce different results proves that the truth exists only in the eye of the beholder. This point of view is elegantly pointed out in [262], which presents the fuzzy region growing (FRG) segmentation algorithm, an interesting method for segmentation that uses fuzzy logic to define the region segments and merges them into larger areas. We believe that structure is a crucial factor for many applications and should be taken into account when evaluating the results of image segmentation algorithms. Finally, it is a blind reference method, having as input only the labels of the regions in a segmented image.

Our proposed evaluation scheme was applied on images produced by three publicly available

image segmentation algorithms with a MATLAB interface. The first algorithm, statistical region merging (SRM) [263] is a computationally efficient algorithm that employs a data structure of disjoint sets with a union-find strategy to perform image segmentation by region merging following a particular order in the choice of regions. The second algorithm was the entropy rate superpixel segmentation (ERSS) [264], which is based on maximization of an objective function with two components, the entropy rate of a random walk on a graph and a balancing term. The first component favors the formation of compact and homogeneous clusters, while the latter favors the formation of clusters of similar sizes. The third algorithm is the widely known edge detection and image segmentation (EDISON) system [265–267], developed at the Robust Image Understanding Laboratory, Rutgers University. EDISON is a low-level feature extraction tool that integrates confidence-based edge detection and mean shift-based image segmentation.

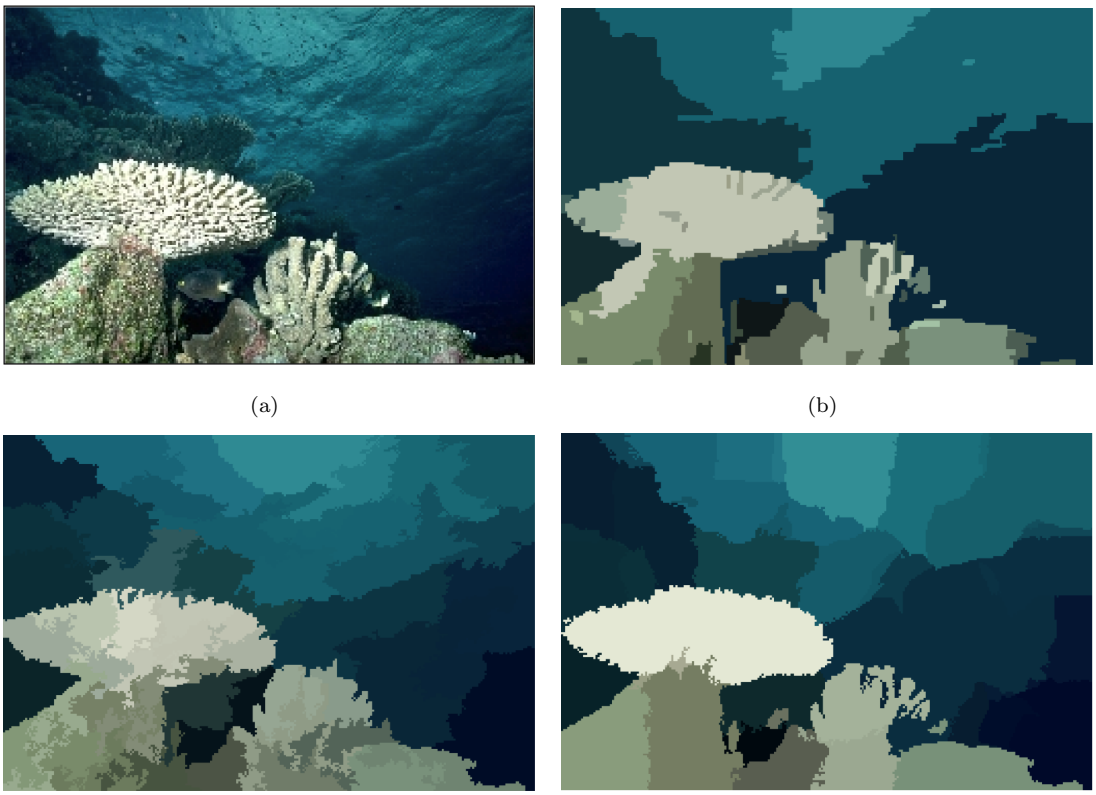
4.2 Notations and Definitions

Image segmentation can be defined as the partitioning of an image area A of an image I into regions R_i, ($i = 1 \dots n$), such that:
1)The union of the regions forms the image: $R_1 \cup R_2 \cup \cdots \cup R_n = A$.
2)The regions are disjoint: $R_i \cap R_j = \emptyset, \forall i \neq j$.
3)All the pixels in a region possess the same or similar properties or features. Respectively, pixels that belong in adjacent regions attribute properties or features to the regions that differentiate them.
Definition 1: A color image segmentation method is characterized as "objective" if and only if the regions are formed by adjacent pixels of the exact same color. Essentially, this is the initial image.

4.3 Graph Based Representation Of Segmented Images

Each segmentation algorithm employs different methods and internal metrics in order to produce a result, selected in a way that makes sense according to the developer. Thus, the formulation of another evaluation metric to measure the "goodness" of a segmentation is expected to favor some types of segmentation unfairly, according to its compliance with the image segmentation algorithms internal mechanism. However, it does not mean that the results that are not considered good according to the new metric are really so. The only true judge is the algorithms developer, who does not necessarily have to treat segmentation as a representation problem, forcing the results to be as close as possible to the original image, or as a semantic problem, where the results should possess a semantic meaning. Image segmentation then is only useful to an end user if their expectations agree with the developers. One such example is shown in Figure 4.1. The results produced using

the SRM and EDISON techniques produce regions with non-uniform sizes (high entropy), whereas the entropy rate (ERSS) based image segmentation technique produces regions with uniform sizes (low entropy).



(a) (b)

(c) (d)

Figure 4.1: Different image segmentations. a) Original image, b) SRM, c) ERSS, d) EDISON

Our graph-based evaluation aims to discover the underlying structure of a segmented image and to provide a quantitative and descriptive evaluation of the details distribution. The final result does not intend to assess the the segmentations quality, but only gives an objective idea of the user, helpful enough to act as a guideline for a subjective evaluation. The proposed evaluation is made in a blind reference manner, so as to keep the result as objective and unbiased as possible. It should be noted, however, that the basic idea behind the proposed evaluation is generic and could be applied to existing image segmentation metrics as well, augmenting them with one crucial aspect they are missing, which is the structural information expressed with graphs. Structural information does not only provide human understandable interpretation, but also the graphs that express it can render the management of the metrics components easier and more efficient.

Detail in our case is defined with respect to a region (or segment) size. Our empirical studies show

that in complex natural images in the case of the "objective" segmentation (as defined in definition 1), which essentially is the original image itself, the vast majority of segments are of 1 pixel size. Thus, we assume that the finest level of detail is represented with a region of size 1 pixel and the bigger the region gets, the less detail information it encompasses. We further assume that each of the segmentation algorithms has already been evaluated by its developers and the results they produce are meaningful according to their expectations. Thus, the remaining aspect that can be evaluated is the distribution of the details in the result. One could conduct this evaluation by measuring the uniformity of the region sizes via their entropy, which is a natural and robust measure. However, the region locations and their distribution cannot be expressed in this way. This information could be very useful and contribute significantly in the image segmentation evaluation.

LG graphs have been proven to be very successful tools for image understanding and object recognition. In our case, a segmented image is represented by its RLG graph, which aims at measuring and describing the distribution of detail by discovering areas of high detail. Each region is represented by one of its most salient points, i.e. its centroid. The skeleton of a graph is based on the Delaunay triangulation, as shown i Figure . Each node holds information about the regions location and size SR_i. The location of the region R_i is indicated by its centroid (X_i, Y_i), where $i = 1, \ldots, N$ and N being the number of regions. The edges of the RLG graph are attributed too and in our case hold information about the distance of the centroids (nodes) they connect. Different detail levels found at the graph's nodes indicate areas of different detail in the image. Figures 4.2(b) - 4.2(d) show the graphs produced in our example image, where the size of the circles at the nodes and the color at the edges indicate their weights.

4.4 Evaluation Scheme

In this section, we present an evaluation scheme that highlights the contribution of the graph-based structure to the evaluation of image segmentation. This scheme is inspired by works emanating from the image analysis domain and aims at indicating areas of interest in the segmented image, in a human-perception-driven manner. Figure 4.2(a) graphically shows what we expect to find from a segmented image. In particular, we would like to see the number of clusters formed by small regions and their distribution in the image area. Thus, we have employed the LG graph model to describe the relations and attributes of the image regions after segmentation, in effort to define a structure that will represent the association of the small regions and their distribution in the entire area. It is known that the clusters of small regions may represent areas of high entropy. At this point, a question that immediately arises is: What is considered close enough in detail and proximity?

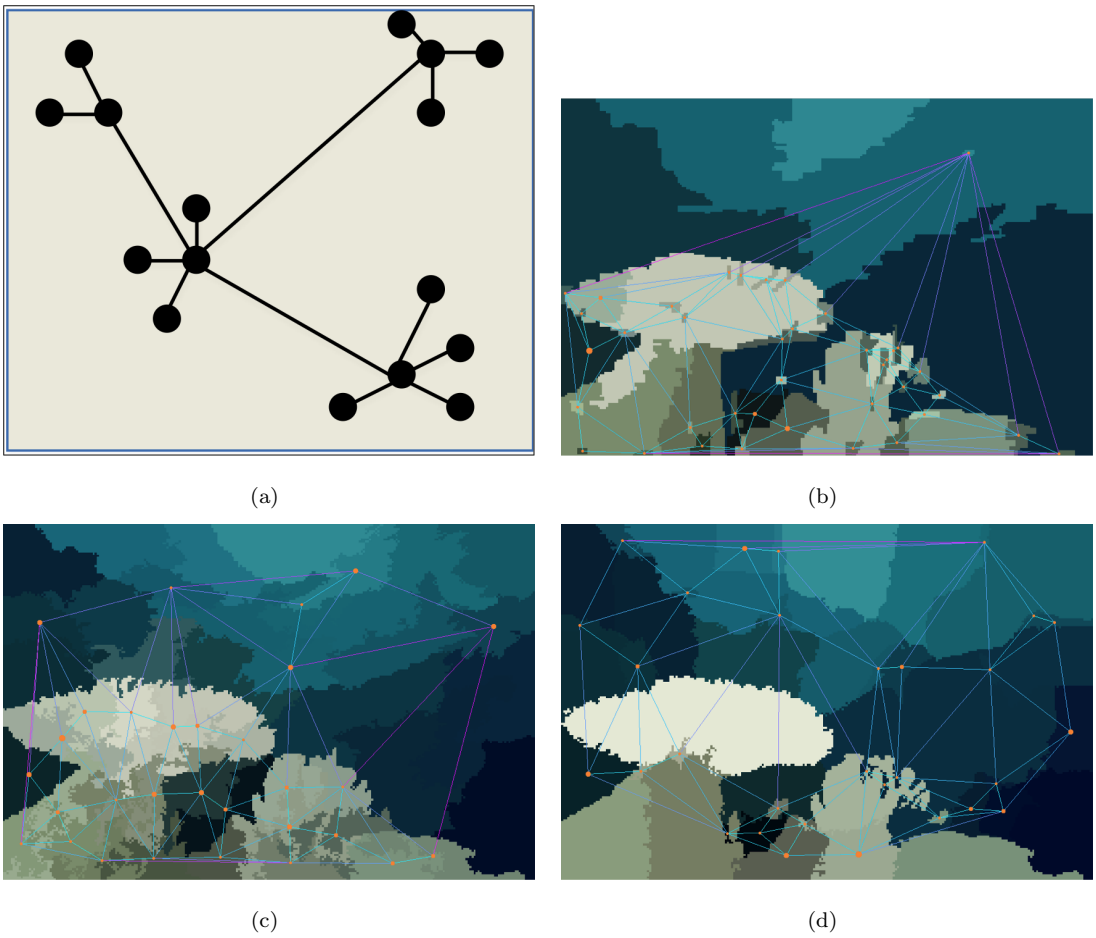


Figure 4.2: RLG graphs: a) A general graphical representation of clusters of small regions, an example for b) SRM, c) ERSS, d) EDISON

One of the key assumptions in this work is about the meaning of the medium region (MR). The notions of big/small, close/far, etc are ambiguous, strictly depend on the task at hand and cannot be generalized. Here, we make the intuitive convention that a medium region size (MRS) in an image with N regions is the one occurring in the case of uniform region size distribution, thus having an area of $\frac{S_I}{N}$. We then aim in defining "closeness" in the spatial domain too. To do so, we now define the Medium Region Distance (MRD). If we consider for simplicity the image being square and the MRS being square as well,being square as well, then the MRD is defined as the distance of the centroids of two adjacent MRs. Since there is no accurate or even general definition of the aforementioned notions, these choices will not affect the evaluation significantly and it can be argued that they can approximate human perception.

After having defined the notions of MRS and MRD, we cluster the regions of high detail as

follows. First, we locate the small regions that hold the highest amount of detail. Since the term small is abstract, we consider a region to be small if its size is less than the MRS, only the centroids of regions that comply with this constraint are used in the graph. The next step is the clustering of these regions. Areas of clusters indicate areas in the image where parts of objects or texture with interesting features might reside. In many applications, especially those related to image understanding, these areas might provide helpful cues for pattern recognition and object detection through synthesis. For example, if the segmentation has been performed on an image depicting an aerial view of a port, clusters of small regions might indicate the buildings at the port and the large segments the sea. For the clustering we construct the dendrogram of clusters, using agglomerative hierarchical clustering and acquire the final clustering by imposing the MRD as a cutoff distance. The result of this process is the number of small regions or high detail regions N_{SR}, which gives an idea about the amount of detail in the segmented image and the number of clusters N_C, which gives an idea about how it is distributed throughout the image. Figure 4.3 shows the clusters of small regions in our general example.

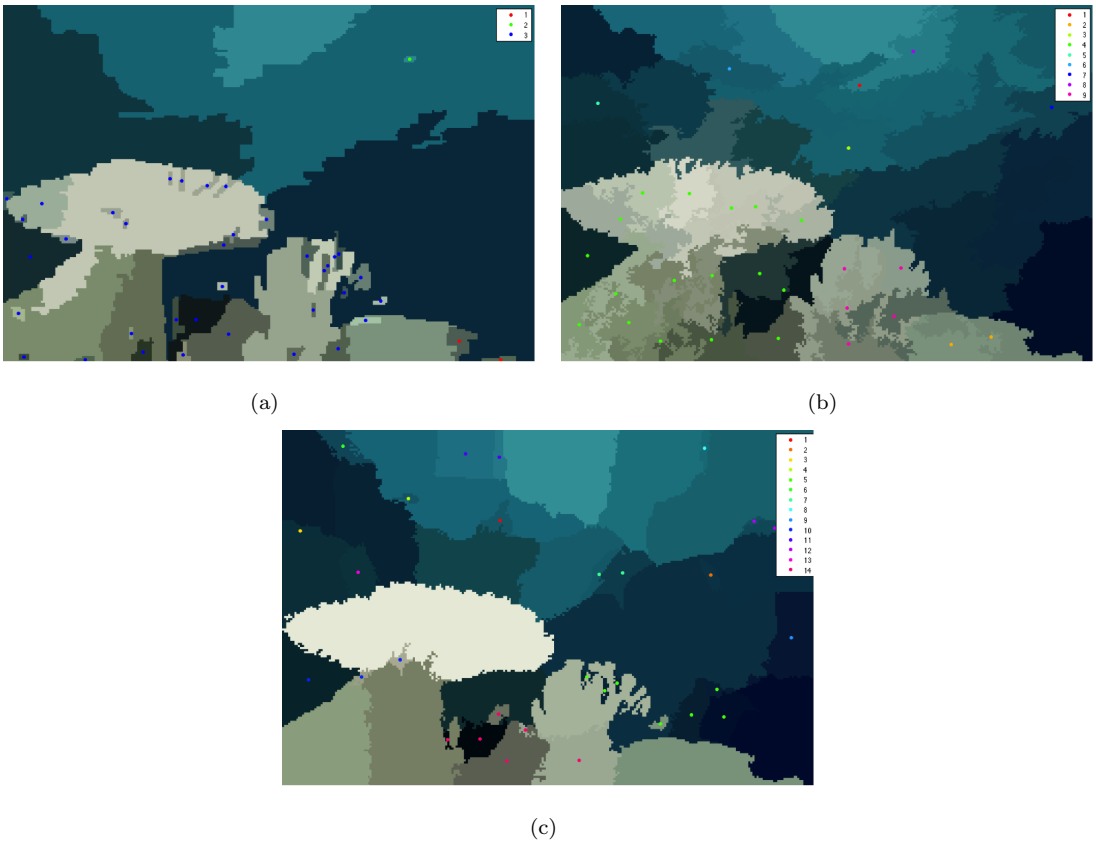
The resulting clustering allows us to measure the amount of detail preserved in the segmentation, along with its spatial distribution. In our case clusters of small regions indicate general areas of potential interest, so we design the measure to increase along with the number of small regions detected and decrease as the number of clusters decreases, since isolated small regions are often a byproduct of the segmentation process and are hard to combine with nearby regions in a region synthesis process. Every small region contributes, however, to increasing the measure, according to its size, which is inversely related to the detail that the region preserves. The measurement is conducted in the cluster level. Each cluster C_i has a weight, according to the number of regions it contains, defined as W_{C_i}:

$$W_{C_i} = \frac{|C_i|}{N_{SR}}$$

The level of detail D_{C_i} of each cluster C_i is related to the mean value of the sizes of the regions that comprise it, symbolized as $S_{R_{C_i}}$, because clusters of small regions are considered to hold more detail. The region sizes are normalized by the MRS. so as to have values in the range (0,1). Finally, since detail is inversely related to the regions size, the outcome is subtracted from 1 so as to acquire the cluster's level of detail:

$$D_{C_i} = 1 - \sum_{i=1}^{|C_i|} \frac{S_{R_{C_i}}}{|C_i| * MRS}$$

We can now define the measure of the Segmentation's Detail Density, SDD as:



(a) (b)

(c)

Figure 4.3: Clustering of small regions. The legend shows the color that corresponds to each cluster. a) SRM, b) ERSS, c) EDISON

$$SDD = \frac{\sum_{i=1}^{N_C} W_{C_i} * D_{C_i}}{N_C} \tag{4.1}$$

where the number of clusters N_C shows how dispersed the detail is and, as it increases, the value of SDD decreases.

The values of SDD are in the range (0,1) and values close to zero signify the existence of disperse regions of medium size, or in other words uniformity in the region sizes and values closer to 1 signify the existence of dense clusters of small regions. The example of Figure 4.3 is a good indication of how the measure SDD performs. In our example image, the number of segments produced by all three segmentation algorithms is 48. In the case of SRM, the SDD measure gets a very high value, because there are many small regions and most of them belong in one cluster, located upon the region where the coral is, which has interesting textural features. The region sizes produced in the case of ERSS are close to uniform, being close to the medium region size MRD and homogeneously placed throughout the image, thus SDD gets a low value in this case. Finally, in the case of EDISON's

segmentation, there are many small regions that increase SDD value; however, they are dispersed and thus the final SDD value is less than that in the case of SRM. It is interesting to notice that in this case the small regions correspond to regions not possessing as much semantic information, such as borders between regions.

At this point, we should mention that another way to express the above measures with use of entropy. Entropy of the region sizes has previously been used by Zhang et al. [253] as an indicator of how uniform the region sizes are. Similarly, one could use the entropy of the distances among the region centroids, as indicated by the RLG graph and acquire additional information about the uniformity of the regions' spatial distribution, in a compact manner. However, we believe that the detection and localization of the small regions and their spatial distribution, as described above, is more descriptive and can prove more useful for evaluation regarding the applicability of a segmentation algorithm in specific tasks.

4.5 Experimental Results

In order to be able to make fairer and clearer comparisons among the results produced by the three image segmentation algorithms, the number of segments for each image was selected to be the same for all algorithms. Of course, each of the algorithms could be fine tuned so as to produce results that better agree with its purpose, but here there was no point in doing so, since the algorithms are not under assessment here. We are only interested in observing how our evaluation scheme can be applied on different results and open new paths for research. The plot in Figure 4.4 shows the results for images produced by the three algorithms and Figures 4.5 and 4.6 show the image segmentation results for one example-image. Images 1-10 depict mostly wild natural scenes, which in general have more chaotic information content, images 11-20 contain buildings, which in general have stronger edges and structural content and images 21-30 contain faces and bodies of humans, since humans are often a subject of study. The images were taken from Berkeleys database for image segmentation evaluation [257].

Clearly, the SDD values indicate that regardless of the input image, ERSS algorithm tends to produce regions of very uniform sizes and thus the resulting values of SDD are distinctly below those in the cases of the two other algorithms. SRM in general produces segmentations with high SDD values and so does EDISON, but on average its SDD values are below those of SRM. One interpretation is that, in most cases, EDISON does not seem to be affected by textures, which are usually unified in large regions. In many applications, this is favorable and it has been argued in the literature that EDISON is able to produce results close to a humans general expectations. However,

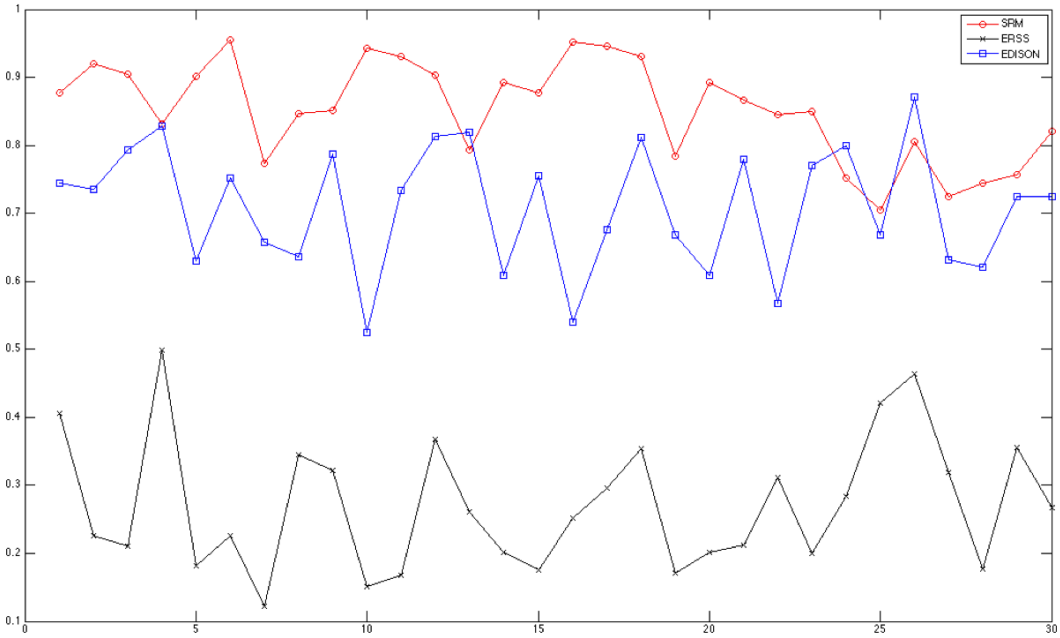
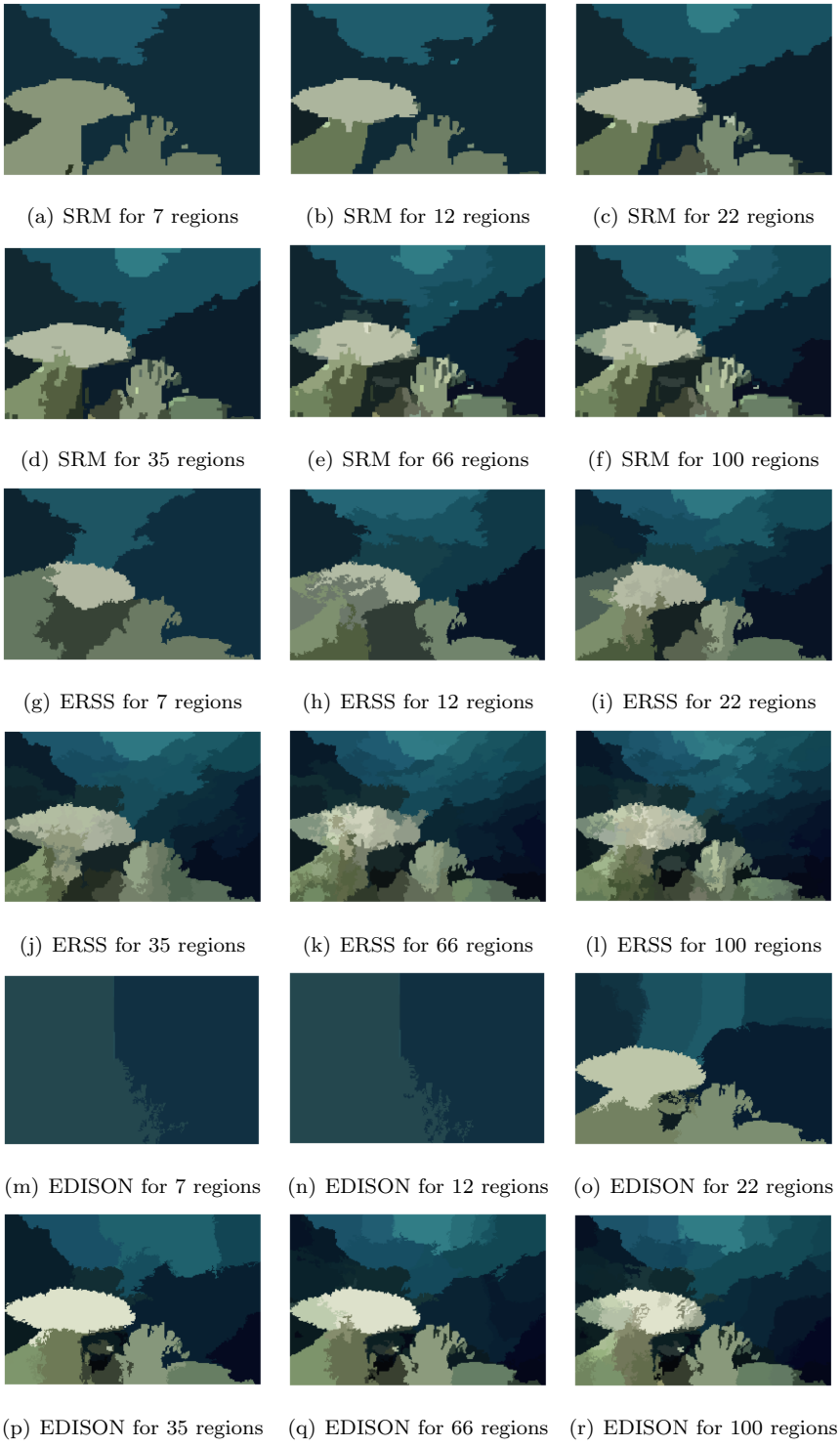


Figure 4.4: Measurements of Segmentation Detail Density, SDD for SRM (red), ERSS (blue) and EDISON (green). The figure shows results for the 30 images (x axis).

since in many cases there are spatial correlations among the images regions, an inhomogeneous distribution indicates that often the small regions produced by EDISON are a byproduct of its process, since they do not represent parts of objects or texture. Conversely, in the case of SRM, more local aspects are concerned and small regions are often parts of texture that differ adequately from their neighboring regions so as to be identified as independent segments. This feature can be very beneficial for region synthesis and detection of objects, such as faces and specifically facial features, in complex environments.

Next, the SDD values are calculated for the three algorithms in the case of one image, segmented into different numbers of regions. Most of the conclusions derived above still apply in this case. The algorithms, as expected, seem to follow the same pattern for different segmentation granularities. Deviation, however, is observed in the case of EDISON, which results in very high values of SDD in the first two cases of small number of segmentations. A supervised or unsupervised evaluation would show that the performance of EDISON in these cases is very poor. In our case, goodness cannot be assumed or estimated. SDD values can be derived, but it depends on the application and additional knowledge for more thorough conclusions and interpretations of its meaning. It can be generally argued though that these values provide a better insight in cases of adequate number of segments, where small regions are more likely to be created.



(a) SRM for 7 regions (b) SRM for 12 regions (c) SRM for 22 regions

(d) SRM for 35 regions (e) SRM for 66 regions (f) SRM for 100 regions

(g) ERSS for 7 regions (h) ERSS for 12 regions (i) ERSS for 22 regions

(j) ERSS for 35 regions (k) ERSS for 66 regions (l) ERSS for 100 regions

(m) EDISON for 7 regions (n) EDISON for 12 regions (o) EDISON for 22 regions

(p) EDISON for 35 regions (q) EDISON for 66 regions (r) EDISON for 100 regions

Figure 4.5: Segmentations for one image, segmented in different granularity levels (x axis), from coarse to fine (small number of segments to big number of segments).

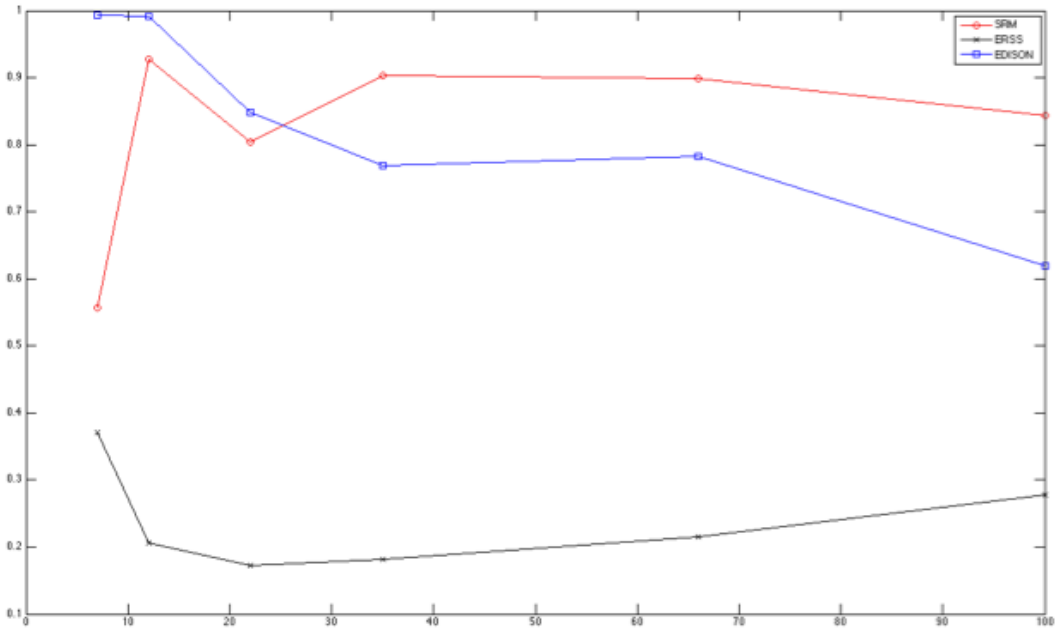


Figure 4.6: Measurements of SDD of each corresponding image in Figure 4.5 for SRM (red), ERSS (blue) and EDISON (green).

5

A Methodology For Detecting Faces From Different Views

5.1 Introduction

Face detection is one of the fundamental techniques that enables natural human-computer interaction (HCI) and is the cornerstone of more advanced processes, including face alignment, face modeling and recognition, head pose tracking, facial expression recognition, to name a few. The problem of face detection has been open for many decades and there are several surveys that provide a thorough overview of the problem [268–271].

The most common categorization of the face detection techniques is that introduced by Yang et al. [271] who grouped them into knowledge-based methods [272], feature invariant approaches [273–278], template matching methods [279, 280], and appearance-based methods [281–289]. In knowledge-based methods, face detection is based on pre-defined rules deriving from human knowledge. In feature invariant approaches the goal is to extract features that adequately describe the facial structure and are also robust to pose and lighting variations. In template matching methods the face is detected after successful matching with a template stored usually in a database. In appearance-based methods machine-learning techniques are usually employed and face models are trained over a number of images.

The robustness of these approaches is challenged by many factors such as changes in illumination across the scene, shadows, cluttered backgrounds, image scale, facial pose, orientation and facial expressions. To achieve a good performance, many of these methods make strong assumptions, like assuming that the face is either segmented or surrounded by a simple background and the images are well-illuminated with frontal-facial pose.

In this chapter, we present a Local-Global (LG) Graph approach for detecting faces in both frontal and profile views. The overall approach to face and facial expression detection presented here - the LG graph approach combined with the skin detection procedure - is robust against cluttered backgrounds, uncontrolled illuminations, shadows, and to a variety of facial poses and orientations. It should be noted that the methodology incorporates a synergy of modules, which can be replaced by other ones of similar functionality and potentially improve the methodology's performance, thus allowing it to evolve along with the advances in the fields of image processing and artificial intelligence. Our main effort is to combine straightforward concepts in an efficient algorithm. The methodology's major merit, however, emanates from the LG graph method, which allows us to build powerful and descriptive models for object recognition, with inherent invariance of rotation and scale. In our case, we demonstrate how it can be used to construct a model for recognizing faces of profile views, which is not usually considered, or not treated effectively.

5.2 Overview of the Methodology

Our methodology is based on a synergy of image processing, analysis and recognition processes and comprises two major modules, the potential face region (PFR) extraction module and the face detection module. The first module characterizes as regions of interest (ROI) regions that contain pixels with color values close to those of human skin. In the initial color image the pixels with color similar to that of the human skin are detected with a neural network trained based on the color constancy approach. Connected component analysis and morphological operations discard the regions that cannot contain face, based on their rough characteristics and the selected potential face regions that remain are refined, so as to compensate for probable errors having occurred during the previous steps. The refined PFRs are then processed by the second module, which decides which of those represent faces, by localizing their facial features. The aforementioned processing includes color segmentation, a feature enhancement process, and detection of the most significant corner points. In the final step, point sets are selected using randomized, smart search and are matched with our anthropometric model of a human face. For each point set, the Local-Global graph, which incorporates the spatial geometry of facial features, is constructed and compared to the face-model LG graph, using a computationally effective graph matching technique. The LG graph approach in general can be applied to any object detection problem, provided that there is a corresponding model in the LG database. An overview of the face detection algorithm is depicted in Figure 5.1.

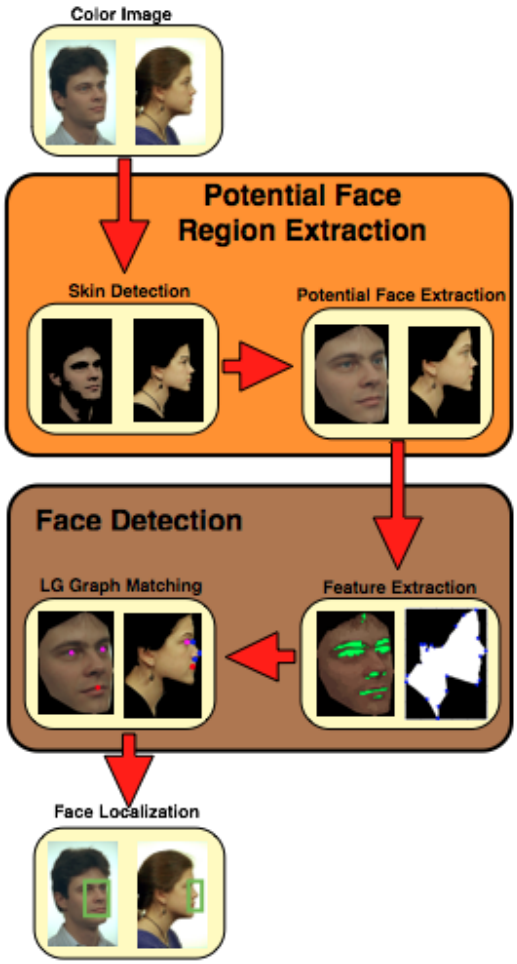


Figure 5.1: Overview of the face detection algorithm.

5.3 Skin detection

Our skin detection approach comprises two steps, namely color correction and skin classification [290–294]. In the color correction step, the color image's illuminant is re-estimated based on a Multilayer Perceptron Neural Network (MLPNN) and achromatic color (gray) is assigned to the skin pixels. The MLPNN has been trained so as to adapt to the skin color on randomly selected images from a database consisting of images collected under various illumination conditions both indoor and outdoor, and containing skin colors of different ethnic groups [290,295]. In the next step, the pixels are classified as skin or non-skin pixels using a simple thresholding technique in RGB space based on the achromatic value of the color corrected images.

The network is a Multilayer Perceptron with two hidden layers. The input layer consists of 1600 neurons, the first hidden layer has 48 neurons, the second hidden layer has 8 neurons and the output

layer has 2 neurons. The normalized input sRGB space is first transformed into rg chromaticity space, where r=R/(R+G+B) and g=G/(R+G+B). The input space (r,g) is divided into 40 * 40 (1600) discrete bins, each (r , g) histogram bin corresponding to one input neuron. Each input neuron can take 1 or 0 as input values, indicating that the chromaticity corresponding to the (r, g) histogram bin is either present or not present in the image. The output of the network is the expected (r,g) chromaticity of the illuminant in the image. The network is trained using the back-propagation algorithm with learning and momentum rates of 10 and 1 respectively. The error function was the Euclidean distance in (r, g) chromaticity space between the network's estimate and the provided expected estimate of the image illuminant. The network inputs ((r,g), (histogram bins) that have a non-zero input were marked active and only those active inputs were used during training. This pruning of the network [296] reduces the noise that may occur if the inactive inputs during training become active during testing. The advantage of using an MLPNN method for color adaptation is that no inherent assumptions are made about the surfaces of the objects in the image or the illumination sources as the input to the neural network is only the color from the image. In [290] it is shown that the overall proposed approach for skin detection is computationally inexpensive and is feasible for real-time applications.

The training and testing data set consists of 326 images [297] with the subjects face always in the upper left corner and with a white patch (to facilitate a reference for white patch). These images were captured by a digital camera, in and around the campus at Wright State University (WSU) over a number of days and during various timings of the day to include various illuminating conditions. Out of these, 255 images were randomly selected to form the training set and the remaining 71 images form the test set. For our tests two different network models were used.

Two models were built, the first having the white patch as reference for white and the second having the face region as a reference for the skin color. The MLPNN was trained so as to bring either the white patch to perfect grey or to bring the average face color to perfect gray in each training image. The image data is expressed in 2D normalized rg chromaticity space. During the training of the neural network, the pixels with all the three sensor (color) values in the range 11-254 were selected for our tests. The above filtering is done to remove pixels, which are too dark or too bright. To provide adequate data for training the two proposed NN models, a random set of 30,000 pixels was drawn 20 times from each image in the training set. These random sets of pixels formed the training sequence, creating an overall training set of 5100 (255 * 20) images. The pruning of the network resulted in 350 and 352 active neurons out of the 1600 inputs for each NN model respectively. Figure 5.2 shows results of the two steps of the skin detection process.

(a) (b) (c)

Figure 5.2: Skin Detection. a) Original image, b) Color corrected image, c) Image containing the skin detected pixels

5.4 Potential Face Region Extraction

The skin detection process produces many regions, among which might be faces, other skin regions and erroneous regions with pixels of skin-like color values. Connected component analysis is used to segment the most homogeneous areas and disconnect thin bridges. In order to clean up the result, mean shift filtering and morphological opening operation discard small pixel clusters, which are treated as noise.

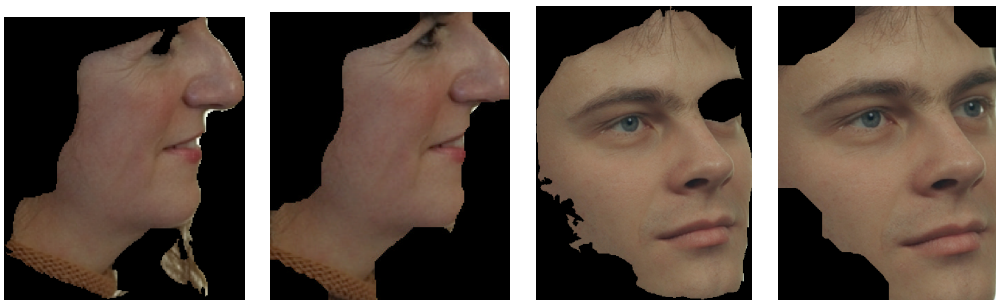
In the remaining regions, bridges among semantically different regions might be present, as depicted in Fig. 3 and reported in [298]. Connected component analysis alone might not be enough to alleviate this problem, mainly because of being sensitive to many application specific parameters. A more generally applicable approach that yields a better overview of the point distribution is the Distance Transformation (DT), which allows us to focus on large point clouds and discard thin lines. DT is first applied on each of the connected regions separately, and the area with the highest response is preserved, according to a threshold that varies depending on the area's size. The binary result of the transformation is loosely thresholded, so as not to damage the PFR, but enough to deal with thin bridges.

The holes in the selected connected components are filled with the morphological hole filling operation. One problem that might exist in this stage is the absence of the eye and eyebrow region, in the case of them being at the edge of the face or connected with hair and thus not surrounded by skin, so hole filling cannot restore them. In the case of the frontal view of a face, the convexity of the region allows us to use the PFR's convex hull as the PFR's region and achieve good recovery of

the lost features, while maintaining the shape of the face. However, it is not as easy to do the same in the case of the profile views, because it would result into restoration of many unwanted regions, which would impose increased computational load. Using a closing operation with an adaptively big circular structuring element yields better results. The PFR is classified as frontal or profile based on the concepts of ellipticity and convexity, as it will be described later. The discussed steps and cases are depicted in Figures 5.3 and 5.4.

(a) (b) (c)

Figure 5.3: Potential Face extraction with the refinement process. a) Original image, b) Skin regions, c) Potential Faces



(a) (b) (c) (d)

Figure 5.4: Restoration of profile and frontal face views

5.5 Feature Extraction

In this section, we describe the steps taken in the extraction of the features used in the face detection process, by forming the local and global relations of the LG graph. Briefly, image segmentation is

used to segment the examined region into sub-regions of interest, among which we expect to find those that represent the eyes and mouth, the most generally easy to discover facial features. For the profile view case, we concluded that the nose contains the highest amount of information, and the three points that represent a generic nose (beginning, tip, end) can be found through corner detection.

5.5.1 Potential Eye and Mouth Regions

Segmentation is one of the most common preprocessing steps in image analysis. The goal of segmentation is to partition the image into connected regions such that each region is homogeneous with respect to one or more characteristics. Each segment is composed of a continuous collection of neighboring pixels. When a segmentation algorithm terminates, each pixel in the image is assigned to a particular segment. During our experiments, several image segmentation algorithms were applied, the most prominent of which being a Statistical Region Merging (SRM) method [264], the widely used Mean Shift [265] algorithm and a Fuzzy Region Growing (FRG) method [299]. Each algorithm has each own merits and downfalls and we will focus on them in future research. In this work, we ended up using FRG image segmentation, because it is computationally inexpensive and gives an adequate amount of detail in the segmentation.

The Fuzzy Region Growing (FRG) segmentation method used in this research is a computationally efficient technique, which uses smoothing, edge information, homogeneity criteria and degree of dissimilarity to segment image regions. The algorithm first performs smoothing and edge operations to determine the interior pixels. A set of segments is then initialized by performing flood fill operations at the interior points. The decision as to whether a given adjacent (four- connected) pixel should be filled during the flood operation is based on its closeness in RGB color space to the original pixel seed of the segment. The pixels that have not been merged with any segment after the flood fill operation are merged appropriately through a region growing procedure. In order to merge the remaining pixels that are not assigned to any particular segment through region growing, the edge pixels of the existing set of segments are propagated outward or grown. Then, as unassigned pixels are encountered, they are merged with the closest segment of most similar color. This specific condition is calculated using a least squares difference of the RGB color components as well as a distance proportional to the distance from the original propagating edge pixel. Figure 5.5 shows an example application of the FRG segmentation. Clearly, the FRG method segments important facial feature regions such as eyes, eye-brows, nose and mouth.

However, applying the image segmentation to the original color image, will lead to the production of many regions and even worse, it is possible to merge the regions of interest with neighboring regions

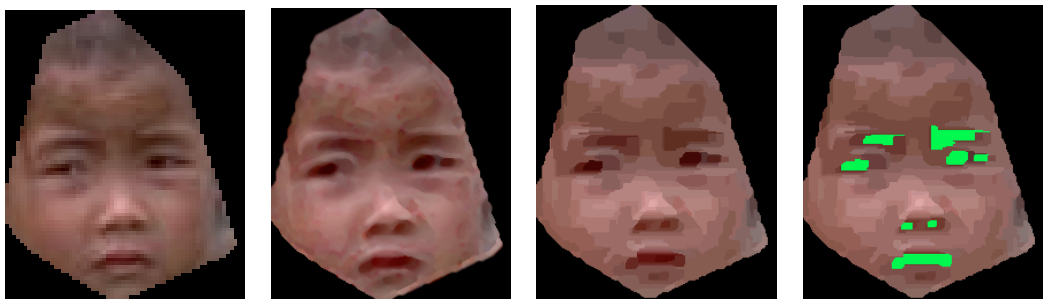
(a) (b)

Figure 5.5: Image Segmentation. a) Potential frontal face, b) Segmented version

of no interest, especially in low resolution images. The first effect imposes an increased computational load and the second might be catastrophic if the regions depicting the eyes and mouth get lost. To cope with these problems, we propose a preprocessing algorithm based on the observation that eyes and mouth appear darker than the skin regions. First, an unsharp filter is applied to the PFR, to enhance the contrast of the edges. The image is then decomposed to its HSV components (Hue, Saturation, Value), where the eyes and mouth colors have distinct response. In general, they might appear brighter in the Hue band, dark regions appear bright and bright regions appear dark in the Saturation band and the luminosity of the regions is unchanged in the Value band. Thus, in order to make the regions of interest stand out, we apply erosion, dilation and erosion in the H, S, V bands, respectively and extremum sharpening in the S and V bands, followed by mean filtering. This process results in growing the bright regions and shrinking the dark and also in increasing their contrast. Then, the image segmentation is performed, with parameters that preserve high amount of detail and produce many regions. The result is thresholded adaptively and the darkest regions are selected. This mask is used to select and merge connected regions that form the regions of interest, among which the mouth and eyes.

5.5.2 Nose Detection

When a PFR is expected to contain a profile view of a face, our additional objective is to discover possible nose regions. In the general case, the shape of the nose forms three corners, its beginning, tip and ending point, with respect to the face line. Thus, our initial step is the extraction of the face line, which is expected to be in the boundary of the region's contour. The contours at this point almost always are expected to contain sharp inwards and outwards bulges, so the contour is obtained using chain codes and smoothed using local regression lines. Then, the smoothed contour is

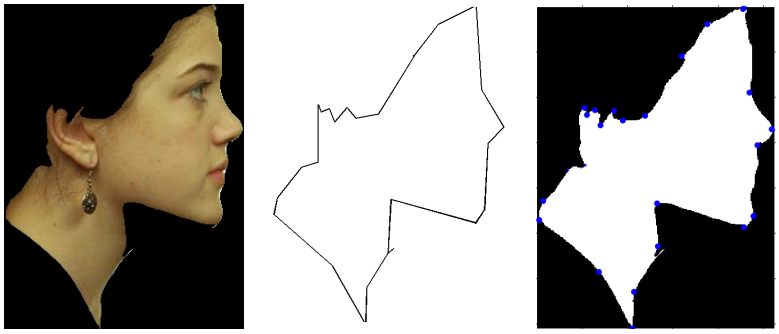


(a) (b) (c) (d)

Figure 5.6: Feature enhancement and selection for frontal views, a) original PFR, b) enhancement (morphological operations, extremum sharpening), c) image segmentation, d) selected facial features.

approximated by line segments, which will bring out the corner points more and the most significant corner points are selected.

However, even in the case of the region truly depicting a side view of a face, corner points will exist in areas of no interest, which will increase the computational power need and hinder the face detection process. We can reduce the number of these corner points by choosing those that have the potential of belonging to the desired nose points. To do so, we slide a 3 point width window over the corner points and eliminate the middle point if it forms too wide or too steep angle with the other two. The process is shown in Figure 5.7.



(a) (b) (c)

Figure 5.7: Corner detection for nose detection. a) Potential profile face, b) Contour approximation with line segments, c) Reduced set of useful corner points

5.6 LG Graph Based Matching

In this section we describe our anthropometric face models and the final step of the face detection, the LG graph matching, for both frontal and profile views of the face. Our methods have been designed in a way to provide a good trade-off between accuracy and complexity.

5.6.1 Frontal And Side Face Models

The anthropometric proportions of the average human face have been widely studied for decades from scientists of many different scientific fields, like medicine, psychology and art, to name a few. Face models have been widely used for face detection [298, 300, 301], because they provide simple, tractable and easily interpretable and applicable rules for defining the human face. Although the proportions of a head will vary from person to person and change slightly with age, there are some basic principles that describe the average face and appropriate selection of deviation thresholds can compensate up to an adequate point for these alterations.

As we can observe in Figure 5.8, a series of useful and descriptive principles can be inferred for the frontal and profile face modeling. It should be noted that these models are not irrelevant to each other, but the one is the projection of the other. Figure 5.8 shows this projection, which allows us to relate the most important facial features we use in this work. In the case of frontal views the centroids of the eyes and mouth form a circle and the nose points are located close to the diameter line that passes through the mouth's centroid. This relationship will be helpful when tracking of the face is also considered, which is part of our future work. The circle that includes the features of interest becomes an ellipse in the profile view, defined by the centroids of eye and mouth and the corner point representing the tip of the nose.

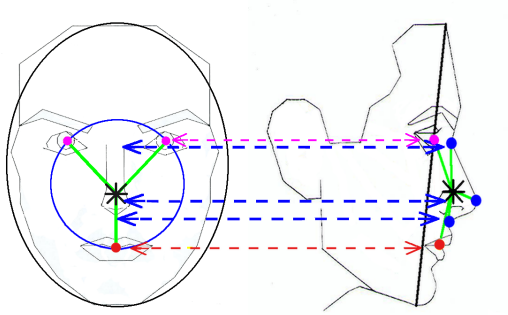


Figure 5.8: Frontal and profile face model. Pink dots denote the location of the centroid of the eye region, red of the mouth and blue of corner-nose points. The black star is their centroid.

Exploiting the geometric characteristics of the facial features and face not only allows us to

build an accurate face detection model, but it also implies an efficient manner to do so. First, we should say that when building the LG graph for the object, we do so in a localized and randomized manner, instead of building the graph connecting every point of interest in the examined object and rely on computationally expensive graph matching methods. The local information stored in each node along with the geometry, give an initial estimation about the nature of the region each node represents (e.g. nose points are defined by corner points and eyes are symmetric regions). Thus, when randomly selecting nodes to form the graphs to be matched with the model's LG graph, many of those will be discarded quickly if they do not meet a set of general constraints and not even reach the matching stage, thus saving computational load. The points are chosen randomly, because better average performance is achieved compared to brute force methods and along with smart searching augmentations, like the ones we just mentioned, searching can be speeded up substantially. Let us now see how these rules and methods apply to the cases of frontal and profile views of a face.

Model based face detection, such as the one discussed here, is inherently invariant of scale and rotation and if appropriate tolerance thresholds are set, it can even tolerate a small amount of shearing. Moreover, construction and rectification of models is easy and straightforward and can be generally applied to any kind of object detection, as long as the object poses some rigid geometric features. In our case, it allowed us to transit from the traditional and well studied frontal face detection to the additional detection of profile views of faces, accurately and efficiently. The final result is twofold, one being the estimation of the location and view type of the face (through the global relations of the LG graph) and also about the localization of the facial features (in accordance with the local information stored in the LG graph's nodes).

5.6.2 Matching

The global view of a model's LG graph can be seen in Figure 5.8. In the frontal view of a face, the regions of interest and nodes of the LG graph are the eyes and mouth and in the profile view the region of the eye, mouth and the three nose points (beginning, tip, end). The local position of each region is encapsulated in the position of its centroid (stored in each of the LG graph's nodes). The feature localization is done progressively, as we will see in the next subsections, by using both the local and global information of the LG graph.

The global information incorporates the spatial relations between the nodes, which are computed implicitly, through the angles and distances of the nodes (region centroids). The local information of each node, except for its position in the image, also denotes whether a node is a region's centroid or a corner point and in the case it represents a centroid's region, if it is a potential eye or mouth region.

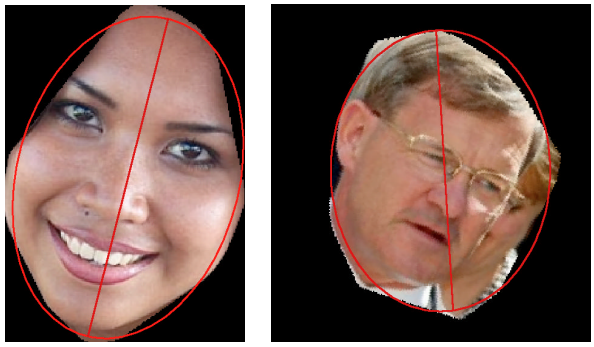
In both cases of frontal and profile views, the points that will be the nodes of the LG graph are selected randomly (3 centroids in the case of frontal view and 5 points in the case of profile view), as discussed above. Our initial model included the eyebrow and nostril regions too, defining a more descriptive model. However, we chose to make it more generic by using only the eye and mouth regions, because they are the most generally easy to observe among different faces and different image resolutions. In its full potential, the local information stored in each node can be more extensive, for instance the color or shape of the region can be stored too. Again, since we are dealing with low resolution and small images too, which do not contain adequate amount of detail, we chose to keep the essential information in the nodes. The next two subsections describe how this information is used to find and match the object's LG graph that corresponds to a face (if of course there is one) with the model's LG graph.

For the segmented regions, the concepts of ellipticity and convexity are used to evaluate the region's probability of depicting a frontal face region, profile face region or generic skin region. The ellipse that best fits the region's contour is estimated. If the ellipse is too elongated, meaning that $\frac{l_M}{l_m} < T_1$, where lM is the length of the major axis and lm is the length of the minor axis, then the area is classified as generic skin region. T_1 is an empirical threshold, set to 3. In the other case, the region is classified as frontal or profile based on the decision of a Support Vector Machine (SVM). The SVM has been trained over samples of frontal and profile faces and their selected features are the ratio of the areas of the region and its convex hull area, the ratio of the perimeter lengths of the region and its convex hull's, the modified Hausdorff Distances between the region and its convex hull's boundaries and their fitted ellipse. These features aim in capturing the aforementioned concepts of ellipticity and convexity. This pre-classification can be regarded as a speed up step. If the face has been pre-classified as frontal or profile, the corresponding matching algorithm, which will is described below) is performed first and if it is successful, the process stops there. In the case that the region has been classified as generic skin area and we choose not to discard it, both algorithms for frontal and profile detection are performed.

5.6.2.1 Frontal Face View

The fitted ellipse gives a rough insight about the general location of features. Eyes are expected to be above the minor axis, so the matching algorithm initiates search from this region. Eyes in the general case are mirrored with respect to the ellipse's major axis, so in order to detect this relation; the distances from the major axis and minor axis are measured. Any pair of centroids that satisfy this constraint is preferred, as it has the potential of depicting the pair of eyes. Thus, an initial small search space is formed. It should be noted, however, that this process assumes a good ellipse fitting,

so that the major axis indeed passes close to the middle of the distance between the eyes. This is not always the case though. To consider cases where the ellipse seems to have fitted adequately, but its orientation is not the desired one; we do not solely rely on the eye regions to be symmetric with respect to the major axis, but rather use it as a speed up step. Additionally, there might be the rare case where mirror regions do not represent the eyes. Thus, we define a small probability p to select two random regions as the potential eye pair. Finally, if the symmetric test fails, then regions expected to represent the eye regions will be randomly selected. Figure 5.9 shows both cases.



(a) (b)

Figure 5.9: a) Case where the speed up method is successful. The eye regions were detected as non-skin regions and are symmetric according to the major axis. b) The eye regions are not symmetric according to the major axis.

At this point, estimation about the possible location of the eyes has been made. Based on this estimation, the algorithm proceeds in searching for the mouth region. If the major axis mirrors the eye pair, the centroids below the minor axis are chosen as potential mouth regions, with higher priority. Else, a random point is selected. After all three centroids have been selected; the matching with the model frontal face is performed. Specifically, if the angles formed by the three centroids are approximately equal, then the algorithm has detected a frontal face. The steps of this process are outlined in Algorithm 1.

```
Extract potential eye and mouth regions and calculate their centroids;
for n = 1:M do
    if there are mirror regions then
        (1) Build triangle t from a random pair of centroids of mirror regions (eyes) and a
        random centroid of the remaining (mouth) Build graph g from the 3 region centroids
        if match(g,model) == TRUE then
            frontal face detected
        end
        else
            go to (1)
        end
    end
    else
        (2) Build graph g from three random centroids if match(g,model) == TRUE then
            frontal face detected
        end
        else
            go to (2)
        end
    end
end
```

Algorithm 1: Frontal face detection algorithm

5.6.2.2 Profile Face View

In this case, the most descriptive features are those that describe the nose, so the corner points that represent it are examined first. As we have already mentioned, these points have already been reduced into points that have the potential of forming a nose. The algorithm begins with a random selection of three corner points, which are assumed to form a nose. Then, the nearest centroids with a distance no longer than the largest distance between the selected corner points are picked out. This reduces the initial search space for this case.

From the selected POI's that potentially correspond to the eye and mouth regions, two points are randomly selected. Before proceeding to the matching process, some general geometric constraints are applied, in order to discard at an early point a combination of points which is highly unlikely to form facial characteristics and prevent unnecessary computational effort. Specifically, the distance between the eye and mouth has to be the longest and also the line segment that connects them cannot intersect with any of the segments that connect the nose points.

The matching with the object's LG graph can now be performed and culminate the detection process. In this case, the node correspondence issue between object and model would force us to use permutations of the object's points. In order to deal with it, we use the centroid of the five POI's as an assistive point. We calculate the distances of the selected nodes from their centroid and express them with respect to the smallest one. The angles formed by the centroid, the node corresponding to the smallest distance and the rest of the nodes are calculated. The same process has been performed in the model, and thus correspondence can now be achieved and the two graphs can be compared. The search space can be even reduced, if we guide the randomized selection of corner points, by for instance not allowing the selected points to have pair wise distances too big or too small, according to the potential face region's dimensions.

Extract potential eye and mouth regions and calculate their centroids;
Calculate potential nose points;
for *n = 1:M* **do**
  (1) Choose three potential nose points and two potential facial region centroids from their neighborhood **if** *general geometric constraints are met* **then**
    Calculate the centroid of the selected nodes and the angles with respect to the centroid and the node with the smallest distance to it;
    Construct graph g;
    if *match(g,model) == TRUE* **then**
      profile face detected
    end
    else
      go to (1)
    end
  end
  else
    go to (1)
  end
end

Algorithm 2: Profile face detection algorithm

5.7 Experimental Results

During the development and experimentation, three well-known publicly available image databases were used. In their entirety, they encompass images varying in their resolutions, views of faces, number and type of subjects, complexity of scene, etc. Portions of the research in this work use FERET database of facial images collected under the FERET program, sponsored by the DOD

Counterdrug Technology Development Program Office [302, 303]. FERET database contains high resolution images of clear faces in simple background and was used first to test the validity of our methodology. The results were very promising, so we then extended our tests to more complex databases, the Faces in the Wild database [304] and the UCD Color Face Image Database for Face Detection [305]. As expected, due to the nature of our model based face detection, the latter database contained the most images were detection would fail. A description of the image databases follows:

The Color FERET Database: The database contains 1564 sets of images for a total of 14,126 images that includes 1199 individuals and 365 duplicate sets of images. A duplicate set is a second set of images of a person already in the database and was usually taken on a different day. Both frontal and side views are included and the images depict the face area with a simple background.

Faces in the Wild: The database consists of 30,281 faces collected from News Photographs. The images have good quality and depict complex scenes, often multiple humans, who are captured in various poses.

UCD Color Face Image Database for Face Detection: The database contains color pictures of faces having a high degree of variability in scale, location, orientation, pose, facial expression and lighting conditions. These images are acquired from a wide variety of sources such as digital cameras; pictures scanned using photo-scanner, other face databases and the World Wide Web. It is a small, but challenging dataset.

Figure shows some interesting examples from these datasets. Figure 5.10(a) exhibits a typical frontal view face detection result, with adequately visible faces in a complex background. In general, skin detection reduces the search space for faces and discards cluttered regions that could create false detections even to robust algorithms, like the well known state-of-the-art well-known Viola-Jones [289] algorithm. Figure 5.10(b) exhibits a case of a partially occluded face that our rectification process managed to recover and thus allow the algorithm to detect it. It is of interest that Viola-Jones failed to detect this face, although most of the facial features are in general visible. Figure 5.10(a) is a case of rotated faces, where Viola-Jones does not perform so well. Our technique however, although simple, can inherently cope with rotations because only the local structural relations of the facial features are taken into account and thus it is invariant of rotations. The other cases show the ability of our algorithm to detect faces even when the facial features are not as clear, but more importantly when the faces are in profile view. Although Viola-Jones can cope successfully with the former challenge, it does not perform as well in the latter. The reason is that the features it employs express better structures with "blockiness" rather than ones that are represented better by their contours.

However, although the first results were promising, the algorithm has to be improved in order

Figure 5.10: Experimental results. The first image corresponds to Viola-Jones face detection result and the second to our method.

to outperform adequately the state-of-the-art. Skin detection is a natural approach to guide face detection, but in the future we will use it as a cue for face localization rather than strongly rely on it. The graph matching technique is efficient and powerful, because it can deal with many structural relations and remain scale and invariant to in-plane rotations. More attention should be given to out-of-plane rotations however, because there are still corner cases were there is not enough information for the profile case (boundary is not clear) and at the same time the distances between the facial features are distorted due to perspective transformations and cannot match the frontal face model. Thresholds can cope to a good extent with this problem, but we plan to make the graph matching more flexible. Towards this direction more features could be used. Here we demonstrated that image

segmentation and corner detection are effective and intuitive ways to extract salient features, but performance could be enhanced if other types of features are added to the LG graph models as well.

6

Human Body Extraction

6.1 Introduction

In this chapter we deal with the extraction of the human body in a single image. Extraction of the human body in unconstrained, still images is a very challenging task due several factors. The most important among them are shading, image noise, occlusions, background clutter, the high degree of human body deformability and the unrestricted positions due to in and out of the image plane rotations. Knowledge about the human body region can further benefit various tasks, such as determination of the human layout [306–309], recognition of actions from static images [44,310,311] and sign language recognition [312,313]. Human body segmentation and silhouette extraction has been a common practice when videos are available in controlled environments, where background information is available and motion can aid the segmentation through background subtraction. In static images however, there are no such cues and the problem of silhouette extraction is much more challenging, especially when we are considering complex cases. Moreover, methodologies that are able to work at a frame level can work for sequences of frames too and facilitate successfully already existing methods for action recognition based on silhouette features and body skeletonization.

In this work, we propose a bottom-up approach for human body segmentation in static images. We decompose the problem into three sequential problems, namely face detection, upper body extraction and lower body extraction, since there is a direct pairwise correlation among them. The initial step of the process is face detection, which first gives a strong indication about the presence of humans in an image, greatly reduces the search space for the upper body, provides information about the skin color adaptive to the specific human and allows the formulation of an appearance based method combined with heuristics emanating from anthropometry. Similarly, the upper body region guides the extraction of the lower body extraction. Finally, upper body extraction provides additional information about the position of hands, the detection of which is very important for

several applications. The basic units upon which calculations are performed are super pixels from multiple levels of image segmentation. The benefit of this approach is twofold. First, different perceptual groupings reveal more meaningful relations among pixels and a higher, however abstract, semantic representation. Second, noise at the pixel level is suppressed and region statistics allow for more efficient and robust computations. Instead of relying on pose estimation as an initial step or making strict pose assumptions, we enforce soft anthropometric constraints to both search a generic pose space and guide the body segmentation process. One of the most important principles employed here is that body regions should be comprised by segments that appear strongly inside the body regions and weakly in the hypothesized background.

The major contributions of this work are as follows:

- We propose a novel framework for automatic segmentation of human bodies in single images.
- We combine information gathered from different levels of image segmentation, which allows efficient and robust computations upon groups of pixels that are perceptually correlated.
- Soft anthropometric constraints permeate the whole process and uncover body regions.
- Without making any assumptions about the foreground and background, except for the assumptions that sleeves are of similar color to the torso region and the lower part of the pants being similar to the upper part of the pants, we structure our searching and extraction algorithm based on the premise that colors in body regions appear strongly inside these regions (foreground) and weakly outside (background).

6.2 Related Work

The problem of human body segmentation from images has gained increased attentions from the research community over the last decade and several solutions have been proposed. We could classify these approaches into three categories. The first one includes *interactive* methods, which, as the name suggests, expect user input in order to discriminate the foreground and background. In general this category differs from the other two, which are automatic and often task-specific. The second category includes *top-down* approaches, which are based upon a-priori knowledge and use the image contents to further refine the initial model. On the other hand, *bottom-up* approaches, which form the third category, use low-level elements such as pixels or superpixels and try to group them into higher level semantic entities.

Interactive segmentation methods are very useful for generic applications and have the potential to produce very accurate results in complex cases. However, since they rely on low-level cues and

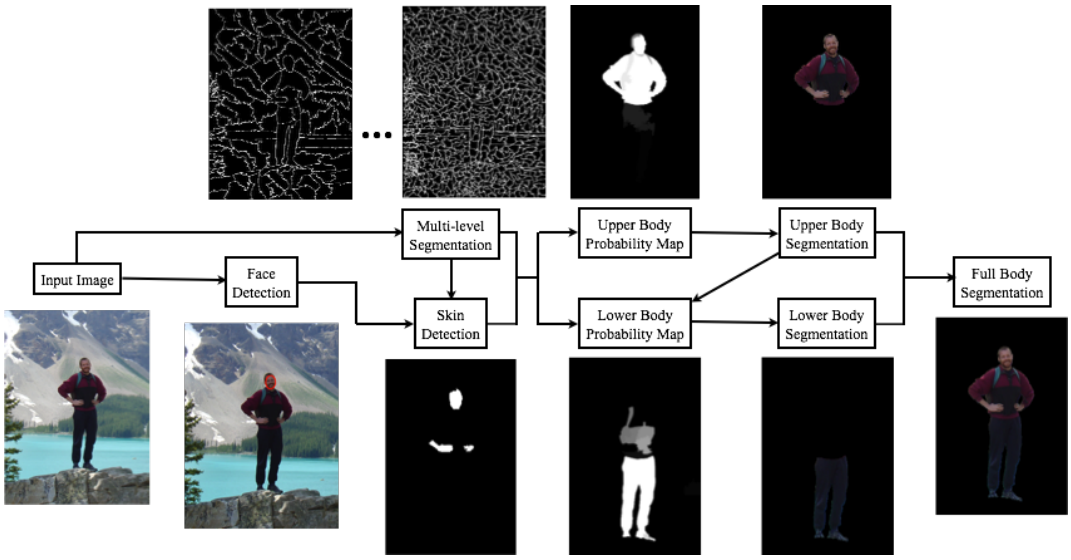


Figure 6.1: Overview of the methodology.

do not employ object-specific knowledge, they often require user input to guide their process and thus become inappropriate for many real-world problems where automation is necessary. In [314], the KDE-EM approach is introduced by applying nonparametric kernel density estimation method in EM-based color clustering. One of the most popular interactive segmentation algorithms is GraphCut [315], where the user selected the foreground and background pixels are nodes of a graph (in MRF fashion) that is partitioned according to the global solution of max-flow algorithm [316]. The cost function over the graph that is to be minimized penalizes nodes of dissimilar color and favors the cut of edges that represent strong edges in the image. GrabCut [317] extends this work by introducing an iterative process and incomplete labeling to the mechanism, which aims in reducing the user interaction, dealing with cases where the foreground is more similar to the background and providing more appealing end results through border matting. Another type of iterative process is implemented in GrowCut [318], which uses Cellular Automaton as an image model. The model represents the image as a grid of cells with strength defined by their pixel values and during during automata evolution neighboring cells "attack" one another iteratively until convergence. A very interesting algorithm is Active Segmentation with Fixation (ASF) [319], which only requires the user to select the point enclosed by the foreground object to be segmented. The most important step of this algorithm is finding the edge enclosing the fixation that best partitions the image in log-polar space. However, although the algorithm produces impressive results with minimal input, it requires a clear and strong boundary between the object of interest and the background. In the same spirit, Geodesic Star Convexity (GSC) [320] extends the algorithm presented in [321] by introducing

multiple stars and geodesic paths. User selected star centers aid the algorithm in outlining the boundary of the foreground object. Random Walks are successfully employed in [322] for multi-label segmentation, where random walks from unseeded regions to seeded are used to estimate the probability of each pixel to reach the seeded points and the final result is calculated by solving a sparse system of equations. Attention has been given to preserve even weak boundaries during random walks and treat cases were object boundaries have small discontinuities.

Recently several top-down approaches have been proposed as solutions to the problem of segmenting human bodies from static images. The main characteristic of these approaches is that they high-level knowledge about the foreground, which in case of humans is their pose. One of the prominent and well studied method for object recognition and pose estimation is the Pictorial Structures (PS) [323]. In its original formulation, PS uses a tree-structured graphical model to make probabilistic inference and poses are found through minimization of a cost function decomposed along the edges and nodes of the tree, where the torso usually serving as the tree's root since it is usually the most visible part, although other types of trees have been proposed too. The PS method has been extended in several ways over the last years. Some of the most important contributions to it appear in [324], where the original PS model is extended to encode the fact that symmetric limb pairs have consistent color and in [325], where PS are augmented by a novel approach for estimating part appearance models. Both reasons are studied in [308]. The authors show how a pre-processing step called foreground highlighting, which removes part of the background clutter by using grabcut algorithm to reduce the searching space of PS models and subsequently implies that bottom-up approaches can improve the results and efficiency of top-down approaches.

Among the most important works that object priors to perform the segmentation are [326] where object category specific MRF is used and [327], where pose-specific MRF is used. The local contrast-dependent MRF is combined with strong global priors emanating from the PS model in [326] and the stickman model in [327], which focuses in the human object category. In [328], bottom-up cues are combined with global top-down knowledge for object class learning with unsupervised segmentation, where the object's shape is learned but not its appearance, so cannot be used achieve high accuracy in cluttered images. In [329] the problem of segmenting occluded object is tackled, where preliminary solutions to segmentation are found with a layout-consistent random field. Shape templates are another way to store object-specific knowledge and are used in [330] to select segments of a segmentation hierarchy that lead to object-like segmentations and in [331] with a KDE-EM framework. The body probability map obtained by the PS model is used as prior knowledge in [332] and it is refined with a superpixel-based EM-like algorithm so as to guide a novel l_1 based graph cuts algorithm. Similarly in [333], cues from PS model are used along with appearance and

spatial constraints to provide good seed determination for a graph cut algorithm. In general, these approaches can deal with various poses and produce impressive results, but they rely on complex high level models that might fail in complex scenarios and often the rest of algorithm cannot recover from these failures. Besides, high-level inference is time consuming and thus these methods usually are computationally expensive.

In bottom up approaches higher level human body segmentation is conducted by grouping lower level elements, such as pixels or superpixels and relying to simpler constraints, heuristics and high level concepts. One popular work that has propelled many others is [334], which is mainly designed for pose detection. Poses are treated as assemblies of parts, identified by specific detectors over superpixels acquired using Normalized Cuts [335]. This method can deal with many unusual poses, but it exploits properties of a specific dataset and from a segmentation perspective, several parts of the human body might be missing. In [336], candidate regions generated by directed aggregation of superpixels are scored based on shape similarity to a database of shape exemplars and assemblies with variable numbers of parts are scored using a simplified hierarchical model of appearance. Advances in face detection methodologies facilitate significantly the related problem of human body segmentation and many works utilize them because they contain significant cues that can be used towards body localization and skin color estimation. In [337], extract the face region using GrabCut and model the skin color using GMMs. Then, they initialize a trimap for the human body in form of a rectangle under the face in order to get a first estimation using GrabCut, which is iteratively refined after sampling small patches along the hypothesized contour. Although the refinement process is promising, it requires unbounded executions of the time consuming GrabCut algorithm and more importantly, the initialization of the process restricts the generalization of the algorithm significantly. The same cues are used in similar manner in [338], where the main contribution lies in a novel torso fitting method [339] and discrimination of the foreground region through constrained Delaunay triangulation. In their work, only the clothes of the upper body are deemed as foreground, as seen in frontal, upright views. In [340] two segmentation levels are employed, where the fine level aims in collecting consistent features from uniform regions and the course level aims in preserving local shape during classification, which is performed using AdaBoost. The authors use the torso detector of [341] and perform full-body segmentation. However, their results are restricted to frontal upright poses and their sampling method for clothes and pants assumes uniform and continuous regions ad accurate torso fitting.

6.3 Face Detection

Localization of the face region in our method is performed using OpenCVs implementation of Viola-Jones algorithm. Besides the obvious reasons of high performance and speed, this algorithm is based on combinations of a vast pool of Haar-like features, which essentially aim in capturing the underlying structure of a human face, regardless to the its skin color. Since skin probability in our methodology is learnt from the face region adaptively, we require an algorithm that is based on structural features.

However, Viola-Jones face detector is prone to false positive detections that can lead to unnecessary activations of our algorithm and faulty skin detections. In order to refine the results of the algorithm, we propose using the skin detection method presented in [295] and face detection algorithm presented in [225]. The skin detection method is based on color constancy and a Multilayer Perceptron Neural Network (MLPNN) trained on images collected under various illumination conditions both indoor and outdoor, and containing skin colors of different ethnic groups. The face detection method is based on facial feature detection and localization using low-level image processing techniques, image segmentation and graph-based verification of the facial structure.

First, the skin probability map of the image pixels is calculated. Then, the elliptical regions of the detected faces in the image found by the Viola-Jones algorithm are evaluated according to the probabilities of the inscribed pixels. More specifically, the average skin probability of the pixels X of potential face region FR_i, for each person i, is compared to threshold $T_{GlobalSkin}$ (set empirically to 0.7 in our experiments). If it passes the global skin test (greater than $T_{GlobalSkin}$), it is further evaluated by our face detector. If the facial features are detected, then FRi is considered to be a true positive detection.

(a) (b) (c)

Figure 6.2: Face detection and verification, a) Viola-Jones face detection, b) global skin detection, c) facial feature and face detection

6.4 Anthropometric Model

The basic source of knowledge that permeates the whole hands detection methodology derives from anthropometric studies that describe the structural composition of the human body. Extraction of the face region allows us to estimate the most important structural block of the body composition, the palm length. In the average human body the major axis of the face's ellipse is almost equal to the length of the palm. Let this distance be called PL from now forth. The location and size of the rest of the body components of interest can be estimated using this distance and the center of the face, $(x_{FaceCenter,i},\ y_{FaceCenter,i})$.

The base of the neck or beginning of chest in an almost frontal and upright pose can be approximately found at one PL directly below the face center. Let the coordinates of this point be $(x_{NeckBase,i},\ y_{NeckBase,i})$. The edges of the perpendicular line segment with length of $2PL$ that passes through this point indicate the approximate location of shoulders or else the joint of the upper arms with the chest. The neck base point can also be used to define the space where the hands can be found. It is used as the center for a series of concentric ellipses that aid in better understanding the hand position and arm posture. The outer ellipse has a major axis length of $5PL$ and beyond this ellipse possible hand regions are automatically rejected. The zone between this ellipse and the next one, which has a major axis length of $4PL$, shows the zone where palms can be when the arms are fully extended. Finally, the most inner ellipse has a major axis length of $2.5PL$ and corresponds to the ellipse delineated by the elbows when rotated in extension. This ellipse aims in giving an indication about the degree the arm is bent.

Extending the anthropometric model to the side-view case, we show that the ellipses in the frontal view are actually ellipsoid spheres. The depth information is lost in single monocular images and it is one of the reasons that hand detection in these images is a difficult problem. In our methodology we do not explicitly deal with depth information but rather rely on body part connectivity, which can be detected in the restricted 2D space of the frontal model.

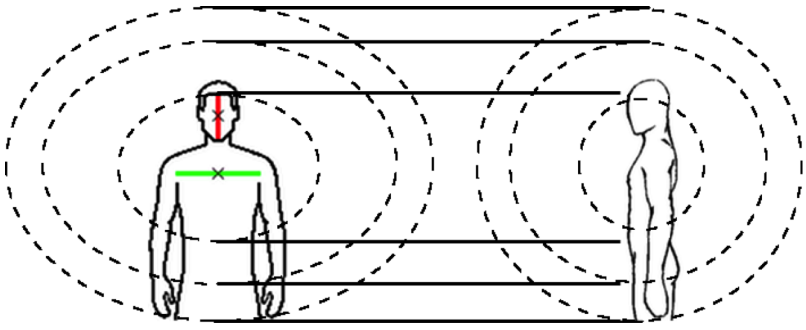


Figure 6.3: Anthropometric model.

6.5 Multiple-level Image Segmentation

Relying solely on independent pixels for complicated inference leads to propagation of errors to the high levels of image processing in complex, real-world scenarios. There are several different sources of noise, such as the digital sensors that captured the image, compression or even the complexity of the image itself and their effect is more severe at the pixel level. A common practice to alleviate the noise dwelling at the pixel level is the use of filters and algorithms that extract more collective information from the pixels. Moreover, groups of pixels express higher meaning. Smaller groups preserving detail and uniform patches and larger groups tend to capture shape and more abstract structures better. Finally, computations based on superpixels are more efficient and facilitate more flexible algorithms.

In this work, we propose using an image segmentation method, in order to process pixels in more meaningful groups. However, there are numerous image segmentation algorithms and the selection of an appropriate one was based in the following criteria. First, we require the algorithm to be able to preserve the strong edges in the image, because they are a good indication of boundaries between semantically different regions. Second, another desirable attribute is the production of segments with relatively uniform sizes. Studies on image segmentation methods [223, 253] show that although these algorithms approach the problem in various different ways, in general they remain in the low levels of image processing and thus their results cannot be guaranteed to comply with the various and subjective human interpretations. Thus, we deem this step as a high-level filtering and prefer to over-segment the image, so as not to lose great amount of detail. Region size uniformity is important because it restrains the algorithm from being tricked by over-segmenting local image patches of high entropy (e.g. complex and high detailed textures) to the expense of more homogeneous regions that could be falsely merged, although they belong to semantically different objects (e.g. human hand over a wooden surface with color similar to skin).

Using superpixels instead of single pixels has been seen before in the well-known work [334], where the authors use Normalized Cuts [335] to segment the image. The method we adopt in this work is the Entropy Rate Superpixel Segmentation (ERSS) algorithm, proposed in [263], which provides a good trade-off between accuracy and computational complexity. This approach is based on optimizing an objective function consisting of two components, the entropy rate of a random walk on a graph and a balancing term. Results of the ERSS are shown in Figure 6.4. More importantly, we propose using multiple levels of segmentations, in order to alleviate the need of selecting an appropriate number for the regions to be created and combine information emanating from different perceptual groupings of pixels. Although our framework can accept any number of segmentation levels, we find that two segmentation levels, of 100 and 200 segments, provide accurate results and

efficient computations. For the skin detection algorithm, a finer segmentation of 500 superpixels is used, because it manages to discriminate better between adjacent skin and skin-like regions and recover skin segments that are often smaller compered to the rest image regions.

(a) (b) (c)

Figure 6.4: Image segmentation for 100, 200 and 500 superpixels.

6.6 Skin Detection

The problem of detecting skin regions in images and video has propelled many works and is still an open problem. Among the most prominent obstacles are the skin tone variations due to illumination, ethnicity, etc., disambiguation of skin like regions and the fact that limbs often do not contain enough contextual information to successfully solve the ambiguities. Here we propose combining the global detection technique [342] with an appearance model created for each face, so as to better adapt to its skin color. The appearance model provides a strong discrimination between skin and skin like pixels and segmentation cues are used to create regions of uncertainty. Regions certainty and uncertainty comprise a map that guides the GrabCut algorithm, which in turn outputs the final skin regions. Even at this step there might be still false positive cases, which are eliminated using anthropometric constraints and body connectivity.

Each face region FR_i is used to construct an adaptive color model for each persons skin color. In this work we propose using the r, g, s, I, Cr and a channels. In more detail, $r = R/(R+G+B)$, $g = G/(R+G+B)$ and $s = (R+G+B)/3$, so r and g are the normalized versions of the R and G channels respectively and s used instead of b to achieve channel independence. Channels I, Cr and a from YIQ (or NTSC), YCbCr and Lab colorspaces, respectively, are chosen because skin color is accentuated in them. The skin color model for each person is estimated after fitting a normal distribution to each channel, using the pixels in each FR_i. The parameters that represent the model are then the mean values μ_{ij} and standard deviations σ_{ij} for each FR_i and channel $j = 1 \dots 6$ for

channels r, g, s, I, Cr and a. Then, each image pixels probability of being a skin pixel is calculated separately for each channel according to a normal probability distribution with the corresponding parameters. We require that a true skin pixels should have strong probability response in all of the selected channels, so the skin probability for each pixel X is:

$$P_{Skin_i}(X) = \prod_{j=1}^{6} \mathcal{N}(X, \mu_{ij}, \sigma_{ij}) \tag{6.1}$$

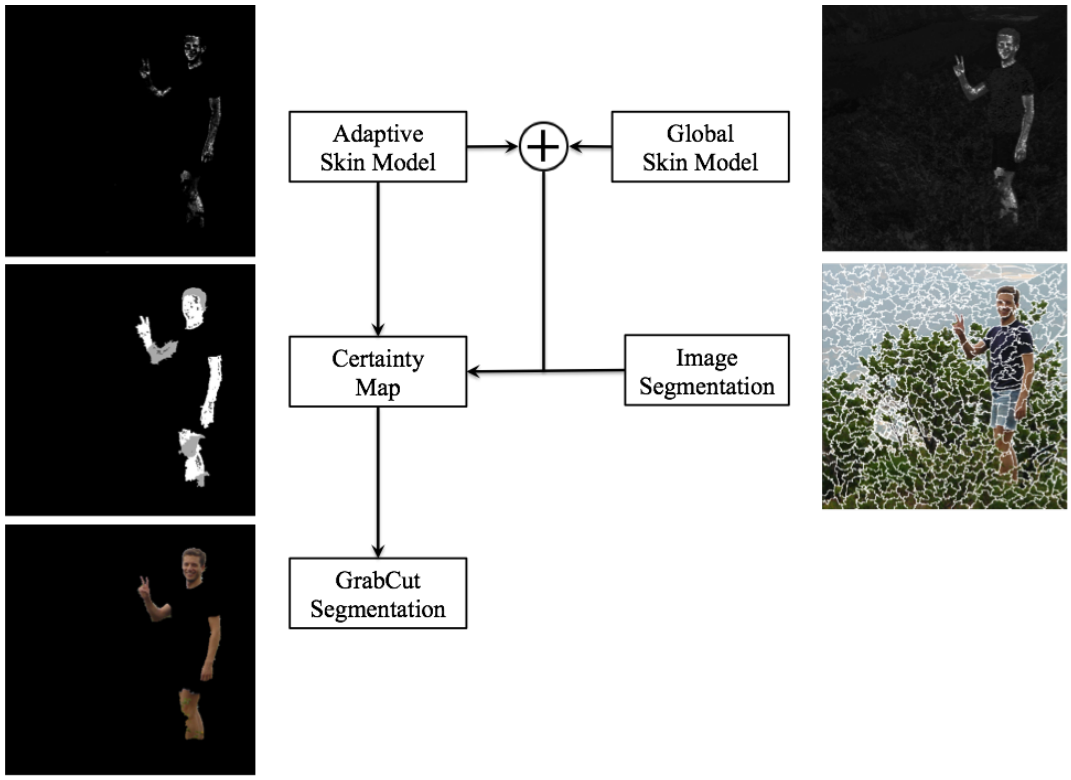


Figure 6.5: Skin detection algorithm.

One characteristic of the adaptive model manages to retrieve the true positive cases. Examples can be seen in Figure 6.6. However, most of the time it is too "strict" and suppresses the values of many skin and skin like pixels that deviate from the true values according to the derived probability distribution. At this point, we find that an influence of the skin global detection algorithm is beneficial because it aids in recovering the uncertain areas. Another reason we choose to extend the skin detection process is that relying solely on an appropriate colorspace to detect skin pixels is often not sufficient for real world applications [343]. Combination of the two proposals is done through weighted averaging (with a weight of 0.25 for the global model and 0.75 for the adaptive model). The finest level of image segmentation is used at this point to characterize segments as

certain and probable background and probable background. Using segments instead of pixels makes the classification more robust, because the perceptual information they convey is exploited and leads to clearer maps for the GrabCut algorithm. For the certain foreground regions however, only the pixels with sufficiently high probability in the adaptive model are used as seeds, so as to control their strong influence. In order to characterize a region as probable background or foreground, its mean probability of the combined probability must be above a certain threshold (empirically set to 0.2 and 0.3, respectively).

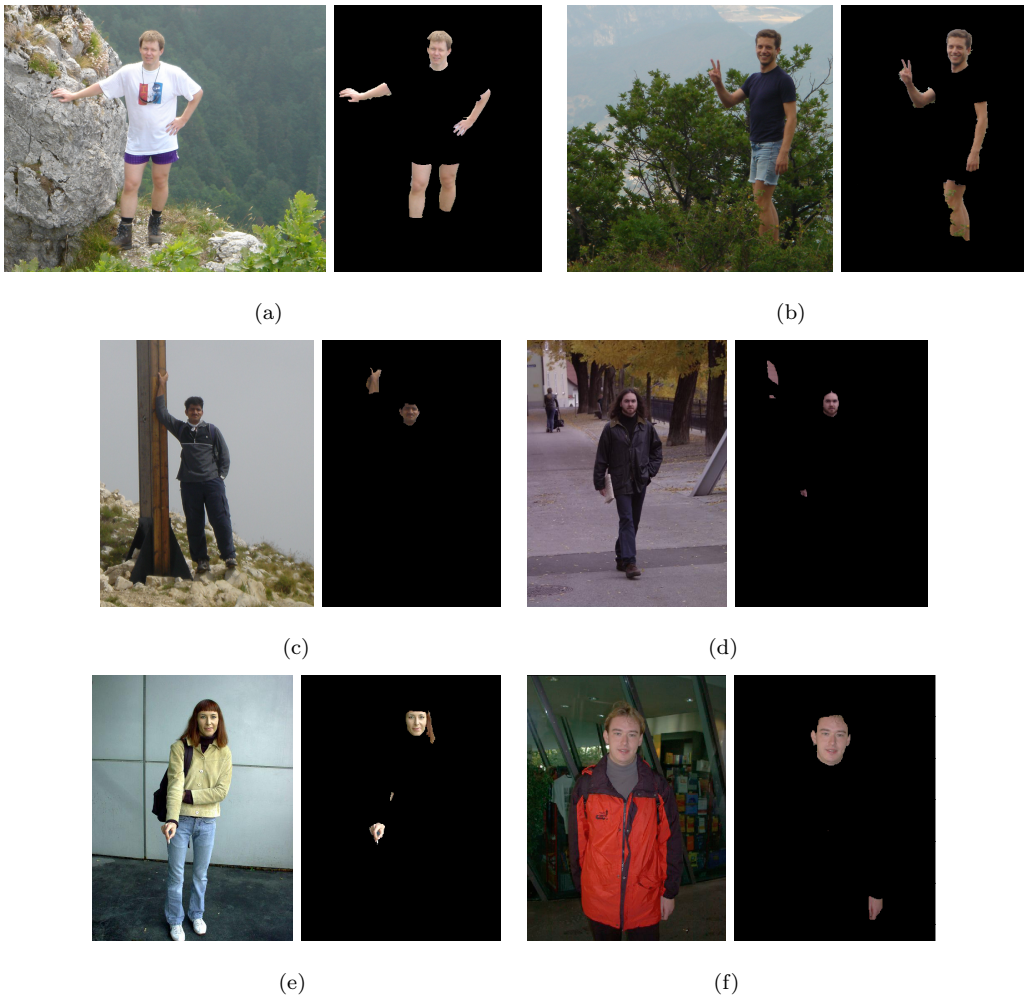


Figure 6.6: Skin detection examples.

6.7 Upper Body Segmentation

In this section, we present a methodology for extraction of the whole upper human body in single images, extending out work in [343] which dealt with the case where torso is almost upright and facing the camera. The only training needed is for the initial step of the process, namely the face detection and a small training set for the global skin detection process, but it is not crucial for the skin detection algorithm. The rest of the methodology is mostly appearance-based and relies on the assumption that there is a connection and continuity between the human body parts. Color-based similarity combined with anthropometry provide sufficient information to support an efficient and effective hands detection methodology, without the need of training data or complex inference about the exact human body pose. Furthermore, we demonstrate how processing using superpixels instead of single pixels, which are acquired by an image segmentation algorithm, yield more accurate results and allow more efficient computations.

The initial and most crucial step in our methodology is the detection of the face region, which guides the rest of the process. The information that can be extracted in this step is significant. First, the color of the skin in a person's face can be used to match the rest of their visible skin areas, making the skin detection process adaptive to each person. Second, the location of the face gives a strong cue about the rough location of the torso. Here, we deal with cases where the torso is below the face region but without strong assumptions about in and out of plane rotations. Third, the size of the face region can further lead to the estimation of the size of body parts according to anthropometric constraints. The general scheme of the methodology can be seen in Figure 6.1.

Face detection is based on OpenCV librarys implementation of the widely used Viola-Jones face detection algorithm, for both frontal and side views. Since face detection is the cornerstone of our methodology, we refine the face detection results by using face detection algorithm. Once the elliptical region of the face is known we proceed to the foreground probability estimation. In order to better utilize the existing spatial and color relations of the image pixels we perform multiple level over-segmentation and examine the resulting superpixels. We regard as skin the superpixels with color similar to that of the face region and as clothes the superpixels with color similar to the regions inside torso masks, hypothesized using anthropometric constraints and dissimilar to the rest of the image. As opposed to other approaches that are based to pose estimation, we employ simple heuristics to conduct a fast and rough torso pose estimation and guide the segmentation process.

Torso is usually the most visible body part, connected to the face region and in most cases below it. Using anthropometric constraints one can roughly estimate the size of the torso and its location. However, different poses and head motion make torso localization a challenging task, especially when the assumption about its pose are relaxed. Instead of searching for the exact torso region or using

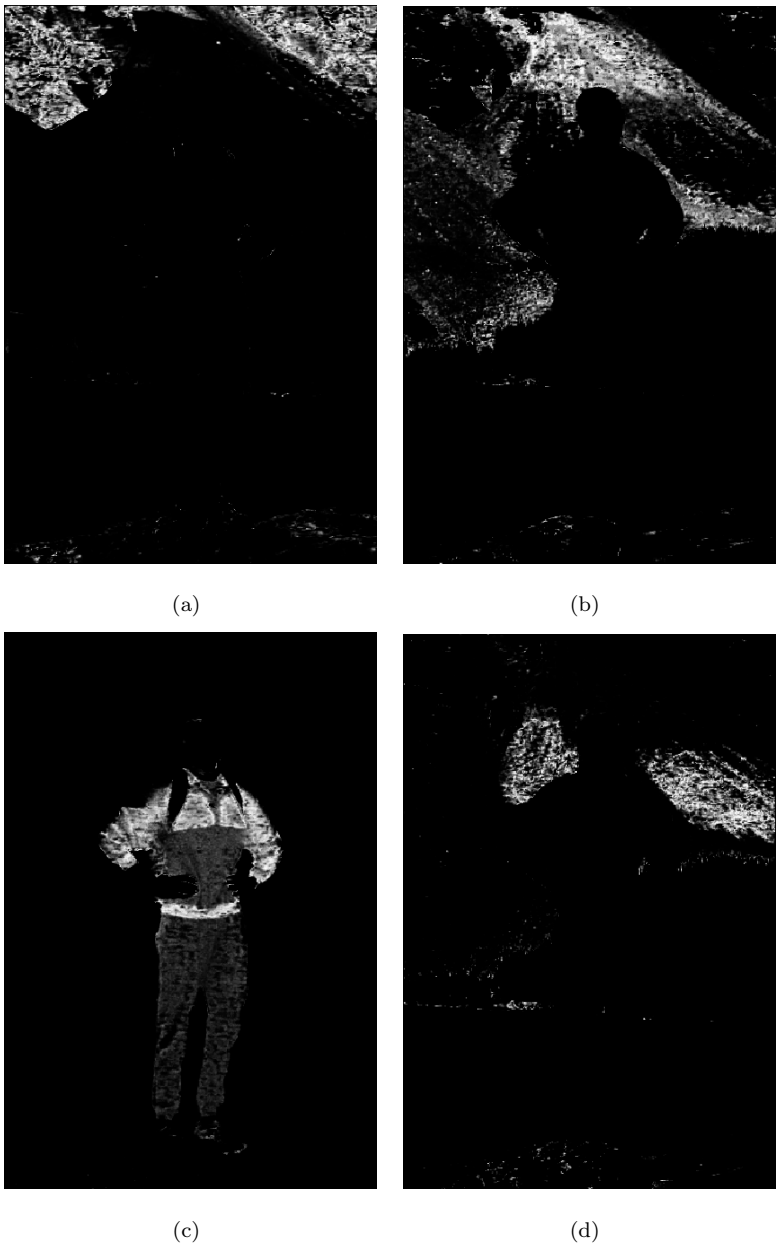
complex pose estimation methods, we propose using a rough approximation of the torso mask select the regions appear mostly in the mask and not so much in the background. This simple and intuitive criterion allows for fast inference about the torso location and relieves the need for the complex task of explicit torso estimation, without however sacrificing accuracy.

As discussed above, different levels of segmentation give rise to different perceptual pixel groupings and each segment is described by the statistics of its color distribution. In each segmentation level, each segment is compared to the rest and its similarity image is created, depicting the probabilistic similarity of each pixel to the segment. Similarly to the skin detection process, the mean μ_i and standard deviation σ_i of segment S_i construct normal distributions in each channel $j = 1, 2, 3$ of the Lab colorspace and the probability for each image pixel of belonging to this probability is calculated. We estimate the final probability as the product of the probabilities (Eq. 6.4) in each channel separately. Example similarity images are shown in Figure 6.7. The main rationale behind this process is that there should be big concentration where the body parts lie of regions that are similar among them and at the same time dissimilar to the rest of the image. Similarity images are gathered for all of the different segmentation levels l. Here, we use two segmentation levels in this stage, of 100 and 200 super pixels, because they provide a good trade-off between perceptual grouping and computational complexity.

$$P_{SimilarityImage_{li}}(X) = \prod_{j=1}^{3} \mathcal{N}(X, \mu_{ij}, \sigma_{ij}) \tag{6.2}$$

Sequentially, a searching phase takes place, where a loose torso mask is used for sampling and rating of regions according to their probability of belonging to torso. Since we assume that sleeves are more similar to the torso colors than the background, this process combined with skin detection actually leads to upper body probability estimation. The mask is used for sufficient sampling instead of torso fitting, so it is estimated as a large square with sides of $2.5PL$, with the top most side centered with respect to the face's center. In order to relax the assumptions about the position and pose of the torso, the mask is rotated by 30 degrees left and right its initial position (0 degrees), as seen in Figure 6.8. By using a large square mask and allowing this degree of freedom, we manage to sample a large area of potential torso locations. On the other hand, by constraining its size according to anthropometric constraints, we make the foreground/background hypotheses more meaningful.

During the search process, the mask is applied to each similarity image and the segment it corresponds to is rated. Let $TorsoMask_t$ be a binary image where pixels are set to 1 (or "on") inside the square mask and 0 (or "off") outside, so that $SimilarityImage_{li} \cap TorsoMask_t$ selects the probabilities of the similarity image that appear inside the mask. Index $t = 1, 2, 3$ corresponds to a torso mask at angle -30, 0 or 30. Thus, Equations 6.3 and 6.4 rate each segment's potential of



(a) (b)

(c) (d)

Figure 6.7: Example of similarity regions for random segments (each image corresponds to one segment).

belonging to foreground and background, respectively and Equation 6.5 combines the two potentials in form of a ratio:

$$P_{Foreground}(S_{tli}) = \sum^{|S_{tli}|} SimilarityImage_{li} \cap TorsoMask_t \tag{6.3}$$

$$P_{Background}(S_{tli}) = \sum^{|S_{tli}|} SimilarityImage_{li} \cap \overline{TorsoMask_t} \tag{6.4}$$

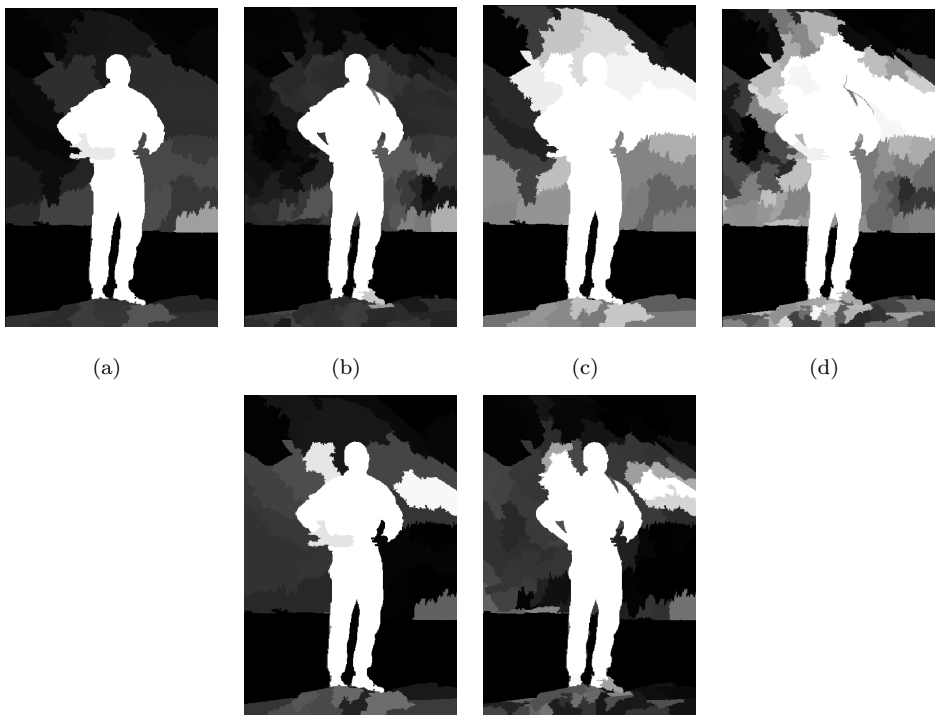
$$TorsoScore(S_{tli}) = \frac{P_{Foreground}(S_{tli})}{P_{Background}(S_{tli}) + \epsilon} \tag{6.5}$$

(a) (b) (c)

Figure 6.8: Masks used for torso localization.

In the end of the previous process, there are images showing the segments' potentials for each segmentation level and for each torso hypothesis, as seen in Figure 6.9. In order to select the torso mask that retrieves the most suitable distribution of potentials, we accumulate the results in each segmentation level for each torso mask. As seen in Figure 6.10 the final potential maps accentuate successfully the torso and arms. Our approach has the advantages of taking different perceptual groupings into account and being able to alleviate the need for an accurate torso mask by conjunctively measuring the foreground and background potentials. The fact that we use superpixels in the computations makes comparisons more meaningful, preserves strong boundaries and improves algorithmic efficiency. Results may be improved by adding more segmentation levels and masks at different sizes and locations, but at the cost of computational complexity. Here we show how even with rough approximations we can achieve accurate and robust results without imposing extreme computational strain.

The obvious step is to threshold the aggregated potential torso images in order to retrieve the upper body mask. In most cases, hands or arms' skin is not sampled enough during the torso searching process, especially in the cases where arms are outstretched. Thus, we use the skin masks estimated during the skin detection process, which are more accurate than they would be in the case they were retrieved during this process anyway, since they were calculated using specifically the face's skin color, in a colorspace more appropriate for skin and segments created at a finer level of segmentation. These segments are superimposed on the aggregated potential torso images and receive the highest potential (which is 1, since the potentials are normalized).



(a) (b) (c) (d)

(e) (f)

Figure 6.9: Segments with potential of belonging to torso, a-b) for segmentation level 1 and 2 and torso mask at $0\,^{\circ}$, c-d) for segmentation level 1 and 2 and torso mask at $30\,^{\circ}$, e-f) for segmentation level 1 and 2 and torso mask at $-30\,^{\circ}$.

(a) (b) (c)

Figure 6.10: Aggregation of torso potentials shown in Figure 6.9, for torso masks at $0\,^{\circ}$, $30\,^{\circ}$ and $-30\,^{\circ}$.

Instead of using a simple or even adaptive thresholding, we use a multiple level thresholding to recover the regions with strong potential according to the method described so far, but at same time

comply to the following criteria: 1) they form a region with size close to the expected torso size (actually bigger in order to allow some freedom for the case arms are outstretched) and 2) the outer perimeter of this region overlaps with sufficiently high gradients. The distance of the selected region at threshold t ($Region_t$) to the expected upper body size ($ExpUpperBodySize$) is calculated as:

$$ScoreSize = e^{\frac{-|Region_t - ExpUpperBodySize|}{ExpUpperBodySize}} \tag{6.6}$$

where $ExpUpperBodySize = 11 \times PL^2$. The score for the second criterion is calculated by averaging the gradient image ($GradientImage$) responses for the pixels that belong to the perimeter ($PRegion_t$) of $Region_t$:

$$ScoreGradient = \frac{1}{|PRegion_t|} \sum^{|PRegion_t|} GradientImage \cap PRegion_t \tag{6.7}$$

Thresholding starts with zero value and becomes increasingly stricter at small steps (0.02). In each thresholding level, the largest connected component is rated and the masks with $ScoreGradient >$ 0.05 and $ScoreSize > 0.6$ are accumulated to a refined potential image, as seen in Figure 6.11. Incorporation of this a priori knowledge to the thresholding process aids further the accentuation of the true upper body regions. Accumulation of surviving masks starts when $ScoreSize > 0.6$ and resulting masks after this point will keep getting closer monotonically to the expected region size. Accumulation ends when $ScoreSize$ drops bellow 0.6. The rationale behind this process is to both restrict and define the thresholding range and focus the interest to segments with high potential of forming the upper body segment. The aggregate mask ($AggregateMask$) can now be processed easier and produce more meaningful results. Specifically, we set a final threshold which allows only regions that have survived more than 20% of the accumulation process in the final mask for the upper body region. This process is performed for every initial torso hypothesis, so in the end there are three corresponding aggregate masks, out of which the one that overlaps the most with the initial torso mask and obtains the highest aggregation score is selected. Aggregation score shows how many times each pixel has appeared in the accumulation process, implicitly implying its potential of belonging to true upper body segment.

6.7.1 Refinement

In many cases the extracted upper body mask is very accurate and can be deemed as the final results. However, since are dealing with complex scenarios, we choose to add an extra refinement step to cope with probable segmentation errors and pixels that manage to survive the multiple thresholding process. One idea that we use here is to give the upper body mask as input to an interactive

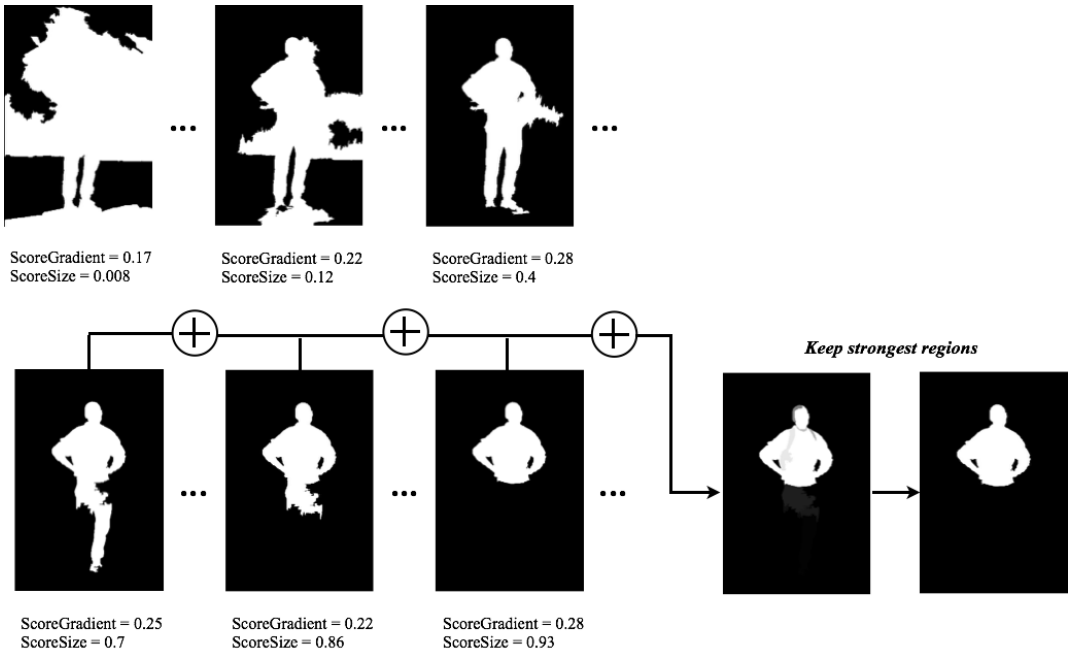
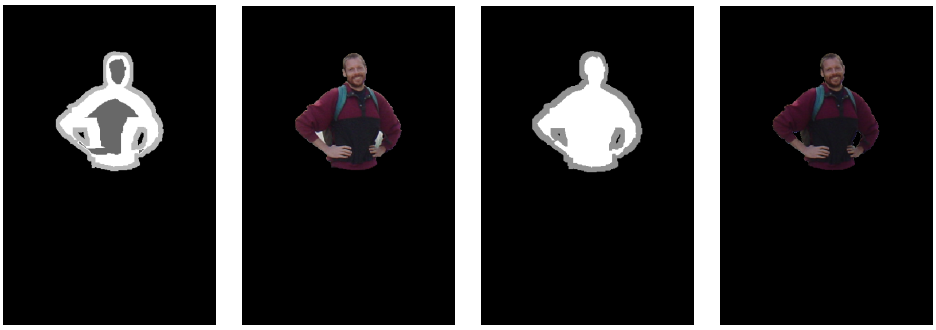


Figure 6.11: Thresholding of the aggregated potential torso images and final upper body mask. Note that the masks in the top row are discarded.

foreground/background algorithm that requires "seeds" denoting the foreground and background. GrowCut and GrabCut are used for experiments.

GrowCut expects the RGB image as input and a map denoting the seeds for background, foreground and uncertain pixels, whereas GrabCut can operate on a more refined map containing the certain foreground, certain background, probable foreground and probable background regions. In order to construct these maps we employ morphological operations on the upper body mask, with adaptive square structural elements (SE) using anthropometric constraints. For GrowCut, the uncertain region is constructed by dilating the upper body mask with a SE with sides equal to $PL/6$, the face's ellipse with a SE with sides equal to $PL/10$ and the skin regions with a SE with sides equal to $PL/12$. Possible holes between the face and torso region are also filled. The certain foreground is similarly constructed with erosions instead of dilations, where the sides of the SEs are now $PL/4$, $PL/4$ and $PL/10$, respectively. The rest of the map is classified as background. For the GrabCut algorithm the possible background ground is constructed by dilating the upper body mask, face's ellipse and skin masks using SEs with sides $PL/10$, $PL/2$ and $PL/12$, the probable foreground is constructed by eroding the masks with SEs with sides $PL/4$, $PL/4$ and $PL/10$, respectively and the certain foreground by eroding them with SEs with sides $PL/1$, $PL/3$ and $PL/8$, respectively. Both algorithms are guided with the extracted upper body mask, so their results are

very similar. Their main difference is that GrabCut can make better guesses in cases of uncertainty and segment large regions loosely defined by the map, whereas GrowCut is more sensitive to the map and influenced a lot by the background seeds. In Figure 6.12 for example, both algorithms extract the upper body successfully, with the difference that GrowCut removes the small enclosed regions by the arms, whereas GrabCut includes them.



(a) (b) (c) (d)

Figure 6.12: Example of foreground/background certainty maps and segmentations for a-b) GrabCut and c-d) GrowCut.

6.7.2 Hands Detection

Knowing the positions of hands can facilitate many higher-level methods, such as gesture and action recognition, human-computer interaction, etc., so we perform an additional step towards their localization and extraction. Previous work can be found in [227]. Hands here are skin regions connected to upper body that obey certain anthropometric constraints. First, our skin detection methodology manages to accurately extract skin patches. Patches that our outside the exterior anthropometric circle are automatically discarded. From the remaining ones, only the ones (almost) adjacent to the upper body have the potential of representing hands.

In cases where the person wears clothes with long sleeves the skin region of the hands is naturally segmented. In cases however where a larger portion of the arms is visible further examination is required. Here, we propose adding a skeletonization step for the skin regions, adopting the implementation from [344]. When the length of the medial axis is greater than 1 PL then the windows around its two extreme points are assumed to contain the hands region. When the length of the medial axis is greater than 1 PL then we rely on the anthropometric model to make an assumption about the most probable hand region. More specifically, the region between the outer and inner ellipses is more probable to represent a hand region. However, when both regions are

inside the inmost ellipse inference is ambiguous, so we choose to report both regions as potential hand regions. Exception is the case where the hand region is sufficiently surrounded by the upper body region. In this case this region is reported as representing a hand, without of course meaning that there not cases where this assumption is incorrect.

Figure 6.13: Example of hand skeletonization (lines are exaggerated for visibility). White line is the skeleton of the skin region and green Xs are the extreme points. Red region is considered as hand region because it is near the outer anthropometric ellipse.

Thus, the extreme points of the skeletonized skin regions with potential of containing hands provide good estimation of the hands locations. A small square window is then created around these points to further examine the corresponding regions. The length of the window was set to $2PL$ in our experiments so as to be large enough to contain the hand and small enough to restrain its region. Using this window we reexamine the skin probability in this window and exaggerate the pixels with high probability by applying a gamma adjustment, using a small gamma value (0.5 in our experiments). Using again the thresholds defined during the skin detection process we construct a trimap of certain, possible and impossible skin locations, depicted as white, gray and black pixels in Figure 6.14. This trimap is one of the inputs to the interactive GrowCut image segmentation method, as in the case of upper body region refinement step. The other input is the original RGB image of the window. Since in many cases varying illumination, skin like regions and blurriness caused by motion tend to distort the original image, a bilateral edge preserving filtering can be used to enhance the segmentation result.

(a) (b) (c)

Figure 6.14: Final result of our methodology, a) trimap seeds for GrowCut algorithm, b) image window encompassing the hand region, c) extracted hand regions (corresponding to the encircled face region)

6.8 Hands Detection Experimental Results

For our experiments images from the dataset in [308] are used. The dataset contains unconstrained frames from the TV show Buffy the Vampire Slayer and it is very challenging, because persons appear at a variety of scales, against highly cluttered background, and wear any kind of clothing. However, since visibility of the upper body and hands is one of our main assumptions, we select images that satisfy it.

(a) (b) (c)

(d) (e) (f)

Figure 6.15: Hands detection experimental results

First, the face verification step eliminates many of the false positive detections and unnecessary computations or unpredictable results are avoided. More importantly, our simple inference framework can perform well in cases of cluttered and skin-like background and focus on the true positive hand regions, even when in they slightly blurred hands because of motion effects. In the final step,

GrowCut algorithm, which is very sensitive to the provided seeds, can follow well the general outline of the hand but it is not guaranteed to capture accurately the whole hand area in the general case, as it might lose some fingers or include small skin-like patches adjacent or encompassed by the hand. Finally, Figure 6.15(b) shows a case where the detected hand could be classified as part of the arm.

A more extensive, qualitative experiment was conducted using the sign language pose recognition dataset collected by [312]. The dataset contains frames captured from signing sequences in news broadcasts, where the background to the left of the signer is constantly changing and unknown. This is a good example of an application our methodology could be used for. Since currently our methodology is restricted to cases where the face is in frontal view and the hands do not overlap, we selected 200 images that meet these criteria to perform the testing. Table I presents the results of this evaluation. In our case, true positives (tp) are considered the cases where the centroid of the detected hand regions lies within a distance of less than $1/2PL$, else they are considered false positives (fp). Finally, false negatives (fn) occur when missed hands, in other words cases where less than two hands are detected.

Precision	Recall	Tp	Fp	Fn
93.6%	94.2%	374	26	23

Table 6.1: Precision and recall for the sign language dataset

6.9 Lower Body Extraction

The algorithm for estimating the lower body part, in order to achieve full body segmentation is very similar to the one for upper body extraction. Its main difference is the anchor points that initiate the leg searching process. In the case of upper body segmentation it was the position of the face that aided the estimation of the upper body location. As expected, the upper body now aids the estimation of the lower body's position. More specifically, the general criterion we employ is that the upper parts of the legs should be underneath (and almost adjacent) the torso region. Although the previously estimated upper body region provides a solid starting point for the leg localization, different types of clothing like long coats, dresses or color similarities between the clothes of upper and lower body might make the torso region appear different (usually longer) than it should be. In order to better estimate the torso region, we perform a more refined torso fitting process, which does not require however extensive computations, since the already estimated shape provides a very good guide for the process.

The expected dimensions of the torso are again calculated based on anthropometric constraints, but aim in a more accurate model. Also, in order to cope with slight body deformations we allow the rectangle to be constructed according to a small parameter space. Specifically, we allow rotations w.r.t. rectangles's center by angle ϕ, translations in x and y axes, τ_x and τ_y and scaling in x and y axes, s_x and s_y. The initial dimensions of the rectangle correspond to the expected torso in full frontal and upright view and it is decreased during searching in order to accommodate other poses as well. The rationale behind the fitting score of each rectangle is measuring how much it covers the upper body region (UBR), since torso is the largest semantic region of the upper body, defined by potential UBC (Upper Body Coverage), while at the same time covering less of the background region, defined by potential S (for Solidity). Finally, in many cases the rectangle needs to be realigned with respect to the face's center ($FaceCenter$) to recover from misalignments caused by different poses and errors. A helpful criterion is the maximum distance of the rectangle's upper corners ($LShoulder$, $RShoulder$) from the face's center (D_{sf}), which should be constrained. Thus, fitting of the torso rectangle is formulated as a maximization problem:

$$\underset{\theta}{\text{maximize}} \;\; FittingScore(\theta) = \alpha_1 \times UBC(\theta) + \alpha_2 \times S(\theta) + \alpha_3 \times D_{sf}(\theta) \tag{6.8}$$

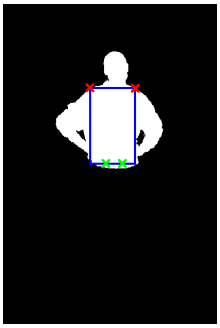


Figure 6.16: Best torso rectangle with shoulder and beginning of the legs positions.

where $\theta = (\phi, \tau_x, \tau_y, s_x, s_y)$,
$UBC(\theta) = \frac{\sum TorsoMask(\theta) \cap UBR}{\sum TorsoMask(\theta)}$,
$S(\theta) = \frac{\sum TorsoMask(\theta)}{\sum UBR}$,
$D_{sf}(\theta) =$
$e^{\frac{-|max(d(FaceCenter,RShoulder(\theta)),d(FaceCenter,LShoulder(\theta))) - 1.5 \times PL|}{1.5 \times PL}}$,
$TorsoMask(\theta)$ is the binary image where pixels inside the rectangle $r_{TorsoMask}(\theta)$ are 1, else 0,
UBR is the binary image where pixels inside the upper body region are 1, else 0,
and a_1, a_2, a_3 are weights, set to 0.4, 0.5 and 0.1, respectively.

After finding the torso rectangle with the best score, we estimate the shoulder positions (top corners of the rectangle) and more importantly the waist positions (lower corners of the rectangle). In turn, waist positions approximately indicate the beginning of the right and left leg, $leg_{BR} = (x, y)$ and $leg_{BL} = (x, y)$, respectively. These points are the middle points of the line segments of the waist points and the point in the center of the line that connects them. Similarly to upper body extraction and the torso rectangle fitting case, we explore hypotheses about the leg positions using rectangles by first creating rectangle mask for the upper leg parts, use them as samples for the pants color, perform appearance matching and evaluate the result. The assumption we make here is that there is uniformity in the color of the upper and lower parts of the pants. In the case of short pants were the lower leg parts are naked, the previously calculated skin regions are used to recover them. In order to reduce computational complexity, the size and position of the upper leg rectangles is fixed and adhering to anthropometric constraints and the only free parameter is their angle of rotation w.r.t. their center, ϕ_{right} and ϕ_{left}. Let $LegMask(\theta)$ be the binary masks for the two hypothesized leg parts, where $\theta = (\phi_{right}, \phi_{left})$.

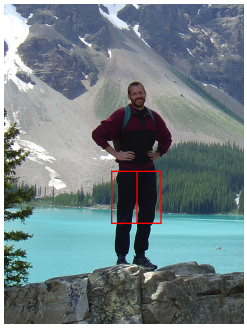


Figure 6.17: Example legs mask for $\phi_{right} = 0$ and $\phi_{left} = 0$.

Every possible upper leg mask is used as a sample of the pants regions and the leg regions are estimated using the clothes and skin detection process (Equations 6.1 - 6.5) described in the upper body extraction method. The hypothesized foreground are the pixels that belong to the leg mask and background is the rest of the image plus the pixels of the upper body mask, without the pixels below the waist line segment (if any). The leg mask retrieved from each hypothesis is the largest connected component of image segments with color similar to the hypothesis and the skin regions retrieved in the previous steps. We should note here that there is no strong need for precise alignment of the masks and the real leg parts, just enough coverage to perform a useful sampling. Thus, the algorithm can recover form slight torso misalignments and perform well for different leg positions without imposing the computational strain of dense searching using many mask parameters.

After the leg potentials are found, the same thresholding process as in the case of upper body takes

(a) (b) (c) (d)

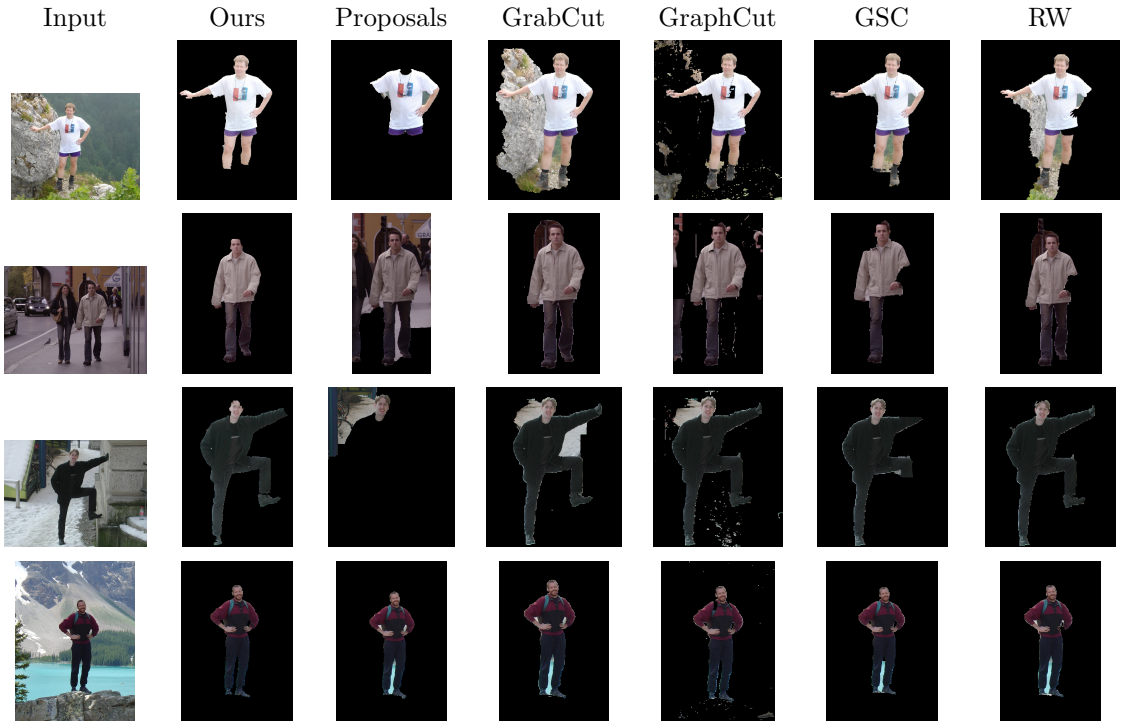
Figure 6.18: Example of foreground/background certainty maps and segmentations for a-b) GrabCut and c-d) GrowCut.

place, with the difference that now the expected body size is used in eq. 6.6, where $ExpUpperBodySize = 6 \times PL^2$. In order to construct the trimap of GrowCut to perform the refinement process for the leg regions the leg mask is eroded by a square structuring element (SE) with side $PL/4$ followed by dilation by a SE with side $PL/5$ in order to create the uncertainty mask and for the certain foreground mask it is eroded using a SE with side $PL/3$. Figure 6.18 shows an example.

6.10 Experimental Results

To evaluate our algorithm we used samples from the widely used and publicly available INRIA person dataset [345], which includes people performing everyday activities in outside environments in mostly upright position. This is a challenging dataset, since the photos are taken under various illumination conditions, in heavily cluttered environments and people appear in various types of clothing. For our experiment, we estimated the performance of our algorithm in segmenting 50 not occluded persons and compared the results with those of Proposals [346], GrabCut, the original version of GraphCut, Geodesic Star Convexity (GSC) and Random Walker (RW). Except for proposals method, which is automatic, the rest of the algorithms are interactive, so they are provided with the foreground and background seeds. For GraphCut, GSC and RW foreground seeds are in form of a straight line from the top of the head to the tip of the shoe, trying to pass through the most important regions of the body (hair, skin, different types of clothing, etc.), while background seeds are two lines almost enclosing the human body and covering the most important parts of the background along their path. For GrabCut foreground and background is estimated through a bounding box that tightly enclosed the human body. Table 6.2 shows some samples of the images used and the corresponding results of each algorithm.

Table 6.2: Sample results of the tested methodologies



The score $EvalScore$ of each silhouette R_a extracted by the algorithms are compared to ground truth silhouettes R_{gt} according to the formula:

$$EvalScore = \frac{\sum R_a \cap R_{gt}}{\sum R_a \cup R_{gt}} \tag{6.9}$$

where the $\cap$ and $\cup$ are the AND and OR operators, respectively. The results for each test are shown in Figure 6.19 and more compactly in Table 6.3, using the mean and standard deviation of the scores. As it can be seen from the samples, the segmentations our algorithm produces are accurate, with smooth boundaries most of the time and manage to preserve the skin regions, which are strongly correlated with body parts so preserving them should be a priority. Proposals algorithm recognizes salient objects from using segments produced by graph cuts, where the seeds are estimated by a hierarchical segmentation method. The algorithm is fully automatic, however in cases like the ones we are interested in the human body cannot be easily separated from the complex scenery, except for a few cases like in example 4 in Table 6.2. GrabCut produces very good results, which are also appealing to the eye and does not require a lot of human effort when the the bounding box version is used. On the other hand, in cases were the limbs are outstretched or enclose background regions, the bounding box may contain big background portions, which are treated as foreground and severely

harm the algorithm's capability to segregate them from the foreground. GraphCut performs very well in general but in many cases it produces many false positives that lower its scores, hence its high standard deviation value. As for example, in row 4 of Table 6.2 where the foreground color distribution is adequately different than that of the background, it is the only one that manages to discard the small regions enclosed by body limbs as opposed to the other algorithms, except for ours. However, in complex scenarios there are cases were the results are not natural. Geodesic Star Convexity (GSC) produces results comparable to GrabCut and in a few cases the best among the tested group. Experiments show that it usually has difficulties segmenting non-convex objects with complex color distributions, which is a usual case in humans due to their articulation. Finally, Random Walker (RW) algorithm's results are comparable to GrabCut's as well and again in a few cases better. RW's ability to guess and complete edges proves to be very powerful, but it often has the side effect of producing more rugged boundaries.

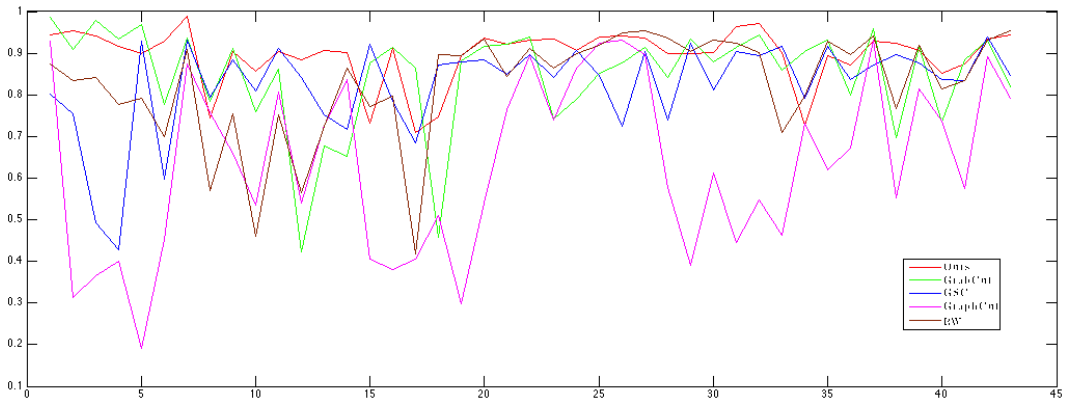


Figure 6.19: Evaluation results for INRIA dataset

Table 6.3: Compact evaluation results for INRIA dataset

Method	Ours	Proposals	GrabCut	GraphCut	GSC	RW
Mean	0.8953	0.4936	0.8476	0.6346	0.8250	0.8269
St. Dev.	0.0664	0.3219	0.1227	0.2067	0.1108	0.1284

Obviously, the first advantage of our methodology over the rest of the tested methodologies is that it can automatically localize and segment the object of interest, which in a case is the human body. Additionally, the final result achieves very good accuracy, even in complex scenarios and the small value of standard deviation shows that it is stable too. Qualitatively, the main advantages of our method are as follows. First, we combine efficiently cues from multiple levels

of segmentation. Although we chose to demonstrate the performance of the algorithm using only two levels of segmentation (and a third one for the skin detection), the algorithm can gather cues from segmentations from any different number of segmentation levels (course to fine). In [334], only one level of segmentation is used, whereas in [336], although multiple levels of a segmentation hierarchy are employed, the algorithm involves a part searching step over all produced segments, which is computationally expensive. Second, during our searching process with the torso and legs masks we try to find arbitrary salient regions, where by salient we mean that they are comprised by segments that appear strongly inside these regions and weakly outside. By considering foreground and background conjunctively, we alleviate the need for exact mask fitting and dense searching and we allow the masks to be large according to anthropometric constraints, so that they may perform sufficient sampling in fewer steps. In [340] foreground regions are sampled using small masks, which is not be sufficient to model the clothing in complex scenarios (non-uniform clothing, cluttered background, different poses). In [337] the human body is assumed to be inside a large mask, but due to the vast variability of human poses this assumptions often fails and the sampling may lead to unrecoverable errors. Third, we demonstrate how soft anthropometric constraints can guide and automate the process in many levels, from efficient mask creation and searching to the refinement of the probabilistic map that leads to the final mask for the body regions.

In addition to the INRIA dataset, where we demonstrate the robustness of the algorithm we performed a test to verify the algorithm's ability to segment bodies in more difficult poses. The experiment was conducted over 163 images from the "lab1" image set in [347] (we excluded images where face detection failed). In these images the foreground and background are simple and the main challenges from a color distribution perspective are caused due to background being similar to skin color. However, the actor performs various movements and we are interested in whether our simple mask hypotheses are able to cope with them.The mean evaluation score reached 97.68%, which indicates that the methodology can cope with various poses as well. Figure 6.20 shows examples of different poses from the dataset.

There are however limitations to the methodology. The most obvious one is that it requires the faces to be visible in profile or side views. We believe however that face is a vast source of information that should be at least be used when available. Another limitation is that hair and shoes are not accounted for explicitly and in many cases they are merged with the background if their appearance is more similar to it than the rest of the body. One way to alleviate this problem without making serious changes to the methodology would be to add masks for extreme parts of the limbs, the position of which can be found from the more stable regions, such as the torso and upper leg part. Such a solution however would impose additional computational complexity. Finally, poor

Figure 6.20: Examples from the dataset (the outline of the segmented body is superimposed to the images to conserve space).

segmentations may occur when one of the basic assumptions of our methodology, namely that there is sufficient discriminability between the foreground and background, is violated. Figure 6.21 shows examples of poor segmentation.

(a) (b)

Figure 6.21: Cases of poor segmentation.

7

Conclusions

7.1 Summary of the Dissertation

In this dissertation, we have dealt with one of the problems (human body detection and extraction) of monitoring and surveillance in a top-down manner. Firstly, we approached the general aspects of the field and moved to proposing solutions for this specific related problem, namely human body extraction from single images.

We began with the study of a multiprocessor system called DIAS to demonstrate the necessity for architectures designed for large-scale surveillance systems. The operation of DIAS' main processing unit was modeled using Stochastic Petri-Nets and an evaluation of simulated operations were used as proof of concept. Through a hierarchical organization of processing elements and parallelism, it was shown that the system can achieve high throughput and stability even when the computational demands are extreme.

Next, we moved into defining and presenting types of methodologies that are designed for monitoring and surveillance. We focused on research conducted in human activity recognition and presented an extensive survey of the literature, where we defined the problem and proposed a hierarchy that connects primitive action cues and complex events. We also classified the methodologies into meaningful general categories to show the current trends, achievements and challenges in the field. Finally, we proposed a maturity evaluation formula that enables a first-level assessment of key features of this type of methodologies, performed an evaluation to representative methodologies and discussed the results.

In the rest of the dissertation we proposed solutions to more specific problems. Since image segmentation is one of the most commonly used image processing techniques and one that permeates our algorithms, we studied it separately and proposed a blind evaluation metric for these algorithms. It differs from the rest of the proposed evaluation methods because it aims in providing a more

objective view of the results of segmentation algorithms by describing the results themselves, without comparing them to subjective golden truth data.

Finally, we reached the core of this work, where we proposed our methodologies for face detection and extracting human bodies from single images. In our face detection methodology, we use skin information to reduce the search space for face regions, employ image segmentation and corner detection to locate the most salient facial features and locate faces in frontal and profile views via graph matching.

In Chapter 6, the main contribution of the dissertation was unraveled. We presented a novel methodology for extracting human bodies from single images. It is a bottom-up approach that combines information from multiple levels of segmentation in order to discover salient regions with high potential of belonging to the human body. The main component of the system is the face detection step, where we estimate the rough location of the body, construct a rough antrhopometric model and model the skin's color. Soft anthropometric constraints guide an efficient searching for the most visible body parts, namely the upper and lower body, avoiding the need for strong prior knowledge, such as the pose of the body. Experiments on a challenging dataset showed that the algorithm can outperform even interactive segmentation algorithms and cope with various types of poses.

7.2 Summary of Contributions

This dissertation studies the problem monitoring and surveillance and focuses to human body extraction from images. It makes the following contributions to this goal:

- *Modeling of the IA unit of DIAS system.* We revisit the DIAS system previously presented in [219, 220], where the main components and flow of information in its IA unit are designed. The IA unit is the heart of operations in the DIAS system, designed for distributed and parallel execution of computer vision tasks in a multiprocessor framework. Here we model all possible flows of information among its components using a formal language and SPNs, in order to perform simulations that reveal the units potential in handling heavy load of data.

- *Survey on methodologies for human activity recognition.* A huge portion of monitoring and surveillance systems focus on observing and understanding human subjects. Here we define the problem and present an extensive review of the literature in this field, classified into meaningful categories according to the way they approach feature extraction and recognition. Our contributions include the proposal of a hierarchy that connects different semantic levels of activities

and behaviors and a first-level maturity evaluation aiming to act as a tool for concisely assessing the main aspects of related methodologies.

- *RLG-based image segmentation metric.* Since image segmentation is one of the prominent techniques in the field and also used extensively in our work, we develop a blind segmentation metric for evaluation of image segmentations in a more abstract and objective manner than the standard existing methods in literature.

- *Face detection in frontal and profile views.* Two of the strongest points of the proposed algorithm is its invariance to in-plane rotations and its potential to treat cases of profile views of faces. Both cases are challenging and the majority of the literature only deals with cases of frontal, upright views of faces.

- *Human body extraction from single images.* This is the key contribution of the dissertation. It is a bottom-up approach, where we fully exploit cues about the person's skin color, rough body location and size through face detection. We propose combining information from different levels of image segmentation and guide searching for the main body parts using color similarities and soft anthropometric constraints.

7.3 Limitations and Future Work

In this section we present the limitations of our work and some thoughts about how to overcome them in the future.

- Although we tried to make our simulation of DIAS' main processing unit with SPNs as realistic as possible, there is still need for more extensive experiments that include real-world operations and components. One idea would be to decompose a complex methodology (e.g. our human body extraction) into hierarchical modules executed by dedicated elements of the system. The next step is the connection of all of the architecture's components to perform complex tasks from the beginning (image acquisition) to end (high level interpretations) in order to expose its potential and shortcomings and study it in more depth.

- We believe that the proposed segmentation metric explores a new direction towards the evaluation of image segmentation algorithms, but it is still in a primal stage and there exists some ambiguity in the interpretation of the results. In the future we plan to combine this method with other metrics and find specific applications that can benefit from it.

- Our face detection algorithm can cope with many cases and more importantly with profile face views, which are not usually treated in literature. It relies however on simple feature extraction and graph matching, which are efficient but can lead to false positives/negatives in challenging scenarios. There are many proposals in this field that could enhance these methods. Finally, although there is some robustness in out of plane rotations, the methodology still operates mainly in the 2D space and relies on visibility of facial features and skin detection. In the future more attention will be given in more general and robust facial cues and skin should act more as a probabilistic indicator that strong decisive factor.

- Finally, in our methodology for segmenting the human body we make some assumptions about the human pose, which restrict it from being applicable to unusual poses and when occlusions are strong. Without necessarily having to resort to a prior pose estimation method, we could employ additional boundary information or probabilistic part detectors. This approach could also aid segmentation in cases where there are similarities between the background and foreground, especially along the boundaries of the body. Problems like missing extreme regions, such as hair, shoes and gloves can be solved by incorporations of more masks in search for these parts, but caution should be taken in keeping the computational complexity from rising excessively.

References

[1] Joshua Candamo, Matthew Shreve, Dmitry B. Goldgof, Deborah B. Sapper, and Rangachar Kasturi. Understanding transit scenes: a survey on human behavior-recognition algorithms. *Intelligent Transportation Systems, IEEE Transactions on*, 11(1):206–224, 2010.

[2] Teddy Ko. A survey on behavior analysis in video surveillance for homeland security applications. In *Applied Imagery Pattern Recognition Workshop, 2008. AIPR'08. 37th IEEE*, pages 1–8, 2008.

[3] Weiming Hu, Tieniu Tan, Liang Wang, and Steve Maybank. A survey on visual surveillance of object motion and behaviors. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 34(3):334–352, 2004.

[4] Thomas B. Moeslund, Adrian Hilton, and Volker Krger. A survey of advances in vision-based human motion capture and analysis. *Computer vision and image understanding*, 104(2):90–126, 2006.

[5] Pavan Turaga, Rama Chellappa, Venkatramana S. Subrahmanian, and Octavian Udrea. Machine recognition of human activities: A survey. *Circuits and Systems for Video Technology, IEEE Transactions on*, 18(11):1473–1488, 2008.

[6] Richard J. Radke. A survey of distributed computer vision algorithms. In *Handbook of Ambient Intelligence and Smart Environments*, pages 35–55. Springer, 2010.

[7] Murtaza Taj and Andrea Cavallaro. Distributed and decentralized multicamera tracking. *Signal Processing Magazine, IEEE*, 28(3):46–58, 2011.

[8] Bernhard Rinner and Wayne Wolf. An introduction to distributed smart cameras. *Proceedings of the IEEE*, 96(10):1565–1575, 2008.

[9] Sanja Fidler, Gregor Berginc, and Ales Leonardis. Hierarchical statistical learning of generic parts of object structure. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 182–189, 2006.

[10] David Barber. *Bayesian reasoning and machine learning*. Cambridge University Press, 2012.

[11] Christopher M. Bishop and Nasser M. Nasrabadi. *Pattern recognition and machine learning*, volume 1. springer New York, 2006.

[12] J. K. Aggarwal and Michael S. Ryoo. Human activity analysis: A review. *ACM Computing Surveys (CSUR)*, 43(3):16, 2011.

[13] Daniel Weinland, Remi Ronfard, and Edmond Boyer. A survey of vision-based methods for action representation, segmentation and recognition. *Computer Vision and Image Understanding*, 115(2):224–241, 2011.

[14] Ronald Poppe. A survey on vision-based human action recognition. *Image and vision computing*, 28(6):976–990, 2010.

[15] Sheng Yi and Hamid Krim. Capturing human activity by a curve. In *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pages 3561–3564, 2009.

[16] Pilar Callau Uson, Kaori Hagihara, Diego Ruiz, and Benot Macq. Towards a visual-hull based multi-agent surveillance system. In *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pages 3573–3576, 2009.

[17] Fabian Nater, Helmut Grabner, and Luc Van Gool. Exploiting simple hierarchies for unsupervised human behavior analysis. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2014–2021, 2010.

[18] Chen Wu and Hamid Aghajan. User-centric environment discovery with camera networks in smart homes. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 41(2):375–383, 2011.

[19] Andreas Zweng and Martin Kampel. Unexpected human behavior recognition in image sequences using multiple features. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 368–371, 2010.

[20] Kaiqi Huang, Dacheng Tao, Yuan Yuan, Xuelong Li, and Tieniu Tan. View-independent behavior analysis. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 39(4):1028–1035, 2009.

[21] Weilun Lao and Jungong Han. Automatic video-based human motion analyzer for consumer surveillance system. *Consumer Electronics, IEEE Transactions on*, 55(2):591–598, 2009.

[22] Francisco Martinez-Contreras, Carlos Orrite-Urunuela, Elias Herrero-Jaraba, Hossein Ragheb, and Sergio A. Velastin. Recognizing human actions using silhouette-based hmm. In *Advanced Video*

and Signal Based Surveillance, 2009. AVSS'09. Sixth IEEE International Conference on, pages 43–48, 2009.

[23] Chin-De Liu, Yi-Nung Chung, and Pau-Choo Chung. An interaction-embedded HMM framework for human behavior understanding: with nursing environments as examples. *Information Technology in Biomedicine, IEEE Transactions on*, 14(5):1236–1246, 2010.

[24] Md Atiqur Rahman Ahad, J. Tan, H. Kim, and S. Ishikawa. Action recognition by employing combined directional motion history and energy images. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pages 73–78, 2010.

[25] Pradeep Natarajan, Vivek Kumar Singh, and Ram Nevatia. Learning 3d action models from a few 2d videos for view invariant action recognition. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 20006–2013, 2010.

[26] Srikanth Cherla, Kaustubh Kulkarni, Amit Kale, and Viswanathan Ramasubramanian. Towards fast, view-invariant human action recognition. In *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW'08. IEEE Computer Society Conference on*, pages 1–8, 2008.

[27] Naohiko Suzuki, Kosuke Hirasawa, Kenichi Tanaka, Yoshinori Kobayashi, Yoichi Sato, and Yozo Fujino. Learning motion patterns and anomaly detection by human trajectory analysis. In *Systems, Man and Cybernetics, 2007. ISIC. IEEE International Conference on*, pages 498–503, 2007.

[28] Andreas Zweng, Sebastian Zambanini, and Martin Kampel. Introducing a statistical behavior model into camera-based fall detection. In *Advances in Visual Computing*, pages 163–172. Springer, 2010.

[29] Wonjun Kim, Jaeho Lee, Minjin Kim, Daeyoung Oh, and Changick Kim. Human action recognition using ordinal measure of accumulated motion. *EURASIP journal on Advances in Signal Processing*, 2010:2, 2010.

[30] Homa Foroughi, Baharak Shakeri Aski, and Hamidreza Pourreza. Intelligent video surveillance for monitoring fall detection of elderly in home environments. In *Computer and Information Technology, 2008. ICCIT 2008. 11th International Conference on*, pages 219–224, 2008.

[31] Chien-Liang Liu, Chia-Hoang Lee, and Ping-Min Lin. A fall detection system using< i> k</i>-nearest neighbor classifier. *Expert Systems with Applications*, 37(10):7174–7181, 2010.

[32] Lykele Hazelhoff and Jungong Han. Video-based fall detection in the home using principal component analysis. In *Advanced Concepts for Intelligent Vision Systems*, pages 298–309, 2008.

[33] Zhongna Zhou, Xi Chen, Yu-Chia Chung, Zhihai He, Tony X. Han, and James M. Keller. Activity analysis, summarization, and visualization for indoor human activity monitoring. *Circuits and Systems for Video Technology, IEEE Transactions on*, 18(11):1489–1498, 2008.

[34] Bi Song, Ahmed T. Kamal, Cristian Soto, Chong Ding, Jay A. Farrell, and Amit K. Roy-Chowdhury. Tracking and activity recognition through consensus in distributed camera networks. *Image Processing, IEEE Transactions on*, 19(10):2564–2579, 2010.

[35] Chen Wu, Amir Hossein Khalili, and Hamid Aghajan. Multiview activity recognition in smart homes with spatio-temporal features. In *Proceedings of the Fourth ACM/IEEE International Conference on Distributed Smart Cameras*, pages 142–149, 2010.

[36] Imran N. Junejo, Emilie Dexter, Ivan Laptev, and Patrick Prez. View-independent action recognition from temporal self-similarities. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(1):172–185, 2011.

[37] M. Hofmann and D. M. Gavrila. Multi-view 3D human pose estimation in complex environment. *International journal of computer vision*, 96(1):103–124, 2012.

[38] Duan-Yu Chen, Sheng-Wen Shih, and H.-YM Liao. Human action recognition using 2-d spatio-temporal templates. In *Multimedia and Expo, 2007 IEEE International Conference on*, pages 667–670, 2007.

[39] Gal Lavee, Latifur Khan, and Bhavani Thuraisingham. A framework for a video analysis tool for suspicious event detection. *Multimedia Tools and Applications*, 35(1):109–123, 2007.

[40] Hongying Meng, N. Pears, and C. Bailey. Recognizing human actions based on motion information and SVM. 2006.

[41] Nicolas Thome, Serge Miguet, and Sbastien Ambellouis. A real-time, multiview fall detection system: A LHMM-based approach. *Circuits and Systems for Video Technology, IEEE Transactions on*, 18(11):1522–1532, 2008.

[42] Tianyu Huang, Chongde Shi, and Fengxia Li. Discriminative random fields for behavior modeling. In *Computer Science and Information Engineering, 2009 WRI World Congress on*, volume 5, pages 17–21, 2009.

[43] Dietmar Bruckner, Jamal Kasbi, Rosemarie Velik, and Wolfgang Herzner. High-level hierarchical semantic processing framework for smart sensor networks. In *Human System Interactions, 2008 Conference on*, pages 668–673, 2008.

[44] Abhinav Gupta, Aniruddha Kembhavi, and Larry S. Davis. Observing human-object interactions: Using spatial and functional compatibility for recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(10):1775–1789, 2009.

[45] Zhang Zhang, Tieniu Tan, and Kaiqi Huang. An extended grammar system for learning and recognizing complex visual events. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(2):240–255, 2011.

[46] Athanasios Bamis, Jia Fang, and Andreas Savvides. Detecting interleaved sequences and groups in camera streams for human behavior sensing. In *Distributed Smart Cameras, 2009. ICDSC 2009. Third ACM/IEEE International Conference on*, pages 1–8, 2009.

[47] Sven Fleck, Roland Loy, Christian Vollrath, Florian Walter, and Wolfgang Straer. SmartClassySurv-a smart camera network for distributed tracking and activity recognition and its application to assisted living. In *Distributed Smart Cameras, 2007. ICDSC'07. First ACM/IEEE International Conference on*, pages 211–218, 2007.

[48] Shehzad Khalid. Motion-based behaviour learning, profiling and classification in the presence of anomalies. *Pattern Recognition*, 43(1):173–186, 2010.

[49] Jianguo Zhang and Shaogang Gong. Action categorization with modified hidden conditional random field. *Pattern Recognition*, 43(1):197–203, 2010.

[50] Matej Per\vse, Matej Kristan, Janez Per\vs, Ga\vsper Mu\vsi\vc, Goran Vu\vckovi\vc, and Stanislav Kova\vci\vc. Analysis of multi-agent activity using petri nets. *Pattern Recognition*, 43(4):1491–1501, 2010.

[51] Xinxiao Wu, Yunde Jia, and Wei Liang. Incremental discriminant-analysis of canonical correlations for action recognition. *Pattern Recognition*, 43(12):4190–4197, 2010.

[52] Antonios Oikonomopoulos, Maja Pantic, and Ioannis Patras. Sparse b-spline polynomial descriptors for human activity recognition. *Image and Vision Computing*, 27(12):1814–1825, 2009.

[53] Moshe Blank, Lena Gorelick, Eli Shechtman, Michal Irani, and Ronen Basri. Actions as space-time shapes. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 2, pages 1395–1402, 2005.

[54] Tao Xiang and Shaogang Gong. Beyond tracking: Modelling activity and understanding behaviour. *International Journal of Computer Vision*, 67(1):21–51, 2006.

[55] Silvio Savarese, Andrey DelPozo, Juan Carlos Niebles, and Li Fei-Fei. Spatial-temporal correlatons for unsupervised action classification. In *Motion and video Computing, 2008. WMVC 2008. IEEE Workshop on*, pages 1–8, 2008.

[56] Matteo Bregonzio, Shaogang Gong, and Tao Xiang. Recognising action as clouds of space-time interest points. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1948–1955, 2009.

[57] Hueihan Jhuang, Thomas Serre, Lior Wolf, and Tomaso Poggio. A biologically inspired system for action recognition. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8, 2007.

[58] Maria-Jose Escobar, Guillaume S. Masson, Thierry Vieville, and Pierre Kornprobst. Action recognition using a bio-inspired feedforward spiking network. *International Journal of Computer Vision*, 82(3):284–301, 2009.

[59] Nikolaos Bourbakis, Jim R. Gattiker, and George Bebis. A synergistic model for representing and interpreting human activities and events from video. *International Journal on Artificial Intelligence Tools*, 12(01):101–116, 2003.

[60] Yuxiao Hu, Liangliang Cao, Fengjun Lv, Shuicheng Yan, Yihong Gong, and Thomas S. Huang. Action detection in complex scenes with spatial and temporal ambiguities. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 128–135, 2009.

[61] Jingen Liu, Jiebo Luo, and Mubarak Shah. Recognizing realistic actions from videos in the wild. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1996–2003, 2009.

[62] Marcin Marszalek, Ivan Laptev, and Cordelia Schmid. Actions in context. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2929–2936, 2009.

[63] Ross Messing, Chris Pal, and Henry Kautz. Activity recognition using the velocity histories of tracked keypoints. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 104–111, 2009.

[64] Konstantinos Rapantzikos, Yannis Avrithis, and Stefanos Kollias. Dense saliency-based spatiotemporal feature points for action recognition. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1454–1461, 2009.

[65] Michael S. Ryoo and Jake K. Aggarwal. Semantic representation and recognition of continued and recursive human activities. *International journal of computer vision*, 82(1):1–24, 2009.

[66] Hae Jong Seo and Peyman Milanfar. Detection of human actions from a single example. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1965–1970, 2009.

[67] Ju Sun, Xiao Wu, Shuicheng Yan, Loong-Fah Cheong, Tat-Seng Chua, and Jintao Li. Hierarchical spatio-temporal context modeling for action recognition. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2004–2011, 2009.

[68] Pavan Turaga, Ashok Veeraraghavan, and Rama Chellappa. Unsupervised view and rate invariant clustering of video sequences. *Computer Vision and Image Understanding*, 113(3):353–371, 2009.

[69] Yang Wang, Hao Jiang, Mark S. Drew, Ze-Nian Li, and Greg Mori. Unsupervised discovery of action classes. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 1654–1661, 2006.

[70] Yang Wang and Greg Mori. Human action recognition by semilatent topic models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(10):1762–1774, 2009.

[71] William Brendel and Sinisa Todorovic. Learning spatiotemporal graphs of human activities. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 778–785, 2011.

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

[73] Dian Gong and Gerard Medioni. Dynamic manifold warping for view invariant action recognition. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 571–578, 2011.

[74] Qiang Qiu, Zhuolin Jiang, and Rama Chellappa. Sparse dictionary-based representation and recognition of action attributes. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 707–714, 2011.

[75] Shandong Wu, Omar Oreifej, and Mubarak Shah. Action recognition in videos acquired by a moving camera using motion decomposition of lagrangian particle trajectories. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1419–1426, 2011.

[76] Bhaskar Chakraborty, Michael B. Holte, Thomas B. Moeslund, and Jordi Gonzlez. Selective spatio-temporal interest points. *Computer Vision and Image Understanding*, 116(3):396–410, 2012.

[77] Gary R. Bradski. Computer vision face tracking for use in a perceptual user interface. 1998.

[78] Zoran Zivkovic and Ferdinand van der Heijden. Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern recognition letters*, 27(7):773–780, 2006.

[79] Jungong Han, Dirk Farin, and Weilun Lao. Real-time video content analysis tool for consumer media storage system. *Consumer Electronics, IEEE Transactions on*, 52(3):870–878, 2006.

[80] William J. Christmas. Filtering requirements for gradient-based optical flow measurement. *Image Processing, IEEE Transactions on*, 9(10):1817–1820, 2000.

[81] Md Ahad, Atiqur Rahman, J. K. Tan, H. S. Kim, and S. Ishikawa. Temporal motion recognition and segmentation approach. *International Journal of Imaging Systems and Technology*, 19(2):91–99, 2009.

[82] Ahmed Elgammal, David Harwood, and Larry Davis. Non-parametric model for background subtraction. In *Computer VisionECCV 2000*, pages 751–767. Springer, 2000.

[83] Chris Stauffer and W. Eric L. Grimson. Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 2, 1999.

[84] Amir Kale, Naresh Cuntoor, B. Yegnanarayana, A. N. Rajagopalan, and Rama Chellappa. Gait analysis for human identification. In *Audio-and Video-Based Biometric Person Authentication*, pages 706–714, 2003.

[85] Yanxi Liu, Robert Collins, and Yanghai Tsin. *Gait sequence analysis using frieze patterns*. Springer, 2002.

[86] Christopher Richard Wren, Ali Azarbayejani, Trevor Darrell, and Alex Paul Pentland. Pfinder: Real-time tracking of the human body. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(7):780–785, 1997.

[87] Erhan Baki Ermis, Venkatesh Saligrama, P.-M. Jodoin, and Janusz Konrad. Abnormal behavior detection and behavior matching for networked cameras. In *Distributed Smart Cameras, 2008. ICDSC 2008. Second ACM/IEEE International Conference on*, pages 1–10, 2008.

[88] Derek Anderson, Robert H. Luke, James M. Keller, Marjorie Skubic, Marilyn Rantz, and Myra Aud. Linguistic summarization of video for fall detection using voxel person and fuzzy logic. *Computer Vision and Image Understanding*, 113(1):80–89, 2009.

[89] Moein Shakeri, Hossein Deldari, Homa Foroughi, Alireza Saberi, and Aabed Naseri. A novel fuzzy background subtraction method based on cellular automata for urban traffic applications. In *Signal Processing, 2008. ICSP 2008. 9th International Conference on*, pages 899–902, 2008.

[90] Liang Wang. From blob metrics to posture classification to activity profiling. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, volume 4, pages 736–739, 2006.

[91] Reza Olfati-Saber. Distributed kalman filtering for sensor networks. In *Decision and Control, 2007 46th IEEE Conference on*, pages 5492–5498, 2007.

[92] Cristian Soto, Bi Song, and Amit K. Roy-Chowdhury. Distributed multi-target tracking in a self-configuring camera network. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1486–1493, 2009.

[93] Ashok Veeraraghavan, Amit K. Roy-Chowdhury, and Rama Chellappa. Matching shape sequences in video with applications in human movement analysis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(12):1896–1909, 2005.

[94] Sy Bor Wang, Ariadna Quattoni, L.-P. Morency, David Demirdjian, and Trevor Darrell. Hidden conditional random fields for gesture recognition. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 1521–1527, 2006.

[95] Tae-Kyun Kim, Josef Kittler, and Roberto Cipolla. Discriminative learning and recognition of image set classes using canonical correlations. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(6):1005–1018, 2007.

[96] Yang Wang, Payam Sabzmeydani, and Greg Mori. Semi-latent dirichlet allocation: A hierarchical model for human action recognition. In *Human Motion-Understanding, Modeling, Capture and Animation*, pages 240–254. Springer, 2007.

[97] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[98] John D. Lafferty and David M. Blei. Correlated topic models. In *Advances in neural information processing systems*, pages 147–154, 2005.

[99] Tian Lan, Yang Wang, and Greg Mori. Discriminative figure-centric models for joint action localization and recognition. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2003–2010, 2011.

[100] Chris Harris and Mike Stephens. A combined corner and edge detector. In *Alvey vision conference*, volume 15, page 50, 1988.

[101] Paul VC Hough. Machine analysis of bubble chamber pictures. In *International Conference on High Energy Accelerators and Instrumentation*, volume 73, 1959.

[102] David G. Lowe. Object recognition from local scale-invariant features. In *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, volume 2, pages 1150–1157, 1999.

[103] Piotr Dollr, Vincent Rabaud, Garrison Cottrell, and Serge Belongie. Behavior recognition via sparse spatio-temporal features. In *Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2005. 2nd Joint IEEE International Workshop on*, pages 65–72, 2005.

[104] I. Laptev and T. Lindeberg. Space-time interest points. In *Ninth IEEE International Conference on Computer Vision, 2003. Proceedings*, pages 432–439 vol.1, 2003.

[105] Douglas L. Vail, Manuela M. Veloso, and John D. Lafferty. Conditional random fields for activity recognition. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*, page 235, 2007.

[106] Geert Willems, Tinne Tuytelaars, and Luc Van Gool. An efficient dense and scale-invariant spatio-temporal interest point detector. In *Computer Vision-ECCV 2008*, pages 650–663. Springer, 2008.

[107] Shu-Fai Wong and Roberto Cipolla. Extracting spatiotemporal interest points using global information. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8, 2007.

[108] Alexander Klaser and Marcin Marszalek. A spatio-temporal descriptor based on 3D-gradients. 2008.

[109] Ivan Laptev and Tony Lindeberg. Local descriptors for spatio-temporal recognition. In *Spatial Coherence for Visual Motion Analysis*, pages 91–103. Springer, 2006.

[110] Ivan Laptev, Marcin Marszalek, Cordelia Schmid, and Benjamin Rozenfeld. Learning realistic human actions from movies. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8, 2008.

[111] Paul Scovanner, Saad Ali, and Mubarak Shah. A 3-dimensional sift descriptor and its application to action recognition. In *Proceedings of the 15th international conference on Multimedia*, pages 357–360, 2007.

[112] Li Fei-Fei and Pietro Perona. A bayesian hierarchical model for learning natural scene categories. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 524–531, 2005.

[113] Michael Hofmann and Dariu M. Gavrila. Single-frame 3D human pose recovery from multiple views. In *Pattern Recognition*, pages 71–80. Springer, 2009.

[114] Lihi Zelnik-Manor and Michal Irani. Event-based analysis of video. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 2, pages II–123, 2001.

[115] Michael J. Black and P. Anandan. A framework for the robust estimation of optical flow. In *Computer Vision, 1993. Proceedings., Fourth International Conference on*, pages 231–236, 1993.

[116] Antonios Oikonomopoulos, Ioannis Patras, and Maja Pantic. Spatiotemporal salient points for visual recognition of human actions. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 36(3):710–719, 2005.

[117] Michael E. Tipping. Sparse bayesian learning and the relevance vector machine. *The Journal of Machine Learning Research*, 1:211–244, 2001.

[118] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). *The annals of statistics*, 28(2):337–407, 2000.

[119] Jim R. Parker. *Algorithms for image processing and computer vision*. Wiley. com, 2010.

[120] Alireza Fathi and Greg Mori. Action recognition by learning mid-level motion features. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8, 2008.

[121] Payam Sabzmeydani and Greg Mori. Detecting pedestrians by learning shapelet features. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8, 2007.

[122] Bruce D. Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In *IJCAI*, volume 81, pages 674–679, 1981.

[123] Paul Viola and Michael Jones. Robust real-time object detection. *International Journal of Computer Vision*, 4, 2001.

[124] Oliver Amft. On the need for quality standards in activity recognition using ubiquitous sensors. In *How To Do Good Research In Activity Recognition. Workshop in conjunction with Pervasive*, 2010.

[125] Ulf Blanke, Diane Larlus, Kristof Van Laerhoven, and Bernt Schiele. Standing on the shoulders of other researchers a position statement. In *Pervasive workshophow to do good activity recognition research? Experimental methodologies, evaluation metrics and reproducibility issues*, 2010.

[126] A. Brush, John Krumm, and James Scott. Activity recognition research: The good, the bad, and the future. In *Pervasive 2010 Workshop*, 2010.

[127] Juan Ye, Lorcan Coyle, Susan McKeever, and Simon Dobson. Dealing with activities with diffuse boundaries. 2010.

[128] Daniel Roggen, Kilian Frster, Alberto Calatroni, Andreas Bulling, and Gerhard Trster. On the issue of variability in labels and sensor configurations in activity recognition systems.

[129] Benjamin Poppinga and Susanne Boll. Activity recognition research is more than finding the ultimate algorithms or parameters.

[130] Tim van Kasteren, Gwenn Englebienne, and Ben Krse. Towards a consistent methodology for evaluating activity recognition model performance. *Pervasive Computing*, 2010.

[131] Sumi Helal, Eunju Kim, and Shantonu Hossain. Scalable approaches to activity recognition research. In *8th International Conference Pervasive Workshop*, volume 234, 2010.

[132] Dawud Gordon, Hedda Schmidtke, and Michael Beigl. Introducing new sensors for activity recognition. In *How To Do Good Research In Activity Recognition: Experimental methodology, performance evaluation and reproducibility. Workshop in conjunction with Pervasive*, 2010.

[133] Timothee Cour, Chris Jordan, Eleni Miltsakaki, and Ben Taskar. Movie/script: Alignment and parsing of video and text transcription. In *Computer Vision-ECCV 2008*, pages 158–171. Springer, 2008.

[134] Mark Everingham, Josef Sivic, and Andrew Zisserman. Hello! my name is... buffy-automatic naming of characters in tv video. 2006.

[135] Christiane Fellbaum. *WordNet*. Springer, 2010.

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

[137] Krystian Mikolajczyk and Cordelia Schmid. Scale & affine invariant interest point detectors. *International journal of computer vision*, 60(1):63–86, 2004.

[138] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[139] Josef Sivic and Andrew Zisserman. Video google: A text retrieval approach to object matching in videos. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 1470–1477, 2003.

[140] Gabriella Csurka, Christopher Dance, Lixin Fan, Jutta Willamowski, and Cdric Bray. Visual categorization with bags of keypoints. In *Workshop on statistical learning in computer vision, ECCV*, volume 1, page 22, 2004.

[141] Alexander J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond*. MIT press, 2002.

[142] Jim Mutch and David G. Lowe. Multiclass object recognition with sparse, localized features. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 11–18, 2006.

[143] MarcAurelio Ranzato, Fu Jie Huang, Y.-L. Boureau, and Yann Lecun. Unsupervised learning of invariant feature hierarchies with applications to object recognition. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8, 2007.

[144] Thomas Serre, Lior Wolf, and Tomaso Poggio. Object recognition with features inspired by visual cortex. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 2, pages 994–1000, 2005.

[145] Donald H. Perkel and Theodore H. Bullock. Neural coding. *Neurosciences Research Program Bulletin*, 1968.

[146] Krystian Mikolajczyk, Tinne Tuytelaars, Cordelia Schmid, Andrew Zisserman, Jiri Matas, Frederik Schaffalitzky, Timor Kadir, and Luc Van Gool. A comparison of affine region detectors. *International journal of computer vision*, 65(1-2):43–72, 2005.

[147] Atsuhiro Kojima, Takeshi Tamura, and Kunio Fukunaga. Natural language description of human activities from video images based on concept hierarchy of actions. *International Journal of Computer Vision*, 50(2):171–184, 2002.

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

[149] Yuri A. Ivanov and Aaron F. Bobick. Recognition of visual activities and interactions by stochastic parsing. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(8):852–872, 2000.

[150] Daniel Weinland, Remi Ronfard, and Edmond Boyer. Free viewpoint action recognition using motion history volumes. *Computer Vision and Image Understanding*, 104(2):249–257, 2006.

[151] Pradeep Natarajan and Ramakant Nevatia. Online, real-time tracking and recognition of human actions. In *Motion and video Computing, 2008. WMVC 2008. IEEE Workshop on*, pages 1–8, 2008.

[152] Hae Jong Seo and Peyman Milanfar. Training-free, generic object detection using locally adaptive regression kernels. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9):1688–1704, 2010.

[153] Eli Shechtman and Michal Irani. Matching local self-similarities across images and videos. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8, 2007.

[154] Eli Shechtman and Michal Irani. Space-time behavior-based correlation-or-how to tell if two underlying motion fields are similar without computing them? *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(11):2045–2056, 2007.

[155] Hiroyuki Takeda, Sina Farsiu, and Peyman Milanfar. Kernel regression for image processing and reconstruction. *Image Processing, IEEE Transactions on*, 16(2):349–366, 2007.

[156] Shandong Wu, Brian E. Moore, and Mubarak Shah. Chaotic invariants of lagrangian particle trajectories for anomaly detection in crowded scenes. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2054–2060, 2010.

[157] Cosmin Grigorescu, Nicolai Petkov, and Michel A. Westenberg. Contour and boundary detection improved by surround suppression of texture edges. *Image and Vision Computing*, 22(8):609–622, 2004.

[158] Tony Lindeberg. Feature detection with automatic scale selection. *International journal of computer vision*, 30(2):79–116, 1998.

[159] Jan J. Koenderink and Andrea J. van Doorn. Representation of local geometry in the visual system. *Biological cybernetics*, 55(6):367–375, 1987.

[160] Atif Ilyas, Mihaela Scuturici, and Serge Miguet. Real time foreground-background segmentation using a modified codebook model. In *Advanced Video and Signal Based Surveillance, 2009. AVSS'09. Sixth IEEE International Conference on*, pages 454–459, 2009.

[161] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893, 2005.

[162] James W. Davis and Aaron F. Bobick. The representation and recognition of human movement using temporal templates. In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 928–934, 1997.

[163] Stuart Andrews, Ioannis Tsochantaridis, and Thomas Hofmann. Support vector machines for multiple-instance learning. In *Advances in neural information processing systems*, pages 561–568, 2002.

[164] Alexei A. Efros, Alexander C. Berg, Greg Mori, and Jitendra Malik. Recognizing action at a distance. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 726–733, 2003.

[165] Zhe Lin, Zhuolin Jiang, and Larry S. Davis. Recognizing actions by shape-motion prototype trees. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 444–451, 2009.

[166] Michal Aharon, Michael Elad, and Alfred Bruckstein. K-SVD: design of dictionaries for sparse representation. *Proceedings of SPARS*, 5:9–12, 2005.

[167] Teuvo Kohonen. Self-organization and associative memory. *Self-Organization and Associative Memory, 100 figs. XV, 312 pages.. Springer-Verlag Berlin Heidelberg New York. Also Springer Series in Information Sciences, volume 8*, 1, 1988.

[168] Dong Zhang, Daniel Gatica-Perez, Samy Bengio, and Iain McCowan. Modeling individual and group actions in meetings with layered HMMs. *Multimedia, IEEE Transactions on*, 8(3):509–520, 2006.

[169] Gale Young and Alston S. Householder. Discussion of a set of points in terms of their mutual distances. *Psychometrika*, 3(1):19–22, 1938.

[170] Gideon Schwarz. Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464, 1978.

[171] Imran N. Junejo, Emilie Dexter, Ivan Laptev, and Patrick Prez. *Cross-view action recognition from temporal self-similarities*. Springer, 2008.

[172] Juan Carlos Niebles and Li Fei-Fei. A hierarchical model of shape and appearance for human action classification. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8, 2007.

[173] Gunnar Johansson. Visual perception of biological motion and a model for its analysis. *Perception & psychophysics*, 14(2):201–211, 1973.

[174] Anant Madabhushi and J. K. Aggarwal. A bayesian approach to human activity recognition. In *Visual Surveillance, 1999. Second IEEE Workshop on,(VS'99)*, pages 25–32, 1999.

[175] Stan Birchfield. *KLT: An implementation of the Kanade-Lucas-Tomasi feature tracker*. 1998.

[176] Yan Ke and Rahul Sukthankar. PCA-SIFT: a more distinctive representation for local image descriptors. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–506, 2004.

[177] Juan Carlos Niebles, Hongcheng Wang, and Li Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words. *International Journal of Computer Vision*, 79(3):299–318, 2008.

[178] Silvio Savarese, John Winn, and Antonio Criminisi. Discriminative object class models of appearance and shape by correlatons. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 2033–2040, 2006.

[179] Pedro F. Felzenszwalb, Ross B. Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9):1627–1645, 2010.

[180] Yang Wang and Greg Mori. Hidden part models for human action recognition: Probabilistic versus max margin. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(7):1310–1323, 2011.

[181] Markus Weber, Max Welling, and Pietro Perona. *Unsupervised learning of models for recognition.* Springer, 2000.

[182] Vladimir Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(10):1568–1583, 2006.

[183] Michael S. Ryoo and Jake K. Aggarwal. Recognition of composite human activities through context-free grammar based representation. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 1709–1718, 2006.

[184] David Minnen, Irfan Essa, and Thad Starner. Expectation grammars: Leveraging high-level expectations for activity recognition. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 2, pages II–626, 2003.

[185] Alexandre RJ Francois, Ram Nevatia, Jerry Hobbs, Robert C. Bolles, and John R. Smith. VERL: an ontology framework for representing and annotating video events. *MultiMedia, IEEE*, 12(4):76–86, 2005.

[186] Neil Robertson and Ian Reid. A general method for human activity recognition in video. *Computer Vision and Image Understanding*, 104(2):232–248, 2006.

[187] Michael S. Ryoo and Jake K. Aggarwal. Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1593–1600, 2009.

[188] Wanqing Li, Zhengyou Zhang, and Zicheng Liu. Expandable data-driven graphical modeling of human actions based on salient postures. *Circuits and Systems for Video Technology, IEEE Transactions on*, 18(11):1499–1510, 2008.

[189] James F. Allen and George Ferguson. Actions and events in interval temporal logic. *Journal of logic and computation*, 4(5):531–579, 1994.

[190] Leslie Lamport. The temporal logic of actions. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 16(3):872–923, 1994.

[191] Camillo J. Taylor. Reconstruction of articulated objects from point correspondences in a single uncalibrated image. In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, volume 1, pages 677–684, 2000.

[192] Gal Lavee, Artyom Borzin, Ehud Rivlin, and Michael Rudzsky. Building petri nets from video event ontologies. In *Advances in Visual Computing*, pages 442–451. Springer, 2007.

[193] Nagia Ghanem, Daniel DeMenthon, David Doermann, and Larry Davis. Representation and recognition of events in surveillance video using petri nets. In *Computer Vision and Pattern Recognition Workshop, 2004. CVPRW'04. Conference on*, pages 112–112, 2004.

[194] M. S. Ryoo and J. K. Aggarwal. Recognition of high-level group activities based on activities of individual members. In *Motion and video Computing, 2008. WMVC 2008. IEEE Workshop on*, pages 1–8, 2008.

[195] Michael S. Ryoo and J. K. Aggarwal. Hierarchical recognition of human activities interacting with objects. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8, 2007.

[196] Michael S. Ryoo and J. K. Aggarwal. Semantic understanding of continued and recursive human activities. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, volume 1, pages 379–378, 2006.

[197] Henry A. Kautz. *A formal theory of plan recognition.* PhD thesis, Bell Laboratories, 1987.

[198] James Allen, Henry Kautz, Richard Pelavin, and Josh Tenenberg. *Reasoning about plans.* Morgan Kaufmann San Mateo, California, 1991.

[199] Wayne Wobcke. Two logical theories of plan recognition. *Journal of Logic and Computation*, 12(3):371–412, 2002.

[200] Bruno Bouchard, Sylvain Giroux, and Abdenour Bouzouane. A smart home agent for plan recognition of cognitively-impaired patients. *Journal of Computers*, 1(5):53–62, 2006.

[201] Juan Carlos Augusto and Chris D. Nugent. The use of temporal reasoning and management of complex events in smart homes. In *ECAI*, volume 16, page 778, 2004.

[202] Liming Chen, Chris Nugent, Maurice Mulvenna, Dewar Finlay, Xin Hong, and Michael Poland. Using event calculus for behaviour reasoning and assistance in a smart home. In *Smart homes and health telematics*, pages 81–89. Springer, 2008.

[203] Emmanuel Munguia Tapia, Tanzeem Choudhury, and Matthai Philipose. Building reliable activity models using hierarchical shrinkage and mined ontology. In *Pervasive Computing*, pages 17–32. Springer, 2006.

[204] Liming Chen, Chris Nugent, Maurice Mulvenna, Dewar Finlay, and Xin Hong. Semantic smart homes: towards knowledge rich assisted living environments. In *Intelligent Patient Management*, pages 279–296. Springer, 2009.

[205] Darnell Moore and Irfan Essa. Recognizing multitasked activities from video using stochastic context-free grammar. In *AAAI/IAAI*, pages 770–776, 2002.

[206] Seong-Wook Joo and Rama Chellappa. Recognition of multi-object events using attribute grammars. In *Image Processing, 2006 IEEE International Conference on*, pages 2897–2900, 2006.

[207] Massimiliano Albanese, Rama Chellappa, Vincenzo Moscato, Antonio Picariello, V. S. Subrahmanian, Pavan Turaga, and Octavian Udrea. A constrained probabilistic petri net framework for human activity detection in video. *Multimedia, IEEE Transactions on*, 10(6):982–996, 2008.

[208] Charles Castel, Laurent Chaudron, and Catherine Tessier. What is going on? a high level interpretation of sequences of images. In *ECCV Workshop on Conceptual Descriptions from Images*, pages 13–27, 1996.

[209] Tyler Gill, James M. Keller, Derek T. Anderson, and R. H. Luke. A system for change detection and human recognition in voxel space using the microsoft kinect sensor. In *Applied Imagery Pattern Recognition Workshop (AIPR), 2011 IEEE*, pages 1–8, 2011.

[210] Robert H. Luke. *A system for change detection and human recognition in voxel space using stereo vision*. PhD thesis, University of Missouri-Columbia, 2010.

[211] Wanqing Li, Zhengyou Zhang, and Zicheng Liu. Action recognition based on a bag of 3d points. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pages 9–14, 2010.

[212] Wanqing Li, Zhengyou Zhang, and Zicheng Liu. Graphical modeling and decoding of human actions. In *Multimedia Signal Processing, 2008 IEEE 10th Workshop on*, pages 175–180, 2008.

[213] Lu Xia, Chia-Chih Chen, and J. K. Aggarwal. View invariant human action recognition using histograms of 3D joints. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pages 20–27, 2012.

[214] Jamie Shotton, Toby Sharp, Alex Kipman, Andrew Fitzgibbon, Mark Finocchio, Andrew Blake, Mat Cook, and Richard Moore. Real-time human pose recognition in parts from single depth images. *Communications of the ACM*, 56(1):116–124, 2013.

[215] Jaeyong Sung, Colin Ponce, Bart Selman, and Ashutosh Saxena. Human activity detection from RGBD images. In *Plan, Activity, and Intent Recognition*, 2011.

[216] 3D sensors and natural interaction solutions.

[217] Andrew McCallum, Dayne Freitag, and Fernando CN Pereira. Maximum entropy markov models for information extraction and segmentation. In *ICML*, pages 591–598, 2000.

[218] Athanasios Tsitsoulis, Ryan Patrick, and Nikolaos Bourbakis. Surveillance Issues in a Smart Home Environment. In *2012 IEEE 24th International Conference on Tools with Artificial Intelligence*, volume 2, pages 18–25. IEEE, November 2012.

[219] Nikolas G. Bourbakis and David F. Thurston. Design of a hardware preprocessor for the DIAS-multilevel picture information system. *Journal of microcomputer applications*, 12(3):213–232, 1989.

[220] Nikolaos G. Bourbakis and David Thurston. Analytical modeling of the performance of a distributed multiprocessor system architecture. *Journal of microcomputer applications*, 15(4):283–295, 1992.

[221] Athanasios Tsitsoulis and Nikolaos Bourbakis. Modeling of Multi-Image/Video Flow on a Multiprocessor Surveillance System. *International Journal of Monitoring and Surveillance Technologies Research*, 1(1):1–15, 2013.

[222] Athanasios Tsitsoulis and Nikolaos Bourbakis. A first stage comparative survey on human activity recognition methodologies. *Special Issue on Visual Computing: 8th International Symposium(ISVC'12) IJAIT*, 22(6), December 2013.

[223] Athanasios Tsitsoulis and Nikolaos Bourbakis. An LG-graph-based early evaluation of segmented images. *Measurement Science and Technology*, 23(11):114007, 2012.

[224] N. Bourbakis and A. Tsitsoulis. A study for selecting a metric for a first level evaluation of image segmentation methods. In *Proceedings of the 2011 IEEE National Aerospace and Electronics Conference (NAECON)*, pages 276–279. IEEE, July 2011.

[225] Athanasios Tsitsoulis and Nikolaos Bourbakis. A methodology for detecting faces from different views. In *Proceedings of the 2012 IEEE 24th International Conference on Tools with Artificial Intelligence - Volume 01*, ICTAI '12, pages 238–245, Washington, DC, USA, 2012. IEEE Computer Society.

[226] Athanasios Tsitsoulis and Nikolaos Bourbakis. Automatic extraction of upper human body in single images. In *Information, Intelligence, Systems and Applications (IISA), 2013 Fourth International Conference on*, pages 1–8, 2013.

[227] Athanasios Tsitsoulis and Nikolaos Bourbakis. Towards automatic hands detection in single images. In *Image Analysis and Processing-ICIAP 2013*, pages 469–478. Springer, 2013.

[228] N. Bourbakis and A. Tsitsoulis. Monitoring and surveillance: Design of a formal language for representing body positions. In *Proceedings of the IEEE 2010 National Aerospace & Electronics Conference*, pages 266–268. IEEE, July 2010.

[229] Justin L. Tripp, Jan Frigo, and Paul Graham. A survey of multi-core coarse-grained reconfigurable arrays for embedded applications. *Proc. of HPEC*, 2007.

[230] M. Valera and S. A. Velastin. Intelligent distributed surveillance systems: a review. In *Vision, Image and Signal Processing, IEE Proceedings-*, volume 152, pages 192–204, 2005.

[231] Tilak Agerwala and Siddhartha Chatterjee. Computer architecture: Challenges and opportunities for the next decade. *Micro, IEEE*, 25(3):58–69, 2005.

[232] Luca Benini and Giovanni De Micheli. Networks on chips: A new SoC paradigm. *Computer*, 35(1):70–78, 2002.

[233] Gary Wang, Zoran Salcic, and Morteza Biglari-Abhari. Customizing multiprocessor implementation of an automated video surveillance system. *EURASIP Journal on Embedded Systems*, 2006(1):3–3, 2006.

[234] Robert I. Davis and Alan Burns. A survey of hard real-time scheduling for multiprocessor systems. *ACM Computing Surveys (CSUR)*, 43(4):35, 2011.

[235] Marco Cavadini, Matthias Wosnitza, and G. Troster. Multiprocessor system for high-resolution image correlation in real time. *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, 9(3):439–449, 2001.

[236] Peter M. Athanas and A. Lynn Abbott. Real-time image processing on a custom computing platform. *Computer*, 28(2):16–25, 1995.

[237] Sanjoy Baruah and Nathan Fisher. The feasibility analysis of multiprocessor real-time systems. In *Real-Time Systems, 2006. 18th Euromicro Conference on*, pages 10–pp, 2006.

[238] Steffen Klupsch, Markus Ernst, Sorin A. Huss, Integrierte Schaltungen und Systeme, M. Rumpf, and R. Strzodka. Real time image processing based on reconfigurable hardware acceleration. In *Workshop Heterogeneous Reconfigurable Systems on Chip (SoC)*, page 1, 2002.

[239] Nikolaos G. Bourbakis. Preface. *International Journal of Pattern Recognition and Artificial Intelligence*, 12(03):263–264, 1998.

[240] S. S. Maniccam and N. Bourbakis. Lossless compression and information hiding in images. *Pattern Recognition*, 37(3):475–486, 2004.

[241] Marco Ajmone Marsan, Gianni Conte, and Gianfranco Balbo. A class of generalized stochastic petri nets for the performance evaluation of multiprocessor systems. *ACM Transactions on Computer Systems (TOCS)*, 2(2):93–122, 1984.

[242] Mary K. Vernon and Mark A. Holliday. *Performance analysis of multiprocessor cache consistency protocols using generalized timed Petri nets*, volume 14. ACM, 1986.

[243] C. Murray Woodside and Yao Li. Performance petri net analysis of communications protocol software by delay-equivalent aggregation. In *Petri Nets and Performance Models, 1991. PNPM91., Proceedings of the Fourth International Workshop on*, pages 64–73, 1991.

[244] Theodosios Pavlidis. Structural pattern recognition. *Springer Series in Electrophysics, Berlin: Springer, 1977*, 1, 1977.

[245] Wladyslaw Skarbek, Andreas Koschan, and Zur Veroffentlichung. Colour image segmentation-a survey. 1994.

[246] Majid Mirmehdi and Maria Petrou. Segmentation of color textures. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(2):142–159, 2000.

[247] Robert M. Haralick and Linda G. Shapiro. Image segmentation techniques. *Computer vision, graphics, and image processing*, 29(1):100–132, 1985.

[248] Nikhil R. Pal and Sankar K. Pal. A review on image segmentation techniques. *Pattern recognition*, 26(9):1277–1294, 1993.

[249] Theodosios Pavlidis and Y.-T. Liow. Integrating region growing and edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 12(3):225–233, 1990.

[250] Jezching Ton, Jon Sticklen, and Anil K. Jain. Knowledge-based segmentation of landsat images. *Geoscience and Remote Sensing, IEEE Transactions on*, 29(2):222–232, 1991.

[251] Raimondo Schettini. A segmentation algorithm for color images. *Pattern Recognition Letters*, 14(6):499–506, 1993.

[252] Eli Saber, A. Murat Tekalp, and Gozde Bozdagi. Fusion of color and edge information for improved segmentation and edge linking. *Image and Vision Computing*, 15(10):769–780, 1997.

[253] Hui Zhang, Jason E. Fritts, and Sally A. Goldman. Image segmentation evaluation: A survey of unsupervised methods. *Computer Vision and Image Understanding*, 110(2):260–280, 2008.

[254] Ikram E. Abdou and William K. Pratt. Quantitative design and evaluation of enhancement/thresholding edge detectors. *Proceedings of the IEEE*, 67(5):753–763, 1979.

[255] Jianqing Liu and Yee-Hong Yang. Multiresolution color image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 16(7):689–700, 1994.

[256] M. Borsotti, Paola Campadelli, and Raimondo Schettini. Quantitative evaluation of color image segmentation results. *Pattern recognition letters*, 19(8):741–747, 1998.

[257] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 2, pages 416–423, 2001.

[258] Debi Prosad Dogra, Arun K. Majumdar, and Shamik Sural. Evaluation of segmentation techniques using region area and boundary matching information. *Journal of Visual Communication and Image Representation*, 23(1):150–160, 2012.

[259] Lutz Goldmann, Tomasz Adamek, Peter Vajda, Mustafa Karaman, Roland Mrzinger, Eric Galmar, Thomas Sikora, Noel E. OConnor, Thien Ha-Minh, and Touradj Ebrahimi. Towards fully automatic image segmentation evaluation. In *Advanced Concepts for Intelligent Vision Systems*, pages 566–577, 2008.

[260] Nikolaos G. Bourbakis. Emulating human visual perception for measuring difference in images using an SPN graph approach. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 32(2):191–201, 2002.

[261] M. A. Eshera and King-Sun Fu. An image understanding system using attributed symbolic representation and inexact graph-matching. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (5):604–618, 1986.

[262] A. Moghaddamzadeh and N. Bourbakis. A fuzzy-like region growing approach for segmentation of colored images, PR society j. *Pattern Recognition*, 30:6.

[263] Ming-Yu Liu, Oncel Tuzel, Srikumar Ramalingam, and Rama Chellappa. Entropy rate superpixel segmentation. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 2097–2104, 2011.

[264] Richard Nock and Frank Nielsen. Statistical region merging. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(11):1452–1458, 2004.

[265] Dorin Comaniciu and Peter Meer. Mean shift: A robust approach toward feature space analysis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(5):603–619, 2002.

[266] Peter Meer and Bogdan Georgescu. Edge detection with embedded confidence. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(12):1351–1365, 2001.

[267] Christopher M. Christoudias, Bogdan Georgescu, and Peter Meer. Synergism in low level vision. In *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, volume 4, pages 150–155, 2002.

[268] Rama Chellappa, Charles L. Wilson, and Saad Sirohey. Human and machine recognition of faces: A survey. *Proceedings of the IEEE*, 83(5):705–741, 1995.

[269] Erik Hjelm\aas and Boon Kee Low. Face detection: A survey. *Computer vision and image understanding*, 83(3):236–274, 2001.

[270] Wenyi Zhao, Rama Chellappa, P. Jonathon Phillips, and Azriel Rosenfeld. Face recognition: A literature survey. *Acm Computing Surveys (CSUR)*, 35(4):399–458, 2003.

[271] Ming-Hsuan Yang, David J. Kriegman, and Narendra Ahuja. Detecting faces in images: A survey. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(1):34–58, 2002.

[272] Guangzheng Yang and Thomas S. Huang. Human face detection in a complex background. *Pattern recognition*, 27(1):53–63, 1994.

[273] Thomas K. Leung, Michael C. Burl, and Pietro Perona. Finding faces in cluttered scenes using random labeled graph matching. In *Computer Vision, 1995. Proceedings., Fifth International Conference on*, pages 637–644, 1995.

[274] Kin Choong Yow and Roberto Cipolla. Feature-based human face detection. *Image and vision computing*, 15(9):713–735, 1997.

[275] Ying Dai and Yasuaki Nakano. Face-texture model based on SGLD and its application in face detection in a color scene. *Pattern recognition*, 29(6):1007–1017, 1996.

[276] Jie Yang and Alex Waibel. A real-time face tracker. In *Applications of Computer Vision, 1996. WACV'96., Proceedings 3rd IEEE Workshop on*, pages 142–147, 1996.

[277] Stephen J. McKenna, Shaogang Gong, and Yogesh Raja. Modelling facial colour and identity with gaussian mixtures. *Pattern recognition*, 31(12):1883–1892, 1998.

[278] Rick Kjeldsen and John Kender. Finding skin in color images. In *Automatic Face and Gesture Recognition, 1996., Proceedings of the Second International Conference on*, pages 312–317, 1996.

[279] Ian Craw, David Tock, and Alan Bennett. Finding face features. In *Computer VisionECCV'92*, pages 92–96, 1992.

[280] Andreas Lanitis, Christopher J. Taylor, and Timothy F. Cootes. Automatic face identification system using flexible appearance models. *Image and vision computing*, 13(5):393–401, 1995.

[281] Matthew Turk and Alex Pentland. Eigenfaces for recognition. *Journal of cognitive neuroscience*, 3(1):71–86, 1991.

[282] K.-K. Sung and Tomaso Poggio. Example-based learning for view-based human face detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(1):39–51, 1998.

[283] Henry A. Rowley, Shumeet Baluja, and Takeo Kanade. Neural network-based face detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(1):23–38, 1998.

[284] Edgar Osuna, Robert Freund, and Federico Girosit. Training support vector machines: an application to face detection. In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 130–136, 1997.

[285] Henry Schneiderman and Takeo Kanade. Probabilistic modeling of local appearance and spatial relationships for object recognition. In *Computer Vision and Pattern Recognition, 1998. Proceedings. 1998 IEEE Computer Society Conference on*, pages 45–51, 1998.

[286] Ambasamudram N. Rajagopalan, K. Sunil Kumar, Jayashree Karlekar, R. Manivasakan, M. Milind Patil, Uday B. Desai, P. G. Poonacha, and Subhasis Chaudhuri. Finding faces in photographs. In *Computer Vision, 1998. Sixth International Conference on*, pages 640–645, 1998.

[287] Michael S. Lew. Information theoretic view-based and modular face detection. In *Automatic Face and Gesture Recognition, 1996., Proceedings of the Second International Conference on*, pages 198–203, 1996.

[288] Antonio J. Colmenarez and Thomas S. Huang. Face detection with information-based maximum discrimination. In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 782–787, 1997.

[289] Paul Viola and Michael J. Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004.

[290] Praveen Kakumanu, Sokratis Makrogiannis, and Nikolaos Bourbakis. A survey of skin-color modeling and detection methods. *Pattern recognition*, 40(3):1106–1122, 2007.

[291] Qiang Zhu, Kwang-Ting Cheng, Ching-Tung Wu, and Yi-Leh Wu. Adaptive learning of an accurate skin-color model. In *Automatic Face and Gesture Recognition, 2004. Proceedings. Sixth IEEE International Conference on*, pages 37–42, 2004.

[292] Son Lam Phung, Abdesselam Bouzerdoum, and Douglas Chai. A novel skin color model in ycbcr color space and its application to human face detection. In *Image Processing. 2002. Proceedings. 2002 International Conference on*, volume 1, pages I–289, 2002.

[293] Hichem Sahbi and Nozha Boujemaa. Coarse to fine face detection based on skin color adaption. In *Biometric Authentication*, pages 112–120. Springer, 2006.

[294] Prem Kuchi, Prasad Gabbur, and P. SUBBANNA BHAT. Human face detection and tracking using skin color modeling and connected component operators. *IETE journal of research*, 48(3-4):289–293, 2002.

[295] P. Kakumanu, S. Makrogiannis, R. Bryll, Sethuraman Panchanathan, and N. Bourbakis. Image chromatic adaptation using ANNs for skin color adaptation. In *Tools with Artificial Intelligence, 2004. ICTAI 2004. 16th IEEE International Conference on*, pages 478–485, 2004.

[296] Vlad Constantin Cardei. *A neural network approach to colour constancy.* PhD thesis, Citeseer, 2000.

[297] ATRC image database. `http://www.cs.wright.edu/atrc/database.html`.

[298] Abbas Cheddad, Dzulkifli Mohamad, and Azizah Abd Manaf. Exploiting voronoi diagram properties in face segmentation and feature extraction. *Pattern recognition*, 41(12):3842–3859, 2008.

[299] X. Yuan, David Goldman, Ali Moghaddamzadeh, and N. Bourbakis. Segmentation of colour images with highlights and shadows sing fuzzy-like reasoning. *Pattern Analysis & Applications*, 4(4):272–282, 2001.

[300] Eli Saber and A. Murat Tekalp. Frontal-view face detection and facial feature extraction using color, shape and symmetry based cost functions. *Pattern Recognition Letters*, 19(8):669–680, 1998.

[301] Kwok-Wai Wong, Kin-Man Lam, and Wan-Chi Siu. An efficient algorithm for human face detection and facial feature extraction under different conditions. *Pattern Recognition*, 34(10):1993–2004, 2001.

[302] P. Jonathon Phillips, Harry Wechsler, Jeffery Huang, and Patrick J. Rauss. The FERET database and evaluation procedure for face-recognition algorithms. *Image and vision computing*, 16(5):295–306, 1998.

[303] P. Jonathon Phillips, Hyeonjoon Moon, Syed A. Rizvi, and Patrick J. Rauss. The FERET evaluation methodology for face-recognition algorithms. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(10):1090–1104, 2000.

[304] Tamara L. Berg, Alexander C. Berg, Jaety Edwards, and D. A. Forsyth. Whos in the picture. *Advances in neural information processing systems*, 17:137–144, 2005.

[305] Prag Sharma and Richard B. Reilly. A colour face image database for benchmarking of automatic face detection algorithms. In *Video/Image Processing and Multimedia Communications, 2003. 4th EURASIP Conference focused on*, volume 1, pages 423–428, 2003.

[306] Mykhaylo Andriluka, Stefan Roth, and Bernt Schiele. Pictorial structures revisited: People detection and articulated pose estimation. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1014–1021, 2009.

[307] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

[308] Vittorio Ferrari, Manual Marin-Jimenez, and Andrew Zisserman. Progressive search space reduction for human pose estimation. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8, 2008.

[309] M. Pawan Kumar, Andrew Zisserman, and Philip HS Torr. Efficient discriminative learning of parts-based models. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 552–559, 2009.

[310] Vincent Delaitre, Ivan Laptev, and Josef Sivic. Recognizing human actions in still images: a study of bag-of-features and part-based representations. In *BMVC*, volume 2, page 7, 2010.

[311] Bangpeng Yao and Li Fei-Fei. Grouplet: A structured image representation for recognizing human and object interactions. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 9–16, 2010.

[312] Patrick Buehler, Mark Everingham, Daniel P. Huttenlocher, and Andrew Zisserman. Long term arm and hand tracking for continuous sign language TV broadcasts. In *Proceedings of the 19th British Machine Vision Conference*, pages 1105–1114, 2008.

[313] Ali Farhadi and David Forsyth. Aligning ASL for statistical translation using a discriminative word model. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 2, pages 1471–1476, 2006.

[314] Liang Zhao and Larry S. Davis. Iterative figure-ground discrimination. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 1, pages 67–70, 2004.

[315] Yuri Y. Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in ND images. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 1, pages 105–112, 2001.

[316] Yuri Boykov and Vladimir Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(9):1124–1137, 2004.

[317] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. In *ACM Transactions on Graphics (TOG)*, volume 23, pages 309–314, 2004.

[318] Vladimir Vezhnevets and Vadim Konouchine. GrowCut: interactive multi-label ND image segmentation by cellular automata. In *proc. of Graphicon*, pages 150–156, 2005.

[319] Ajay Mishra, Yiannis Aloimonos, and Cheong Loong Fah. Active segmentation with fixation. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 468–475, 2009.

[320] Varun Gulshan, Carsten Rother, Antonio Criminisi, Andrew Blake, and Andrew Zisserman. Geodesic star convexity for interactive image segmentation. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 3129–3136, 2010.

[321] Olga Veksler. Star shape prior for graph-cut image segmentation. In *Computer Vision-ECCV 2008*, pages 454–467. Springer, 2008.

[322] Leo Grady. Random walks for image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(11):1768–1783, 2006.

[323] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Pictorial structures for object recognition. *International Journal of Computer Vision*, 61(1):55–79, 2005.

[324] Deva Ramanan. Learning to parse images of articulated bodies. In *Advances in neural information processing systems*, pages 1129–1136, 2006.

[325] Marcin Eichner and Vittorio Ferrari. Better appearance models for pictorial structures. 2009.

[326] M. Pawan Kumar, P. H. S. Ton, and Andrew Zisserman. Obj cut. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 18–25, 2005.

[327] Matthieu Bray, Pushmeet Kohli, and Philip HS Torr. Posecut: Simultaneous segmentation and 3d pose estimation of humans using dynamic graph-cuts. In *Computer Vision-ECCV 2006*, pages 642–655. Springer, 2006.

[328] John Winn and Nebojsa Jojic. Locus: Learning object classes with unsupervised segmentation. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 1, pages 756–763, 2005.

[329] John Winn and Jamie Shotton. The layout consistent random field for recognizing and segmenting partially occluded objects. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 37–44, 2006.

[330] Eran Borenstein and Jitendra Malik. Shape guided object segmentation. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 969–976, 2006.

[331] Liang Zhao and Larry S. Davis. Closely coupled object detection and segmentation. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 1, pages 454–461, 2005.

[332] Shifeng Li, Huchuan Lu, and Lei Zhang. Arbitrary body segmentation in static images. *Pattern Recognition*, 45(9):3402–3413, 2012.

[333] Lei Huang, Sheng Tang, Yongdong Zhang, Shiguo Lian, and Shouxun Lin. Robust human body segmentation based on part appearance and spatial constraint. *Neurocomputing*, 2013.

[334] Greg Mori, Xiaofeng Ren, Alexei A. Efros, and Jitendra Malik. Recovering human body configurations: Combining segmentation and recognition. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–326, 2004.

[335] Jitendra Malik, Serge Belongie, Thomas Leung, and Jianbo Shi. Contour and texture analysis for image segmentation. *International journal of computer vision*, 43(1):7–27, 2001.

[336] Yihang Bo and Charless C. Fowlkes. Shape-based pedestrian parsing. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 2265–2272, 2011.

[337] Yi Hu. Human body region extraction from photos. In *MVA*, pages 473–476, 2007.

[338] Zhilan Hu, Hong Yan, and Xinggang Lin. Clothing segmentation using foreground and background estimation based on the constrained delaunay triangulation. *Pattern Recognition*, 41(5):1581–1592, 2008.

[339] Zhilan Hu, Xinggang Lin, and Hong Yan. Torso detection in static images. In *Signal Processing, 2006 8th International Conference on*, volume 3, 2006.

[340] Meng Yao and Huchuan Lu. Human body segmentation in a static image with multiscale superpixels. In *Awareness Science and Technology (iCAST), 2011 3rd International Conference on*, pages 32–35, 2011.

[341] Zhilan Hu, Guijin Wang, Xinggang Lin, and Hong Yan. Recovery of upper body poses in static images based on joints detection. *Pattern Recognition Letters*, 30(5):503–512, 2009.

[342] Ciarn O. Conaire, Noel E. O'Connor, and Alan F. Smeaton. Detector adaptation by maximising agreement between independent data sources. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–6, 2007.

[343] Min C. Shin, Kyong I. Chang, and Leonid V. Tsap. Does colorspace transformation make any difference on skin detection? In *Applications of Computer Vision, 2002.(WACV 2002). Proceedings. Sixth IEEE Workshop on*, pages 275–279, 2002.

[344] Better skeletonization - file exchange - MATLAB central.

[345] INRIA person dataset. `http://pascal.inrialpes.fr/data/human/`.

[346] Ian Endres and Derek Hoiem. Category independent object proposals. In *Computer Vision-ECCV 2010*, pages 575–588. Springer, 2010.

[347] Hao Jiang. Human pose estimation using consistent max covering. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(9):1911–1918, 2011.