ACTION RECOGNITION IN VIDEO USING SPARSE CODING AND RELATIVE FEATURES

ANALÍ JESÚS ALFARO ALFARO

Thesis submitted to the Office of Research and Graduate Studies in partial fulfillment of the requirements for the degree of Master of Science in Engineering

Advisor:
DOMINGO MERY
ALVARO SOTO

Santiago de Chile, Marzo 2018

© MMVII, ANALÍ J. ALFARO ALFARO
ACTION RECOGNITION IN VIDEO USING SPARSE CODING AND RELATIVE FEATURES

ANALÍ JESÚS ALFARO ALFARO

Members of the Committee:
DOMINGO MERY
ALVARO SOTO
MIGUEL TORRES
PABLO ZEGERS
JORGE VÁSQUEZ

Thesis submitted to the Office of Research and Graduate Studies in partial fulfillment of the requirements for the degree of Master of Science in Engineering

Santiago de Chile, Marzo 2018

© MMVII, Analí J. Alfaro Alfaro
To the memory of my beloved father
and to my family
ACKNOWLEDGEMENTS

I would like to offer my thanks to all the people who contributed in the realization of this thesis and supported me during the last years.

I would like to thank my advisors Domingo Mery and Alvaro Soto, for their continuous guidance and understanding during the development of the thesis. They gave me support when it was needed. This thesis would have not been possible without their advise.

I want to thank all the people who filled my life in Chile with great moments. To my friends Teresa Bracamonte, Daniel Moreno, Violeta Chang, José Saavedra and Carlos Bedregal. Also, I want to thank to the DCC community for the great times in Chile: professors, administrative staff and colleagues whom I will always remember.

I want to dedicate this work to the memory of my beloved father Vicente Alfaro who inspire me to be a better version of myself always. Thank you to my mothers Soledad Alfaro, Jesús Cárdenas and my sister Diana Vargas, for their unconditional love and support.

Last but not least, a special acknowledgement goes for my loved family: my dear husband Iván Sipirán and my daughter Rafaela Sipirán. You are my strength and my happiness. I love you infinitely. Because, together with you, I discovered that life is beautiful.
# TABLE OF CONTENTS

ACKNOWLEDGEMENTS ......................................................... iv

LIST OF FIGURES ............................................................... vii

LIST OF TABLES ............................................................... viii

ABSTRACT ................................................................. ix

RESUMEN ................................................................. xi

1. INTRODUCTION ................................................................. 1
   1.1. Motivation ............................................................. 3
   1.2. Hypothesis ............................................................ 6
   1.3. Goal Descriptions ................................................. 6
      1.3.1. Main Goal ....................................................... 6
      1.3.2. Specific Goals ................................................ 7
   1.4. Previous Concepts ................................................ 7
   1.5. Summary of Contributions ...................................... 8
   1.6. Thesis Outline ..................................................... 9
   1.7. Thesis Publications ............................................... 10

2. LITERATURE REVIEW .......................................................... 12

3. CLASS-BASED VIDEO SUMMARIZATION ..................................... 17
   3.1. A Temporal Sliding Approach to find Key-Sequences .......... 17
      3.1.1. Describing the Video Key-Sequences ...................... 19
   3.2. A Class-based Sparse Coding Approach to find Key-Frames .... 20
      3.2.1. Optimization .................................................. 22
      3.2.2. Selection of Key-Sequences ................................ 22

4. RELATIVE FEATURES TO VIDEO DESCRIPTION .............................. 26
### LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Shows the central frame from key-sequences of lifting and diving actions to UCF-Sports database.</td>
<td>19</td>
</tr>
<tr>
<td>3.2</td>
<td>Process to extract low-level features from a key-sequence</td>
<td>20</td>
</tr>
<tr>
<td>3.3</td>
<td>Overview of the proposed method to extract key-sequences from an input video.</td>
<td>21</td>
</tr>
<tr>
<td>3.4</td>
<td>Shows the matrix of coefficients $W^i$</td>
<td>23</td>
</tr>
<tr>
<td>3.5</td>
<td>Key-frames selected by the proposed method for three different videos from Clean and Jerk action in the Olympic dataset.</td>
<td>24</td>
</tr>
<tr>
<td>3.6</td>
<td>Key-frames selected by the proposed method for three different videos from Discus throwing action in the Olympic dataset.</td>
<td>25</td>
</tr>
<tr>
<td>4.1</td>
<td>Overview of the method to obtain the ITRA descriptor</td>
<td>32</td>
</tr>
<tr>
<td>5.1</td>
<td>Shows samples from KTH dataset.</td>
<td>36</td>
</tr>
<tr>
<td>5.2</td>
<td>Presents the confusion matrix of our method on KTH dataset</td>
<td>36</td>
</tr>
<tr>
<td>5.3</td>
<td>Samples of Olympic dataset (Top) and the confusion matrix of our method for Olympic dataset (Bottom).</td>
<td>41</td>
</tr>
<tr>
<td>5.4</td>
<td>Samples of HOHA dataset (Top) and the confusion matrix of our method for HOHA dataset (Bottom).</td>
<td>42</td>
</tr>
</tbody>
</table>
LIST OF TABLES

5.1 Presents the thresholds used to filter out uninformative cuboids from the key-sequences .................................................. 34
5.2 Comparison of recognition performance using KTH dataset. ............. 35
5.3 Comparison of recognition performance using Olympic dataset. .......... 37
5.4 Comparison of recognition performance using HOHA dataset. .......... 38
5.5 Performances of our method and alternative strategies to extract key-sequences. 39
5.6 Performances of our method and alternative strategies to construct the video descriptor using sparse coding techniques. .......... 40
Currently humanity is producing and consuming a vast amount of video information. A key aspect to deal with such amount is to devise efficient and effective algorithms for storing, transmitting and analyzing this data. The video analysis content plays an important role to perform high level tasks such as human action recognition. The ability of determining people behavior or the form that people interact with other people or with objects is useful to create applications as as: video surveillance, human-computer interaction, video annotation, so forth. However, human action recognition is a hard problem due to occlusions, the variety of existing human poses, the length of a scene, dynamic backgrounds, etc.

In this thesis, we propose an approach that combines two steps in an effective method to classify human actions: i) the summarization of a video into atomic acts or key-sequences and ii) the search of reliable features to properly represent a human action. Our proposal address the previous steps giving a solution for each case.

First, we present a method for category-based video summarization using sparse coding. Our method is based in a new approach to tackle intra-class variations by decomposing each input video using a reduced set of brief temporal acts or key-sequences These key-sequences are selected using two criteria: i) the information of the video itself and ii) the information of the complete set of videos in the corresponding class. We achieve this by introducing a loss function that leads to the identification of a sparse set of representatives that simultaneously capture both, the relevant particularities in the input video and the generalities of videos in the entire class.

Second, we present a new video descriptor: ITRA (Inter-temporal Relational Act Descriptor). ITRA exploits the power of comparative reasoning by using sparse coding techniques to capture relative local similarity relations among atomic acts The relations demonstrates to be highly effective to differentiate between several action classes.
Our experiments show that the proposed approach reaches remarkable action recognition rates on several popular benchmarks, reducing the recognition error by up 24% with respect to state-of-the-art methods.

**Keywords:** action recognition, summarization, sparse coding, relative features.
RESUMEN

Actualmente la humanidad está constantemente produciendo y consumiendo una gran cantidad de información de vídeo. Un aspecto clave para tratar tremendos volúmenes de información es diseñar algoritmos eficientes y efectivos para almacenar, transmitir y analizar vídeos. Comprender el contenido de un vídeo es importante para el desarrollo de tareas de alto nivel como el reconocimiento de acciones. El reconocimiento de acciones humanas en vídeo permite determinar qué está haciendo una persona o cómo interactúa con otras personas o objetos, esto es útil para aplicaciones como por ejemplo: vídeo monitoreo, interacción hombre-computador, anotación de vídeo, etc. Sin embargo, el reconocimiento de acciones humanas es un problema difícil debido a la presencia de factores como: oclusión, diversidad de poses del cuerpo humano, duración de una escena, fondos dinámicos, etc.

En esta tesis, proponemos un enfoque que combina dos pasos en un método eficaz para clasificar acciones humanas: i) sintetizar un vídeo en actos atómicos o secuencias clave y ii) caracterizar adecuadamente la acción humana. Nuestra propuesta aborda los pasos previos presentando una solución para cada caso.

En primer lugar, presentamos un método para la síntesis de vídeo basado en usar la información que brinda la clase utilizando un método de codificación dispersa. Nuestro método enfrenta las variaciones presentes dentro de la clase mediante la descomposición de cada vídeo en un conjunto reducido de breves actos temporales o secuencias clave. Estas secuencias clave se seleccionan utilizando dos criterios: i) la información propia del vídeo y ii) la información del conjunto completo de vídeos en la clase correspondiente. Con este fin, proponemos una función de pérdida que permite identificar un conjunto de frames representantes que capturan simultáneamente las particularidades relevantes en el vídeo a analizar y las generalidades de los vídeos de su misma clase.

En segundo lugar, presentamos un nuevo descriptor de vídeo: ITRA (Descriptor Relacional de Actos Inter-Temporales), el cual explota el poder del razonamiento comparativo
mediante el uso de técnicas de codificación dispersa para capturar las relaciones de simil-
itud locales y relativas entre los actos atómicos que sintetizan al vídeo. Las relaciones
demuestran ser altamente efectivas para diferenciar varias clases de acciones.

Nuestros experimentos muestran que el enfoque propuesto alcanza notables tasas de
reconocimiento al ser evaluado en diversas bases de datos estándares, reduciendo el error
de reconocimiento hasta un 24 % respecto a otros métodos modernos.

**Palabras Claves:** reconocimiento de acciones, síntesis de vídeo, codificación esparsa,
características relativas.
1. INTRODUCTION

Nowadays, the video information is present in our daily life and it can be definitely useful to build interesting applications. However, since that huge amounts of video are produced we require to properly handle this kind of information. One of the high-level task consists of determining which actions are being performed by an actor in the video. Therefore, action recognition is crucial in the future development of video applications such as surveillance systems, human-computer interaction, video annotation and so on. The actions can also involve complex interactions between human and objects. On the other hand, an action class presents a higher intra-class variation since that each human have your own way to perform an action. Additionally, there are others aspects that add complexity as dynamic background, illumination, occlusions and so on. All these aspects make the action recognition a challenging problem.

In this work, we present a new approach which represent a video action as a set of meaningful acts. Afterwards, these acts are used to quantify relative local temporal similarities among the acts. These similarities form our core feature representation to future action recognition.

To achieve our goal, our approach is a built on two steps:

- Summarize the video into meaningful acts or key-sequences, that provides our core representation to capture relevant *intra-class* variations
- Extract relative temporal local features among action classes, that provides our core representation to capture discriminative *inter-class* relations.

In terms of the proposed technique to video summarization, several previous works have also built their action recognition schemes on top of key-frames (Zhao & Elgammal, 2008), Snippets (Schindler & Gool, 2008), Exemplars (Weinland & Boyer, 2008), Actoms (Gaidon, Harchaoui, & Schmid, 2011), or other informative subset of short video sub-sequences (Niebles, Chen, & Fei-Fei, 2010)(Raptis, Kokkinos, & Soatto, 2012). By representing a video using a compressed set of distinctive sub-sequences, it is possible to
eliminate spurious temporal patterns that can potentially impair classification, while still retaining enough information to recognize a target action (Assa, Caspi, & Cohen-Or, 2005). Furthermore, it is possible to obtain a normalized video representation that avoids distracting sources of intra-class variation, such as different velocities in the execution of an action. Here, we will refer to these sets of distinctive video sub-sequences as key-frames or key-sequences, if they are composed of one or of a few neighboring frames, respectively. Previous works have mainly defined a set of key-frames using manual labeling (Gaidon et al., 2011), unsupervised clustering and vector quantization techniques (Zhao & Elgammal, 2008), or discriminative approaches (Raptis et al., 2012). In the case of clustering and vector quantization techniques, the usual loss functions produce a set of key-frames that captures temporal action patterns occurring frequently in the target classes. As a drawback, training instances presenting less common patterns are usually poorly represented (Zhu, Anguelov, & Ramanan, 2014), therefore, the diversity of intra-class patterns is not fully captured. In the case of discriminative approaches, identification of relevant key-frames is usually connected to classification stages, focusing learning on mining patterns that capture relevant inter-class differences. Again, the mining of key-frames does not focus directly on effectively capturing the diversity of intra-class patterns that usually arise in complex action videos.

In this work we present a new technique to identify acts in an action video. In contrast to most previous work, this technique explicitly focuses on an effective mining of relevant intra-class variations. As a guiding strategy, the proposed technique selects from each training video a set of key-frames that balances two main objectives: (i) being informative about the input video, and (ii) being informative about the complete set of videos in an action class. We achieve this by relying on the machinery of sparse coding. Specifically, we establish a loss function that, for each video, leads to the identification of a sparse set of informative key-frames that simultaneously minimizes not only the reconstruction error of the input video, but also of all the remaining videos in the class collection. In other words, this loss function simultaneously favors the selection of relevant particularities arising in the input video, as well as the generalities arising in the entire class.
In terms of the proposed technique for extracting relative local temporal features, most visual recognition approaches base their feature coding strategy on quantifying the absolute presence or absence of a set of visual features. As a relevant alternative, recent works have shown that the relative strength (Yagnik, Strelow, Ross, & Lin, 2011) or the similarity among visual features (Parikh & Grauman, 2011) can be powerful cues to perform visual recognition. As an example, on PASCAL VOC 2010, the work in (Yagnik et al., 2011) demonstrates a notably increase in object recognition performance by using the relative ordering, or rank, among the feature dimensions, instead of their absolute values. Similarly, the work in (Kumar, Berg, Belhumeur, & Nayar, 2009) shows that a feature coding strategy based on similarities among pairs of attributes (similes), leads to state-of-the-art performance in a highly unstructured face recognition task.

1.1. Motivation

The first studies about the human motion with an artistic purpose emerge in 15th century. The artists showed interest in understanding the motion of muscles, bones and nerves as described by the own Leonardo da Vinci. At the end of 16th century, Giovanni Borelli introduced the biomechanics which study the motion from a mechanic and mathematical point of view.

In the middle of 1800’s approximately, the chronophotography started to be used to combine several frames of movement into a single image. The pioneer of this type of photography was the French scientist tienne-Jules Marey who came up with it to describe photographs of motion, further used for the purpose of studying motion in science. At the same time, the photoghapher Eadweard Muybridge was intrigued by the motion. Muybridge made a record of images to describe the locomotion of human and animals and also he invented a machine to display those images in a continuous way. From 1900 other studies have been developed about the motion perception oriented to understand the human behavior.
Thomas Edison invented the phonograph in 1877, where he also wanted to provide a visual accompaniment to the phonograph. Then he commissioned to William Dickson to create a motion-picture device. Based in the Marey and Muybridge foundations, William Dickson invented the first motion picture camera, the Kinetograph, in 1888. This device combined the two final essentials of motion-picture recording and viewing technology. Later, the Edison device that combined sound and image was known as Kinetoscope. The Kinetoscope exhibition in Paris inspired to Auguste and Louis Lumiere brothers, to invent the cinématographe in 1895.

For 1939 the first portable mini-camera appeared. A few years later in 1942, german scientists invented the Closed Circuit Television (CCTV). In 1951, the Video Tape Recorder (VTR) system was created to record video using a magnetic tape. This VTR system was used to record images in live from a television camera. After, VTR was coupled to CCTV to produce gray-scale video which was recorde to see it off-line. However, it was in 1990 when digital multiplexers revolutionized the surveillance industry by enabling recording on several cameras at once. Digital multiplex also added features like time-lapse and motion-only recording, which saved a great deal of wasted videotape. The IP cameras emerged in 1996 to show the future because it was possible to send and receive information from computers. With the time the Webcam displaced the CCTV system. From this time to the date several discoveries have contributed to the development of advanced cameras and with their arrival the creation of interesting applications oriented to the human action analysis.

The human action analysis has as objective the detection and recognition of human actions from videos so that the computer is able to understand human behaviors and take decisions. Understanding the human actions is a key aspect for the development of applications as surveillance systems, human-computer interaction, video annotation and so on. Therefore, it is important to know which actions are occurring in a video. However, what is an action? We define a human action as a conscious and coordinate body movement that a person develop with a purpose. An action is composed of a set of atomic movements or acts developed with the limbs. The deformable nature of the human body makes possible to perform simple and complex movements. However, this can be a problem when we need
to obtain a 2D representation as an image. For instance, the self-occlusion of the limbs, dynamic backgrounds that can be confused with limbs or the interact with other persons or objects which can cause confusion.

Human action recognition is a complex problem due to several aspects. We identified and grouped these aspects in two categories:

- The high intra-class variation in terms of variable video content, video quality and video length because they coming from different resources as internet, personal cameras, TV and so on.
- The deformable nature of the human body in terms of poses, self-occlusion, form and height of the human body.
- The effective representation of the information in terms of discriminative features to describe an action.

There are methods in the state-of-the-art that address the action recognition problem from different points of view.

For instance, some methods focus in the video decomposition in key-frames as a relevant step to the recognition. A group of them manually select the key-frames based on (Carlsson & Sullivan, 2001) poses, (Weinland & Boyer, 2008) exemplars, or key-sequences as (Schindler & Gool, 2008) snippets, (Gaidon et al., 2011) actoms. These methods requires manual selection and time to labeling. Other group of methods propose a discriminative model to select key-frames as (Zhao & Elgammal, 2008), (L. Liu, Shao, & Rockett, 2013), (Raptis & Sigal, 2013). However, the majority select the key-frames individually without taking in account information about the remaining frames.

On the other hand, other methods focus in the video representation as the main step to a successful recognition. A group of methods use low-level visual features as description (Scovanner, Ali, & Shah, 2007) and (Kläser, Marszalek, & Schmid, 2008), which are inspired in popular descriptors as SIFT (Lowe, 2004), HOG (Dalal & Triggs, 2005), respectively. But these methods have problems to recognize similar actions. As an alternative are the methods that use visual descriptors based on higher levels of abstraction which take
in account semantic attributes (J. Liu, Kuipers, & Savarese, 2011), (Sadanand & Corso, 2012).

We believe that is possible integrated both steps in a sucessful methodology to human action recognition. A video can be summarize in relevant sequences which can be described as well as their relationships.

1.2. Hypothesis

The video information presents limitations to face the human action recognition problems as poses, self-occlusion due the deformable nature of the human body. Also, there are a high intra-class variation in videos belonging to the same class, these variations corresponding with actors, ilumination, scale, scenarios and so on. The intra-class variation is because videos came from different sources for instance internet or tv. Therefore, an effective video representation is necessary to achieve discriminative features that guarantee a successful recognition.

we believe that it is possible to achieve an effective human action recognition from video information using an act-based representation which summarize the video and characterize these acts and their temporal relationships through relative features. The relative features are obtained by exploting the comparative reasoning using sparse coding to spatio-temporal levels.

1.3. Goal Descriptions

Now we describe our main goal and the specific goals that support our research. These goals must be fulfilled in order to verify our hypothesis.

1.3.1. Main Goal

Our main goal is the development of an approach to make an effective human action recognition from videos using an act-based representation which summarize the video and relative features to characterize that acts and their spatio-temporal relationships.
1.3.2. Specific Goals

To achieve our main goal, the follow specific goals must be fulfilled:

- Summarize a input video in a set of relevant sequences or atomic acts which must provide a suitable video representation and the remaining videos in the class.
- Describe the video acts and their temporal relationships using relative features.
- Recognize effectively the human action that is occurring in a video.

1.4. Previous Concepts

Lately, sparse coding techniques have received wide attention in the computer vision community, mainly due to their attractive properties to generate informative feature representations (Wright, Yang, Ganesh, Sastry, & Ma, 2009; Yang, Yu, Gong, & Huang, 2009; Elhamifar, Sapiro, & Vidal, 2012). The goal of sparse coding is to model the data as a sparse linear combination of a set of overcomplete basis vectors. Each of these basis is denominated an atom and, jointly, they form a dictionary to represent the data.

Let \( \mathbf{Y} = [\mathbf{y}_1 \ \mathbf{y}_2 \ \cdots \ \mathbf{y}_n] \in \mathbb{R}^{m \times n} \) be a matrix containing \( n \) input signals of \( m \) entries in its columns; \( \mathbf{D} = [\mathbf{d}_1 \ \mathbf{d}_2 \ \cdots \ \mathbf{d}_k] \in \mathbb{R}^{m \times k} \) be an overcomplete dictionary matrix that contains \( k \) prototype signals or atoms in its columns; and \( \mathbf{X} = [\mathbf{x}_1 \ \mathbf{x}_2 \ \cdots \ \mathbf{x}_n] \in \mathbb{R}^{k \times n} \) be the set of coefficients that encode the representation of signals in \( \mathbf{Y} \) using dictionary \( \mathbf{D} \). It is possible to obtain a dictionary \( \mathbf{D} \) and a set of coefficients \( \mathbf{X} \) by minimizing the objective function:

\[
\sum_{i=1}^{n} \| \mathbf{y}_i - \mathbf{D} \mathbf{x}_i \|_2^2 = \| \mathbf{Y} - \mathbf{D} \mathbf{X} \|_F^2, \tag{1.1}
\]

where \( \| \cdot \|_F^2 \) corresponds to the squared Frobenius norm. There are several options to face the previous dictionary learning problem. A popular approach is to choose the dictionary from a known set of transforms (e.g., Curvelet, Wavelet, etc.). An alternative approach is to learn a dictionary that has a good fit to the data. Among the possible solutions, a dictionary that closely models the data using sparse expansions, provides an informative and parsimonious data representation (Aharon, Elad, & Bruckstein, 2006; Donoho & Elad,
A straightforward measure of sparsity is the number of nonzero entries in $X$. It is possible to introduce this measure by using the pseudo-norm $\ell_0$:

$$D^*, X^* = \arg\min_{D, X} \|Y - DX\|^2_F \text{ s.t. } \|X_i\|_0 < \tau, \forall i,$$  \hspace{1cm} (1.2)

where $\|X_i\|_0$ is the $\ell_0$ norm, counting the number of nonzero entries in $x_i$, and $\tau \ll k$. Unfortunately, in the case of an underdetermined system of linear equations, using the pseudo-norm $\ell_0$ to find the sparsest solution is a NP-hard problem. While it is possible to obtain approximate solutions (Aharon et al., 2006), it has been shown (Donoho & Elad, 2003) that, under mild assumptions, the sparsest representation can also be obtained by solving a convex problem. Specifically, an equivalent solution to the optimization problem in Eq. (1.2) can be obtained by replacing the $\ell_0$ pseudo-norm by the $\ell_1$ norm, leading to the following convex optimization problem:

$$D^*, X^* = \arg\min_{D, X} \|Y - DX\|^2_F \text{ s.t. } \|X_i\|_1 < \tau, \forall i.$$  \hspace{1cm} (1.3)

This convex problem can be solved in polynomial time using linear programming methods (Donoho & Elad, 2003).

1.5. Summary of Contributions

In this work, we contribute to the increasing literature supporting the power of comparative reasoning (Yagnik et al., 2011). We achieve this by presenting a new method that encodes visual features representing local temporal similarities among action categories in video. As a building block, we use our proposed technique to identify key-frames. To incorporate motion information, these key-frames are augmented by neighboring frames to form key-sequences or brief local acts. These acts, in conjunction with sparse coding techniques, are then used to learn local temporal class-dependent dictionaries of acts. As a key observation, cross-projections of acts into dictionaries coming from different classes allow us to quantify local temporal similarities among the action categories. As we demonstrate, these similarities, or relative local temporal features, prove to be highly discriminative to perform action recognition in video.
We validate our ideas with a comprehensive experimentation. Our results shows an effectively mining intra and inter-class relations, which allow us obtain a remarkable action recognition performance on several popular benchmark datasets, reducing error by up to 24% with respect to previous published results.

This work makes the following three contributions:

- We propose a new method for identification of key-frames in video that manages intra-class variations by preserving essential temporal intra-class patterns, and filtering out irrelevant and redundant frames.
- A new method for extraction of relative features that manages inter-class variations by quantifying relative local temporal similarities among local acts.
- Empirical evidence indicating that the combination of the two previous ideas provides a substantial increase in recognition performance with respect to alternative state-of-the-art techniques.

1.6. Thesis Outline

This thesis is organized as follows:

- Chapter 2 presents the related work of our research. Because the state-of-the-art in action recognition is extensive, the literature is divided taking into account the major topics of this work: video summarization into key-sequences, video descriptors and solutions to action recognition problem based on sparse coding.
- Chapter 3 presents two approaches to take advantage of the class information available to summarize a video into a set of relevant frames. The first method addresses the summarization as a clustering process which use clips or small video sequences from an action class. The second method formulates the video summarization as a reconstruction problem using information from the class of the videos.
Chapter 4 introduces our video description proposal. Our method obtains a video description based on quantifying relative intra and inter-class similarities among local temporal patterns or key-sequences. We name our descriptor *ITRA* (Relative Local Temporal Features).

Chapter 5 shows the experiments and results of our work. The experiments are tested in three popular benchmark for action recognition: KTH, Olympic and HOHA database. Our approach achieves a remarkable action recognition performance compared with state-of-the-art approaches. Also, we evaluate the relevance of our approach to select key-frames and the power of discrimination of ITRA descriptor.

Chapter 6 draws the conclusions of our research work and delineates future research directions.

1.7. Thesis Publications

The ideas presented in this thesis have been published in the following papers:

**Conference Proceedings**

- (Alfaro, Mery, & Soto, 2013) This paper presents the *ITRA* descriptor for action recognition. We reduce a video to a set of key-sequences or meaningful acts. We generate small sequences from a video which are described and clustered. To obtain the video key-sequences from a video we select the sequences that are similar to the class centroids. Therefore, for each video our method selects $K$ key-sequence. Next, we introduce our descriptor *ITRA* to describe the video key-sequences. The *ITRA* descriptor exploits the power of comparative reasoning by using sparse coding to capture relative local similarities among key-sequences.

- (Alfaro, Mery, & Soto, 2016) This paper presents our method for action recognition using Sparse Coding and Relative Features. Here, we present a novel method to Class-based Video Summarization to obtain the video key-frames. We introduce a loss function that leads to the identification of a sparse set of
key-frames that simultaneously capture both, the relevant particularities in the input video and the generalities of videos in the entire class. The video key-sequences from a video are described using $ITRA$ descriptor which quantify the relations among acts.
2. LITERATURE REVIEW

There is a large body of literature related to category-based action recognition in video, as it is one of the most active topics in the area of visual recognition. We refer the reader to (Herath, Harandi, & Porikli, 2017; Aggarwal & Ryoo, 2011) for an extensive review of recent advances in the field. Here, we focus our discussion to previous works that also decompose the input video into a set of key-sequences, propose related video descriptors, or use solutions based on sparse coding.

Key-sequences: Several previous works have tackled the problem of action recognition in video by representing each video through a reduced set of meaningful temporal parts. Carlsson and Sullivan (Carlsson & Sullivan, 2001) propose to manually select a single key-frame to characterize a single body-pose, and then use a shape-matching strategy to perform action recognition. Weiland and Boyer (Weinland & Boyer, 2008) further explore this idea by extending the single key-frame representation to a set of key-frames that they refer to as Exemplars. By selecting key-frames that are informative for the entire class, this strategy reduces intra-class variations, but discards relevant motion information to discriminate among similar actions. Schindler and Van Gool (Schindler & Gool, 2008) focus on studying the amount of frames, or Snippets, needed to recognize periodic actions (for instance walking). However, they do not propose any strategy to automatically select these key-frames. Gaidon et al. (Gaidon et al., 2011) present an action recognition approach that is built on top of atomic action units, or Actoms. A Bag-of-Words (BoW) approach is used to describe each Actom, and a video is represented by a temporal ordered sequence of Actoms.

A relevant disadvantage of the previous methods is that, at training time, they require a manual selection or labeling of a set of key-frames or key-sequences. In terms of methods that do not impose this requirement, a straightforward approach is to describe each video by all its possible key-sequences. An example of this scheme is the approach followed by Guo et al. (Guo, Ishwar, & Konrad, 2010), where they split a video into a set of short key-sequences that overlap and span the complete video. These key-sequences are then used to...
feed a suitable descriptor and a classifier. A major disadvantage of this type of schemes is that they do not take advantage of the benefits of a suitable selection of key-sequences, such as a reduction of problems associated to spurious patterns or variations in action duration.

Discriminative approaches to identify key-frames or key-sequences have also been a popular strategy. Zhao and Elgammal (Zhao & Elgammal, 2008) use an entropy-based score to select as key-frames the most discriminative frames from each video. Recently, Liu et al. (L. Liu et al., 2013) propose a method to select key-frames using the Adaboost classifier to identify highly discriminative frames for each target class. The underlying hypothesis behind these works is that discriminative frames capture representative motion patterns of the corresponding action class. In contrast to our approach, these methods select each frame independently using a discriminative approach, while we jointly select a group of frames using a generative approach.

A set of key-sequences can be considered a temporal simile of a spatial configuration of object-parts. Consequently, latent SVM approaches, originally used for object recognition (Felzenszwalb, Girshick, & McAllester, 2010), have been extended to the case of action recognition (Niebles et al., 2010; Y. Wang & Mori, 2011; Raptis & Sigal, 2013). Niebles et al. (Niebles et al., 2010) represent a video using global information in combination with a video decomposition into short temporal motion segments of variable length. A latent SVM approach is used to learn motion segment classifiers and to infer their latent temporal positions. Raptis and Sigal (Raptis & Sigal, 2013) use a video frame representation based on max-pooled poselet (Bourdev, Maji, Brox, & Malik, 2010) activations, in conjunction with a latent SVM approach to select relevant key-frames and learn action classifiers.

In contrast to discriminative approaches, we use a generative approach that directly focuses on managing intra-class variations. We believe that this strategy leads to the identification of more suitable class-based representatives. Furthermore, we do not assume that all videos in an action class share the same set of key-sequences, but they are adaptively selected from each video, considering their similarities to other local temporal patterns present in the class collection.
Video descriptors: Lately, feature based approaches have been the dominant technique to perform action recognition in video. Extensive research has been oriented to propose suitable spatio-temporal low-level features (Laptev & Lindeberg, 2003; Dollar, Rabaud, Cottrell, & Belongie, 2005). Several of these methods propose extensions of popular low-level visual features used in the context of object or scene recognition. As an example, (Scovanner et al., 2007) and (Kläser et al., 2008) propose spatio-temporal extensions of the popular low-level spatial feature descriptors SIFT (Lowe, 2004) and HOG (Dalal & Triggs, 2005), respectively. BoW approaches have also been extensively used as mid-level representations for action recognition (Kläser et al., 2008; Laptev, Marszalek, Schmid, & Rozenfeld, 2008; J. Liu, Ali, & Shah, 2008; Yu, Kim, & Cipolla, 2010). Recently, feature descriptors based on a dense tracking of feature points that are able to compensate camera motions, have achieved state-of-the-art results on several benchmark datasets (H. Wang, Kläser, Schmid, & Liu, 2013).

Limitations of the previous methods to discriminate similar actions or to provide more semantic representations have motivated the creation of visual descriptors based on higher levels of abstraction. As an example, Liu et al. (J. Liu et al., 2011) represent an action using a set of semantic attributes, such as right leg motion. Also, Sadanand and Corso (Sadanand & Corso, 2012) propose a high-level representation of action videos that is based on the concatenated response of a set of action detectors or Action Bank. In terms of discriminative approaches, Wang and Mori (Y. Wang & Mori, 2011) use a structural SVM approach to jointly learn a mid-level representation and top-level action classifiers.

In our case, we build our descriptor on top of key-sequences that are characterized by low-level spatio-temporal features. In this sense, the proposed descriptor is more closely related to mid-level representations. As a main distinguishable characteristic, the proposed representation focuses on coding local temporal similarities among key-sequences. This frames our approach into the context of relative features or comparative reasoning. In the context of face recognition, Kumar et al. (Kumar et al., 2009) propose a method that exploits facial similarities with respect to a specific list of reference people. They call these features similes. As a remarkable result, they are able to reduce recognition error by
more than 20% with respect to previous published results. Yagnik et al. (Yagnik et al., 2011) presents a locality sensitive hashing approach that provides a feature representation based on relative rank ordering. Using this scheme, they achieve significant improvement in an object recognition task. Similarly, Parikh and Grauman (Parikh & Grauman, 2011) use a max-margin approach to also learn a function for relative rank ordering. In this case, they learn to rank semantic attributes of object classes, reporting clear advantages over alternative feature representations. Wang et al. (G. Wang, Forsyth, & Hoiem, 2013) present a method that uses information about object-class similarities to train a classifier that responds more strongly to examples of similar categories than to examples of dissimilar categories. Using this scheme, they report significant increase in recognition performance for object categories containing few training examples. As a major limitation, the last two methods need a training set with labels about similarity relations among object classes.

The works above share with our approach the idea of explicitly using the relative strength of visual properties to build a representation to achieve visual recognition. However, they are not based on sparse coding techniques or they do not exploit relative temporal relations among visual patterns. As far as we know, representations based on relative features have not been explicitly used before to build a descriptor for action videos.

Sparse Coding: A functional approach to action recognition is to create dictionaries based on low-level representations. Several methods can be used to produce a dictionary, including BoW (Gaidon et al., 2011; Laptev et al., 2008; J. Liu et al., 2008), Fisher vectors (Oneata, Verbeek, & Schmid, 2013; Atmosukarto, Ghanem, & Ahuja, 2012), random forest (Yao, Gall, & Gool, 2010; Yu et al., 2010), and, more recently, sparse coding techniques (Guha & Ward, 2012; Guo et al., 2010; Tran, Kakadiaris, & Shah, 2011; Castrodad & Sapiro, 2012). Guo et al. (Guo et al., 2010) extend the sparse coding approach in (Wright et al., 2009) to the case of action recognition. They divide each input video into a set of sequences encoded using silhouette-based features. Training sequences from all classes are directly used as atoms to form a shared dictionary. This dictionary is used to classify each sequence according to the class that contributes the most to its sparse reconstruction. A
majority vote scheme is then used to classify each new video by combining the classification of its individual sequences. As a relevant drawback, the method used to obtain the video sequences limits each video to contain only one moving object.

Tran et al. (Tran et al., 2011) use motion information from human body parts and sparse coding techniques to classify human actions in video. For each body part, they build a dictionary that integrates information from all classes. Similarly to (Wright et al., 2009), the atoms in each dictionary are given by the training samples themselves. To classify new videos, a majority vote scheme selects the class whose atoms contribute the most to the reconstruction of the body parts detected in the input video. As a main drawback, at training time, this method requires manual annotation of human body parts. Guha and Ward (Guha & Ward, 2012) explore several schemes to construct an overcomplete dictionary from a set of spatio-temporal descriptors extracted from the training videos. In particular, class-specific and class-shared sparse dictionary learning strategies are tested. Using these strategies, they achieve state-of-the-art results on several action recognition datasets and report superior performance for the case of class-specific dictionaries. This method does not use key-sequences or relative features in its operation.

Castrodad et al. (Castrodad & Sapiro, 2012) propose a hierarchical two-level sparse coding approach for action recognition. At the first level, they randomly sample spatio-temporal patches from each training video and use them to obtain a set of class-specific dictionaries. Similarly to our proposed approach, they concatenate these dictionaries and access inter-class relationships by quantifying the support provided by each class to the reconstruction of a projected patch. At the second level, they use inter-class relations to learn a set of dictionaries that are used to test different action classification schemes. In contrast to our approach, this work uses a global representation that discards local temporal information. Furthermore, this method only provides inter-class relations, while our proposed technique allows us to access inter and intra-class relations among local temporal acts.
3. CLASS-BASED VIDEO SUMMARIZATION

It is largely acknowledged that the execution of an action can be effectively summarized by a set of representative poses of the main actor. As an example, in the scientific study of motion, chronophotography, a photographic technique that captures a discrete set of prints from an action, has been extensively used to summarize action sequences (Etienne-Jules, 1902). Therefore, it is possible to represent an action video using a few frames instead of the complete video. Nevertheless, the set of key-frames or key-sequences selected should effectively characterize the action from the video.

This chapter presents two approaches which take advantage of the class information available to decompose a video into a set of relevant frames.

The first method proposes to use a temporal sliding to scan the videos from a class and then generates small sequences. The sequences from a class are described and clusterized. Hence, the centroids from the clusters summarize the overall behavior from the class. To characterize a video with respect to the class we need to select the sequences close to the class centroids; such sequences are namely key-sequences.

The second method addresses the video summarization as a reconstruction problem using information from the class of the videos. Thus, our formulation simultaneously provide a suitable representation for an input video and the complete class from the video.

3.1. A Temporal Sliding Approach to find Key-Sequences

In order to reduce the dimensionality of the recognition problem, we propose summarizing a video sequence using a few representative key-sequences. A key-sequence is a small number of consecutive frames composed of acts (or atomic motions) of an action that can be used to recognize it. For instance, the action ‘diving’ can be split into three acts: ‘jump’, ‘twist’ and ‘entry’. These acts should contain enough information to address a successful classification of the action. The key-sequences should satisfy the following consistency criteria: i) They should strongly characterize the class of an action, which
means that an action can be visually recognized by observing its key-sequences.  

ii) The key-sequences should be temporally sorted, thus, the \( k \)-th key-sequence of a video of an action should be similar to the \( k \)-th key-sequence of another video of the same action. Note that each person has his or her own way for doing an action, which means that the acts may appear to be different. Nevertheless, the acts should be consistent for all videos belonging to the same class.

The procedure for obtaining the key-sequences of a class is repeated for each class \( c \), for \( c = 1 \ldots N \). Let \( B_c = \{ v_i \}_{i=1}^p \) be a set of \( p \) training videos of class \( c \), where \( v_i = \{ f_{i,j} \}_{j=1}^{r_i} \) is a set of \( r_i \) frames. For each video \( v_i \), we built a set \( S_i \) with all of the possible subsequences \( s_{i,j} = \{ f_{i,k} \}_{k=j}^{j+t-1} \) of \( t \) consecutive frames: \( S_i = \{ s_{i,j} \}_{j=1}^{r_i-t+1} \). Thus, the set of subsequences of the class is \( S^c = \{ S_i \}_{i=1}^p \).

Each subsequence in \( S^c \) is described in appearance and motion using the well-known HOG3D descriptor (Kläser et al., 2008). It generates a feature space \( H^c = f_{\text{HOG3D}}(S^c) \) with a large collection of high-dimensional spatio-temporal descriptors. Our goal is to find groups of descriptors with high levels of similarity that summarize the overall behavior from the class. We find these groups by applying a \( K \)-means clustering algorithm over \( H^c \), and define \( Z^c = \{ z_k \}_{k=1}^K \) as the set of the \( K \) estimated centroids for class \( c \). We expect each centroid to represent an act which should be present in every video. Thus, we define the \( K \) key-sequences of video \( v_i \) as those \( K \) subsequences \( \{ \hat{s}_{i,k} \}_{k=1}^K \in S_i \), where \( \hat{s}_{i,k} = s_{i,q(i,k)} \) is a subsequence, and where description \( h_{i,q} = f_{\text{HOG3D}}(s_{i,q}) \) is the most similar one to each centroid of \( Z^c \) and \( q(i,1) < q(i,2) \cdots < q(i,K) \) in order to ensure the temporally sorted subsequences. The key-sequences are estimated as follows: First, we compute the indices of the most similar subsequences as \( w_k = \text{argmin}_{q} \| z_k - h_{i,q} \| \). Second, we sort the indices \( w \) as \( \{ q(i,k) \}_{k=1}^K = \text{sort}(\{ w_k \}_{k=1}^K) \).

Some results from this method can be see in Fig.3.1. It shows the central frame corresponding to the video key-sequences from lifting and diving actions from UCFSports.
database. Each action is decomposes in three atomic acts which describes visually the action in the video. Also, we note that videos belonging to a class has consistent acts even thought it may appear different in each person.

**Figure 3.1.** Shows the central frame from key-sequences of lifting and diving actions to UCF-Sports database. Own Source (Alfaro et al., 2016).

### 3.1.1. Describing the Video Key-Sequences

A sequence of the video $v_i$ could contain not only the actor, but also some noise from dynamic backgrounds, the objects intervening in the scene, clothes, etc. In order to overcome this problem, the actor is detected using a model of body part detection applied to the key-sequences of the video (Andriluka, Roth, & Schiele, 2009). The goal is to extract relevant information from limbs where motions are performed. Thus, given a key-sequence of a video $\hat{s}_{i,k}$ with $t$ frames, we apply the body part model only to the central frame $(t - 1)/2$, for an odd $t$, because it is enough to produce an estimate of the location of the parts and to avoid the cost of making this in each frame. The model delivers the bounding box of ten parts of the body: torso, forearms, arms, thighs, legs and head (see Fig. 3.2).

We then extract local spatio-temporal features from spatio-temporal *cuboids* defined by the bounding box of the body parts propagated across the frames of the key-sequence.
For each cuboid, we randomly generate spatial patches of $n \times n$ pixels and extract sub-cuboids. A sub-cuboid is formed by lining up the 2D patches from each frame of the key-sequence. The size of each spatio-temporal sub-cuboid is $n \times n \times t$. These sub-cuboids are described using HOG3D, which yields a collection of descriptors $Y_{i,k}$ for video $v_i$ and key-sequence $k$. We call this function $Y_{i,k} = f_{low-level}(v_i, k, c)$. Hereafter, the descriptors of the sub-cuboids will be denominated low-level features. The description of the $k$-th key-sequence of all $n_c$ videos of class $c$ are arranged in $Y^c_k = [Y_{1,k} \ldots Y_{i,k} \ldots Y_{n_c,k}]$, and it will be called the low-level features from class $c$ and temporal order $k$.

**Figure 3.2.** Shows the process to extract low-level features from a key-sequence. Own Source (Alfaro et al., 2013).

### 3.2. A Class-based Sparse Coding Approach to find Key-Frames

The problem of key-frames selection can be address as a reconstruction problem using sparse coding techniques (Donoho & Elad, 2003). Let $V = \{v_i\}_{i=1}^p$ be a set of $p$ training videos of a given action class, where video $v_i$ contains $n_i$ frames $f_{i,j}$, $j \in [1 \ldots n_i]$. We encode each frame $f_{i,j}$ using a pyramid of histograms of oriented gradients or PHOG–descriptor (Bosch, Zisserman, & Muñoz, 2007). Then, video $v_i$ is represented by a matrix $Z_i \in \mathbb{R}^{m \times n_i}$, where column $j$ contains the $m$-dimensional PHOG–descriptor of frame $f_{i,j}$.

Our sparse coding representation considers two main design goals. First, similarly to (Elhamifar et al., 2012), the atoms of the resulting representation must correspond to frames from the video. Second, as mentioned before, the resulting atoms must simultaneously
provide a suitable representation of the input video and the complete class. To achieve this, for each input video we solve the following optimization:

\[
\begin{align*}
\min_{W_i, W_{(-i)}} & \|Z_i - Z_i W_i\|_F^2 + \alpha \|Z_{(-i)} - Z_i W_{(-i)}\|_F^2 \\
\text{s.t.} & \|W_i\|_{1,2} \leq \lambda, \quad \|W_{(-i)}\|_{1,2} \leq \lambda, \quad 1^T W_i = 1^T, \quad 1^T W_{(-i)} = 1^T
\end{align*}
\] (3.1)

where \(W_i \in \mathbb{R}^{n_i \times n_i}\) corresponds to the matrix of coefficients that minimize the constrained reconstruction of the \(n_i\) frame descriptors in \(Z_i\). \(Z_{(-i)} = [\ldots, Z_{i-1}, Z_{i+1}, \ldots] \in \mathbb{R}^{m \times (n-n_i)}\)
corresponds to the matrix of PHOG descriptors for all the \( n \) frames in a target class, excluding the \( n_i \) frames from video \( v_i \). \( W(\cdot) = [\ldots, W_{i-1}, W_{i+1}, \ldots] \in \mathbb{R}^{n_i \times (n-n_i)} \) corresponds to the sparse representation of \( Z(\cdot) \) using the frame descriptors in \( Z_i \). The mixed \( \ell_1/\ell_2 \) norm is defined as \( \|A\|_{1,2} \triangleq \sum_{i=1}^{N} \|a_i\|_2 \), where \( A \) is a sparse matrix and \( a_i \) denotes the \( i \)-th row of \( A \). Then, the mixed norm expresses the sum of the \( \ell_2 \) norms of the rows of \( A \). Parameter \( \lambda > 0 \) controls the level of sparsity in the reconstruction, and parameter \( \alpha > 0 \) balances the penalty between errors in the reconstruction of video \( v_i \) and errors in the reconstruction of the remaining videos in the class collection. Following (Elhamifar et al., 2012), we solve the constrained optimization in Eq. 3.1 using the Alternating Direction Method of Multipliers (ADMM) technique (Gabay & Mercier, 1976).

3.2.1. Optimization

In our experiments, we obtain a high action recognition performance using a value of \( \alpha = 1 \) in Eq. (3.1). In this case, we obtain the following optimization problem:

\[
\min_{W_i, W(\cdot)} \| [Z_i \ Z(\cdot)] - Z_i [W_i \ W(\cdot)] \|_F^2 \quad \text{s.t.} \quad \| [W_i \ W(\cdot)] \|_{1,t} \leq \lambda, \quad 1^T [W_i \ W(\cdot)] = 1^T
\]  

(3.3)

By denoting \( Z = [Z_i \ Z(\cdot)] \) and \( W = [W_i \ W(\cdot)] \), the previous formulation reduces to:

\[
\min_W \| Z - Z_i W \|_F^2 \quad \text{s.t.} \quad \| W \|_{1,t} \leq \lambda, \quad 1^T W = 1^T
\]

(3.4)

where \( Z \in \mathbb{R}^{m \times n}, W \in \mathbb{R}^{n_i \times n} \). Eq. (3.4) is a typical optimization problem in sparse coding. In this work, we solve this optimization problem using the Alternating Direction Methods of Multipliers (ADMM) technique (Gabay & Mercier, 1976).

3.2.2. Selection of Key-Sequences

The matrix of coefficients \( W^i = [W_i \ W(\cdot)] \) provides information about the contribution of each frame in \( v_i \) to summarize each of the videos in the entire class collection.
Fig. 3.4 shows a diagram of matrix $W^i$ that highlights this property. Specifically, each row $j$ in $W^i$ provides information about the contribution provided by frame $j$ in video $v_i$, $f_i^j$, to reconstruct the $p$ videos in the class collection. Using this property and the notation in Fig. 3.4, we define the following score to quantify the contribution of frame $f_i^j$ to the reconstruction process:

$$R(f_i^j) = \sum_{l=1}^{p} \sum_{s=1}^{n_i} w_{l,j,s}.$$  \hspace{1cm} (3.5)

$R(f_i^j)$ corresponds to the sum of the elements in the $j$-th row of matrix $W^i$. We use this score to rank the frames in video $v_i$ according to their contribution to the reconstruction process. In particular, a frame with a high ranking score provides a high contribution to the reconstruction of the videos in the class collection. Therefore, high scoring frames represent good candidates to be selected as key-frames for video $v_i$.

Let $L_i$ be the set of frames from $v_i$ that satisfy a minimum ranking score condition. Specifically, $f_i^j \in L_i$ satisfies $R(f_i^j) > \theta$, where $\theta$ is a suitable threshold. We obtain a set of key-frames from $v_i$ by selecting $K$ frames from the candidates in $L_i$. Several criterion can be used to select these $K$ frames. A straightforward one is to select the frames with highest ranking score. However, this criterion does not guarantee that the selected frames are well distributed in time with respect to the temporal span of video $v_i$. As an alternative,
we decide to choose $K$ time instants uniformly distributed with respect to the time span of video $v_i$. Then, for each of these time instances, we select as a key-frame the closest neighboring frame in $L_i$. In our experiments, we adjust threshold $\theta$ in a way that, for regular videos, the number of frames with high ranking score is greater than $K$.

We provide some instances of key-frames obtained from our approach for actions taken from the Olympic dataset (Niebles et al., 2010). The figures 3.5 and 3.6 shows the key-frames obtained from Clean and jerk and Discuss throwing actions. In both cases, it is possible to observe a reduction of inner-class variation among temporally corresponding key-frames. Therefore, our method is able to provide a suitable video representation take in advantage the class information.

Figure 3.5. Key-frames selected by the proposed method for three different videos from Clean and Jerk action in the Olympic dataset using K=3. Own Source.
In some cases, the information contained in the key-frames is enough to perform a successful recognition. However, for actions with similar appearance, such as *run* and *walk*, it is important to also include motion cues. We achieve this by adding neighboring frames to each key-frame in order to form acts that we refer to as key-sequences. Specifically, for a key-frame $f^j_i$ in video $v_i$, its corresponding key-sequence is given by the set $s^j_i = \{f^l_i\}_{l=j-t}^{j+t}$, i.e., $2t+1$ consecutive frames centered at the corresponding key-frame. Consequently, each input video $v_i$ is decomposed into a set $s_i = \{s^1_i, \ldots, s^K_i\}$, corresponding to $K$ temporally ordered key-sequences.
4. RELATIVE FEATURES TO VIDEO DESCRIPTION

The human actions can be similar to each other or they can share features such as the case of ‘run’ and ‘long jump’ actions, for example. When these actions are broken down into their key-sequences we note the presence of the running act in both actions. Subsequently, the local information extracted from this act is clearly similar in both actions. Therefore, we believe that it is possible to represent an video as a combination of characteristics from different actions. A high value of this combination in a given class indicates a high probability that a video belongs to that class.

Our method obtains a video description based on quantifying relative intra and inter-class similarities among local temporal patterns or key-sequences. To quantify these similarities we learn temporal class-dependent dictionaries from the key-sequences. Our key idea is compute cross-projections of acts into dictionaries coming from different temporal positions or action classes to quantify relative local similarities among action categories. As we demonstrate later, these similarities prove to be highly discriminative to perform action recognition in video. We namely such similarities as Relative Local Temporal Features.

An overview that summarizes the main steps to build our video descriptor is presented in Fig. 4.1.

4.1. Relative Local Temporal Features

At the core of our method to obtain relative features is the use of sparse coding to learn a set of dictionaries that encode local temporal patterns present in the action classes. Specifically, in the case of $C$ action classes and $K$ temporal key-sequences, we use training data to learn a total of $C \times K$ dictionaries, where dictionary $D_{c_i k_j}^i$, $c_i \in [1 \ldots C]$, $k_j \in [1 \ldots K]$, encodes relevant local temporal patterns occurring in class $c_i$ at time instance $k_j$.

As a key observation, by projecting a given key-sequence to a concatenated version of a subset of the dictionaries, it is possible to quantify the relative similarity between the key-sequence and the individual dictionaries. This can be achieved by quantifying
the total contribution of the atoms in each individual dictionary to the reconstruction of the projected key-sequence. As an example, consider the case of a concatenated dictionary that encodes local patterns learnt from sport actions. In this case, key-sequences from an action class such as running should use in their reconstruction a significant amount of dictionary atoms coming from similar action classes, such as jogging and walking. As a consequence, by quantifying the cross-talk among reconstruction contributions coming from different dictionaries, one can obtain a feature vector that encodes relative local similarities between the projected key-sequence and the temporal patterns encoded in each dictionary. Next, we exploit this property to apply two concatenation strategies that allow us to obtain a video descriptor capturing inter and intra-class similarity relations.

4.1.1. Inter-class Relative Act Descriptor

Our method to obtain inter-class relative local temporal features is composed of three main steps. In the first step we obtain a low-level feature representation for each key-sequence. Specifically, we randomly sample a set of spatio-temporal cuboids (300 in our experiments) from each key-sequence. These cuboids are encoded using the spatio-temporal HOG3D descriptor (Kläser et al., 2008). Chapter 5 provides further implementation details.

In the second step we use the resulting HOG3D descriptors and sparse coding to build a set of local temporal dictionaries for each class. Temporal locality is given by organizing the key-sequences according to their $K$ temporal positions in the training videos. Let $Y_j^c$ be the set of HOG3D descriptors extracted from all key-sequences occurring at the $j$-th temporal position in the training videos from class $c$, where $j \in [1, \ldots, K], c \in [1, \ldots, C]$. We find a class-based temporal dictionary $D_j^c$ for position $j$ using the K-SVD algorithm (Aharon et al., 2006) to solve:

$$\min_{D_j^c, X_j^c} \|Y_j^c - D_j^cX_j^c\|_F^2 \quad \text{s.t.} \quad \|x_i\|_0 \leq \lambda_1,$$

(4.1)

where $Y_j^c \in \mathbb{R}^{m \times n_s}$, $m$ is the dimensionality of the descriptors and $n_s$ is the total number of cuboids sampled from videos of class $c$ and temporal position $j$, $D_j^c \in \mathbb{R}^{m \times n_s}$, $X_j^c \in \mathbb{R}^{n_s \times n_c}$. Chapter 5 provides further implementation details.
\[ \mathbb{R}^{n_a \times n_a}, \quad n_a \text{ is the number of atoms in each dictionary } D^c_j, \text{ and the sparsity restriction on} \]
\[ \text{each column } x_i \in X^c_j \text{ indicates that its total number of nonzero entries must not exceed } \lambda_1. \]

Finally, in the third step we use the previous set of dictionaries to obtain a local temporal similarity descriptor for each key-sequence. To achieve this, for each temporal position \( j \), we concatenate the \( C \) class-based dictionaries obtained in the previous step. This provides a set of \( K \) temporal dictionaries, where each dictionary contains information about local patterns occurring in all target classes at a given temporal position \( j \). These \( K \) representations allow us to quantify local temporal similarities among the target classes. Specifically, let \( D_j = [D^1_j \ D^2_j \ldots \ D^C_j] \) be the concatenated temporal dictionary corresponding to temporal position \( j \). To obtain a descriptor for key-sequence \( s^i_j \) from video \( v_i \), we first project \( s^i_j \) onto dictionary \( D_j \) imposing a sparsity constraint. We achieve this by using the Orthogonal Matching Pursuit (OMP) (Pati, Rezaiifar, & Krishnaprasad, 1993) technique to solve:
\[
\min_{x^i_j} \|s^i_j - D_j x^i_j\|_F^2 \quad \text{s.t. } \|x^i_j\|_0 \leq \lambda_2, \tag{4.2}
\]
where vector \( x^i_j = \{x^i_j[D^1_j], \ldots, x^i_j[D^C_j]\} \) is the resulting set of coefficients, and a component vector \( x^i_j[D^c_j] \in \mathbb{R}^{n_a} \) corresponds to the coefficients associated to the projection of \( s^i_j \) onto the atoms in subdictionary \( D^c_j \).

We quantify the similarity of \( s^i_j \) to the atoms corresponding to each class by using a sum-pooling operator that evaluates the contribution provided by the words in each subdictionary \( D^c_j \) to the reconstruction of \( s^i_j \). We define this sum-pooling operator as:
\[
\phi^c_j(s^i_j) = \sum_{l=1}^{n_a} x^i_j[D^c_j](l). \tag{4.3}
\]

By applying the previous method to the set of \( K \) key-sequences \( s^i_j \) in a video \( v_i \), we obtain a video descriptor \( \Phi^i = [\phi^1, \ldots, \phi^K] \in \mathbb{R}^{C \times K} \), where each component vector \( \phi^i \) is given by \( \phi^i = [\phi^1_j, \ldots, \phi^K_j] \). In this way, \( \Phi^i \) contains information about relative inter-class similarities among key-sequences or acts. Therefore, we refer to this descriptor as Inter-class Relative Act Descriptor.
4.1.2. Intra-class Relative Act Descriptor

The procedure in Section 4.1.1 provides a descriptor that encodes relative local temporal similarities across the target classes. In this section, we use a similar procedure to obtain local temporal similarities at an intra-class level. This scheme provides informative features to characterize the temporal dynamics corresponding to each action class. Specifically, we quantify the similarity of a key-sequence occurring at temporal position $j$ with respect to the patterns occurring at the remaining $K-1$ temporal positions in a target class.

To do this, we follow the procedure described in Section 4.1.1 but, this time we project a key-sequence $s_i^j$ onto the concatenated dictionary $D^{c}_{(-j)} = [..., D^{c}_{j-1}, D^{c}_{j+1}, ...]$, i.e., the concatenation of the $k-1$ key-sequence dictionaries for class $c$, excepting the dictionary corresponding to temporal position $j$. We again use the OMP technique to perform this projection, i.e., to solve:

\[
\min_{x} \| s_i^j - D^{c}_{(-j)} x_i^j \|_F^2 \quad \text{s.t.} \quad \| x_i^j \|_0 \leq \lambda_3. \tag{4.4}
\]

Similarly to the Inter-class Relative Act Descriptor, we obtain a video descriptor, $\Psi = [\psi^1, ..., \psi^K] \in \mathbb{R}^{K \times (K-1)}$, by applying the projection to all key-sequences in a video $v_i$ and then using the corresponding sum-pooling operations to quantify the reconstruction contribution of each subdictionary $D^{c}_j$. In this way, $\Psi$ contains information about relative intra-class similarities among key-sequences or local acts, therefore, we refer to this descriptor as Intra-class Relative Act Descriptor.

4.1.3. Inter-Temporal Relational Act Descriptor (ITRA)

We obtain a final feature vector descriptor for a video $v_i$ by concatenating the Inter and Intra-class Relative Act Descriptors. We refer to this new descriptor as Inter Temporal Relational Act Descriptor or ITRA, where $\text{ITRA}(v_i) = \{ \Phi^i \cup \Psi^i \} \in \mathbb{R}^{K \times (C+(K-1))}$. 

29
4.2. Video Classification

ITRA descriptors can be used to feed any off-the-shelf supervised classification scheme, such as a random forest or a SVM classifier. Here, we use a classification scheme based on sparse coding. We provide next the details of this scheme.

4.2.1. Training

During the training phase, we first use the method described in the previous chapter (See Chap.3) to decompose each training video into a set of key-sequences. Later, each training video is described using the ITRA Descriptor presented in this chapter.

Afterwards, these descriptors, along with sparse coding techniques, are used to build a dictionary for each target class. Specifically, let $Y^c$ be a matrix containing in its columns the ITRA descriptors corresponding to the training videos from action class $c \in [1, \ldots, C]$.

For each action class, we use the K-SVD algorithm (Aharon et al., 2006) to obtain a class-based dictionary $B^c$ by solving:

$$
\min_{B^c, X^c} \|Y^c - B^c X^c\|_F^2 \quad \text{s.t.} \quad \|x_i\|_0 \leq \lambda_4, \quad \forall i,
$$

where $B^c \in \mathbb{R}^{|\text{ITRA}| \times n_a}$, $|\text{ITRA}|$ represents the dimensionality of the ITRA descriptor, and $n_a$ is the selected number of atoms to build the dictionary. $X^c$ corresponds to the matrix of coefficients and vectors $x_i$ to its columns.

As a final step, we concatenate the $C$ class-based dictionaries to obtain the joint dictionary $B = [B^1|B^2|\cdots|B^C]$ that forms the core representation to classify new action videos.

4.2.2. Inference

To classify a new input video, similarly to the training phase, we first use the method in Chapter 3 to decompose the video and then we obtain its ITRA descriptor. As a relevant difference from the training phase, in this case we do not know the class label of the input video, therefore, we need to obtain its key-sequence decomposition with respect to each
target class. This task leads to $C$ ITRA descriptors to represent each input video. Consequently, the classification of an input video consists of projecting these $C$ descriptors onto the joint dictionary $B$ and then using a majority vote scheme to assign the video to the class that contributes the most to the reconstruction of the descriptors. Specifically, let $v_q$ be a test video, and $\Omega^c(v_q)$ its ITRA descriptor with respect to class $c$. We obtain a sparse representation for each of the $C$ ITRA descriptors using the OMP technique to solve:

$$\min_{\alpha_c} \|\Omega^c(v_q) - B\alpha_c\|_2^2 \quad \text{s.t.} \quad \|\alpha_c\|_0 \leq \lambda_5.$$  (4.6)

The previous process provides $C$ sets of coefficients $\alpha_c$. We use each of these sets to obtain a partial classification of the input video. We achieve this by applying to each set a sum-pooling operator similar to the one presented in Eq. (4.3), and classifying the input video according to the class that presents the greatest contribution to the reconstruction of the corresponding set $\alpha_c$. Finally, using these $C$ partial classifications, we use majority vote to assign the input video to the most voted class.
Figure 4.1. Presents the overview of the method to obtain the ITRA descriptor. Own Source (Alfaro et al., 2016).
5. EXPERIMENTS AND RESULTS

This chapter presents the validation of our proposed method by conducting a series of experiments. We start the chapter by providing implementation details about the selection of the main parameters behind our approach. Afterwards, we test our approach using three popular benchmark datasets for action recognition: KTH (Schüldt, Laptev, & Caputo, 2004), Olympic (Niebles et al., 2010), and HOHA (Laptev et al., 2008). We also use these datasets to compare the recognition performance of our method to the results of several state-of-the-art techniques. Finally, we use different baselines to evaluate the relevance of the two main contributions of our method: (i) A new proposed method to select key-frames, and (ii) A new proposed similarity-based descriptor, ITRA.

5.1. Experimental Settings

In all our experiments, we select values for the main parameters of the proposed method using the following criteria.

5.1.1. Estimation of Key-Sequences:

Underlying the estimation of the key-sequences is the estimation of the parameter $K$, that controls the number key-frames selected from each video. Decomposing a video into many key-sequences or acts might imply a high overlap, particularly in the case of periodic actions. As an example, the action run is cyclic and, therefore, the selection of a high number of acts might produce a high level of redundancy. We use training data from the Olympic dataset to tune the number of acts to represent an action. Experimentally, we find that 3 acts are enough to achieve high recognition performance. Hence, in all our experiments we select $K = 3$ key-sequences to represent each training or test video.

In terms of the time span of each key-sequence, as suggested by previous works (Schindler & Gool, 2008; Guo et al., 2010), each act should consist of a short number of frames. This improves the estimation of local motion and also reduces computational complexity. In our experiments, we take a fixed group of 7 frames ($t = 3$) to form each
key-sequence. For each sequence we randomly extract 300 cuboids, which are described using HOG3D (300 dimensions). To filter out uninformative cuboids, we set a threshold to the magnitude of the HOG3D descriptor. We calibrate this threshold to eliminate the 5% least informative cuboids from each dataset. Afterwards, the remaining descriptors are normalized. Table 5.1 shows the value of the resulting thresholds for each dataset.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Train</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>KTH</td>
<td>2.5</td>
<td>2.5</td>
</tr>
<tr>
<td>Olympic</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>HOHA</td>
<td>1.3</td>
<td>1.6</td>
</tr>
</tbody>
</table>

Table 5.1. Presents the thresholds used to filter out uninformative cuboids from the key-sequences. For each dataset, we calibrate this threshold to eliminate the 5% least informative cuboids. Own Source (Alfaro et al., 2016).

5.1.2. Estimation of ITRA descriptor:

Parameters for the extraction of ITRA descriptors are related to the construction of the dictionaries described in Section 4. Let $\mu$ be the redundancy $^1$ and let $\delta$ be the dimensionality of a descriptor. Following the empirical results in (Guha & Ward, 2012), we fix the number of atoms in each local dictionary to be $n_a = \mu \times \delta$. Therefore, the number of atoms for the concatenated dictionaries are: $P = \mu \times \delta \times C$ for the extraction of Inter-class Relative Act Descriptors, $\Phi$, and $P = \mu \times \delta \times (K - 1)$ for the extraction of Intra-class Relative Act Descriptors, $\Psi$. In our experiments, we use $\mu = 2$ and $\delta = 300$, as a result, the dimension of the ITRA descriptors for KTH, Olympic, and HOHA datasets are 24, 54 and 30, respectively. Also, following (Guha & Ward, 2012), the sparsity parameters $\lambda_1$, $\lambda_2$, and $\lambda_3$, are set to be 10% of the number of atoms.

The Parameters for the classification step, i.e., $P$, $\mu$, $\lambda_4$, and $\lambda_5$ are configured using the same scheme described above.

---

$^1$Redundancy indicates the folds of basis vectors that need to be identified with respect to the dimensionality of the descriptor.
5.2. Action Recognition Performance

5.2.1. KTH Dataset:

This set contains 2391 video sequences displaying six types of human actions. Each action is performed by 25 actors in four different scenarios: outdoors, outdoors with scale variation, outdoors with different clothes, and indoors, as illustrated in Fig. 5.1. In our experiments we use the original setup (Schüldt et al., 2004) to divide the data into training and test sets. Table 5.2 shows the recognition performance reached by our method. Table 5.2 also includes the performance of alternative action recognition schemes proposed in the literature, including approaches that also use sparse coding techniques (Alfaro et al., 2013; Castrodad & Sapiro, 2012). To guarantee a fair comparison, we compare our approach to methods that use the same testing protocol.

Our method obtains a recognition performance of 97.5%. Fig. 5.2 shows the confusion matrix reported by our method. Actions such as boxing, hand clapping, and hand waving present some level of confusion. This is expected because these actions are characterized by similar hand motions. A similar situation is observed for the actions jogging, running, and walking, due to the leg-centered motion. Nevertheless, our method obtains high accuracy in all classes. This implies that, for this dataset, the selected acts and their local temporal relationships capture highly discriminative information.

<table>
<thead>
<tr>
<th>Method</th>
<th>Acc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laptev et al. (Laptev et al., 2008) (2008)</td>
<td>91.8%</td>
</tr>
<tr>
<td>Niebles et al. (Niebles et al., 2010) (2010)</td>
<td>91.3%</td>
</tr>
<tr>
<td>Castrodad et al. (Castrodad &amp; Sapiro, 2012) (2012)</td>
<td>96.3%</td>
</tr>
<tr>
<td>Alfaro et al. (Alfaro et al., 2013) (2013)</td>
<td>95.7%</td>
</tr>
<tr>
<td>Zhang et al. (Zang, Gao, Chen, Luo, &amp; Sang, 2017) (2017)</td>
<td>96.8 %</td>
</tr>
<tr>
<td>Our method</td>
<td>97.5%</td>
</tr>
</tbody>
</table>

Table 5.2. Recognition performance of our method and several alternative techniques on KTH dataset. In all cases, the same testing protocol is used. Own Source.
FIGURE 5.1. Shows samples from KTH dataset for four scenarios (rows), s1 = outdoors, s2 = outdoors with scale variation, s3 = outdoors with different clothes, and s4 = indoors. Source (Schüldt et al., 2004).

FIGURE 5.2. Presents the confusion matrix of our method on KTH dataset. Own Source

5.2.2. Olympic Dataset:

This dataset contains 16 actions corresponding to 783 videos of athletes practicing different sports (Niebles et al., 2010). This dataset is challenging because its videos display camera motions and different viewpoints. Fig. 5.3 shows sample frames displaying the
action classes. In our experiments, we use the original setup (Niebles et al., 2010) to divide the data into training and test sets. Table 5.3 shows the recognition performance reached by our method and several alternative state-of-the-art techniques. Our approach achieves a recognition rate of 96.3%. This is a remarkable increase in performance with respect to previous state-of-the-art approaches. Fig. 5.3 shows the confusion matrix reported by our method. We note that many actions from this dataset have a perfect recognition rate: basketball, discus throw, diving platform, hammer throw, javeling throw, long jump, pole vault, shot put, snatch, tennis serve, triple jump, and vault. Therefore, our approach effectively captures relevant acts and their temporal relationships. This allows us to discriminate between actions involving whole body motions, such as hammer throw, and actions involving moderated motions and short durations, such as clean and jerk, pole vault, and tennis serve. Furthermore, actions sharing similar act representations such as basketball, javeling, and triple jump can be discriminated by exploiting intra-class temporal relations among their main acts.

<table>
<thead>
<tr>
<th>Method</th>
<th>Acc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Niebles et al. (Niebles et al., 2010) (2010)</td>
<td>72.1%</td>
</tr>
<tr>
<td>Liu et al. (J. Liu et al., 2011) (2011)</td>
<td>74.4%</td>
</tr>
<tr>
<td>Jiang et al. (Jiang, Dai, Xue, Liu, &amp; Ngo, 2012) (2012)</td>
<td>80.6%</td>
</tr>
<tr>
<td>Alfaro et al. (Alfaro et al., 2013) (2013)</td>
<td>81.3%</td>
</tr>
<tr>
<td>Zhang et al. (Zang et al., 2017) (2017)</td>
<td>91.3%</td>
</tr>
<tr>
<td>Our method</td>
<td>96.3%</td>
</tr>
</tbody>
</table>

Table 5.3. Recognition performance of our method and several alternative techniques on Olympic dataset. In all cases, the same testing protocol is used. Own Source (Alfaro et al., 2016).

5.2.3. Hollywood Dataset:

This dataset contains video clips extracted from 32 movies and displaying 8 action classes. This is a complex dataset, as it includes severe appearance, scale, and viewpoint changes, among other visual complexities. Fig. 5.4 shows sample frames displaying the action classes. 12 movies compose the training set and the rest are used for testing. This leads to a training set containing 219 videos and a test test containing 211 videos. We use only
the videos with manual annotations (clean training file) and we limit the dataset to videos with a single label. This is the same testing protocol used by the alternative techniques considered here. Table 5.4 shows the recognition performance of our method and several alternative state-of-the-art techniques. Our approach achieves a recognition rate of 71.9%. Again, this is a remarkable increase in performance with respect to previous state-of-the-art approaches. Fig. 5.4 shows the confusion matrix reported by our method. Actions such as *answer phone*, *handshake*, and *hug person* obtain high recognition rates. These actions describe interactions between human-human and human-object that our method is able to correctly identify, in spite of the visual complexities present in the videos. In contrast, the actions *get out car*, *kiss*, and *sit up* present a lower recognition performance, with recognition rates close to 60.0%. According to the confusion matrix in Fig. 5.4, these actions present a high confusion rate with respect to the action *answer phone*. A manual inspection of the videos reveals that all these low scoring actions present a common pattern given by a slow incorporation of an actor into the current scene. As an example, in several videos of the class *get out car*, the actor enters the scene presenting motion patterns similar to the ones in several videos of the *answer phone* and *sit up* action classes.

<table>
<thead>
<tr>
<th>Method</th>
<th>Acc.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laptev et al. (Laptev et al., 2008) (2008)</td>
<td>38.4%</td>
</tr>
<tr>
<td>Wang et al. (H. Wang, Ullah, Kläser, Laptev, &amp; Schmid, 2009) (2009)</td>
<td>47.4%</td>
</tr>
<tr>
<td>Wu et al. (Wu, Oreifej, &amp; Shah, 2011) (2011)</td>
<td>47.6%</td>
</tr>
<tr>
<td>Zhang et al. (Zang et al., 2017) (2017)</td>
<td>62.8%</td>
</tr>
<tr>
<td>Our method</td>
<td>71.9%</td>
</tr>
</tbody>
</table>

**TABLE 5.4.** Recognition performance of our method and several alternative techniques on HOHA dataset. In all cases, the same testing protocol is used. Own Source (Alfaro et al., 2016).
5.3. Evaluation of Method to Extract Key-Sequences

An important aspect of our approach is the extraction of suitable key-sequences and the related method to obtain key-frames. In this section, we evaluate the relevance of the proposed method to obtain key-frames by replacing this step of our approach by alternative strategies. Besides this modification, we maintain the remaining steps of our approach and use the same parameter values reported in Section 5.2. In particular, we implement the following three baseline strategies to obtain a set of key-frames:

- **Random Selection (RS):** We extract $K$ random key-frames from each video and we sort them in temporal order.
- **Uniform Selection (US):** We split the video into $K$ equal-sized temporal segments and select the central frame of each segment as a key-frame.
- **K-Means (KS):** We generate all possible video sequences containing $2t+1$ frames by applying a temporal sliding window. We then apply the K-Means clustering algorithm to each class to obtain $K$ cluster centers per class. For each video, we select as key-frames the most similar descriptor to each cluster center.

Table 5.5 shows the resulting recognition performance. As shown in Table 5.5, there is a significant drop-off in recognition performance when we obtain the key-sequences using any of the alternative baselines. This result highlights the relevance of using a suitable technique to reduce inner-class variations when selecting the set of key-frames.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Method</th>
<th>RS</th>
<th>US</th>
<th>KS</th>
<th>Ours</th>
</tr>
</thead>
<tbody>
<tr>
<td>HOHA</td>
<td>33.6%</td>
<td>34.2%</td>
<td>37.2%</td>
<td><strong>71.9%</strong></td>
<td></td>
</tr>
<tr>
<td>Olympic</td>
<td>44.7%</td>
<td>46.3%</td>
<td>63.4%</td>
<td><strong>96.3%</strong></td>
<td></td>
</tr>
</tbody>
</table>

**Table 5.5.** Performances of our method and alternative strategies to extract key-sequences.

5.4. Evaluation of ITRA descriptor

We evaluate the effectiveness of our ITRA descriptor by replacing this part of our approach by alternative schemes to obtain a video descriptor. These alternative schemes
are also based on sparse coding techniques but they do not exploit relative local or temporal information. Specifically, we consider the following three baseline strategies:

- **Ignoring relative local temporal information** (B1): All key-sequences from all temporal positions are combined to build a single class-shared joint dictionary that do not preserve temporal order among the key-sequences. This baseline can be considered as a BoW type of representation that does not encode relative temporal relations among key-sequences.

- **Ignoring key-sequence dependencies** (B2): key-sequences for each temporal position are considered independently for dictionary construction and for classification. A majority vote scheme is used to integrate the classification of the individual key-sequences. Consequently, while ITRA descriptor concatenates the key-sequence information to provide a single descriptor that preserves temporal order, this baseline ignores key-sequence dependencies.

- **Ignoring intra-class relations** (B3): this baseline only considers the term in ITRA descriptor associated to the *Inter-Class Relative Act Descriptor* $\Phi^i$, discarding intra-class relations provided by the *Intra-Class Relative Act Descriptor* $\Psi^i$.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Method</th>
<th>B1</th>
<th>B2</th>
<th>B3</th>
<th>Ours</th>
</tr>
</thead>
<tbody>
<tr>
<td>HOHA</td>
<td></td>
<td>42.2%</td>
<td>19.2%</td>
<td>51.3%</td>
<td>71.9%</td>
</tr>
<tr>
<td>Olympic</td>
<td></td>
<td>72.4%</td>
<td>29.1%</td>
<td>87.3%</td>
<td>96.3%</td>
</tr>
</tbody>
</table>

**Table 5.6.** Performances of our method and alternative strategies to construct the video descriptor using sparse coding techniques.

Table 5.6 shows that the ITRA descriptor largely outperforms the recognition performances provided by the alternative baseline strategies. This highlights the relevance of considering local temporal relations among atomic action acts.
Figure 5.3. Shows samples from each action class of Olympic dataset (Top) and the confusion matrix for our method on Olympic dataset (Bottom). Own Source (Alfaro et al., 2016).
FIGURE 5.4. Shows samples from each action class of HOHA dataset (Top) and the confusion matrix for our method on HOHA dataset (Bottom). Own Source (Alfaro et al., 2016).
6. CONCLUSIONS AND FUTURE RESEARCH

In this thesis, we proposed a novel method for category-based action recognition in video. As main contributions, this method incorporates: (i) An adaptive scheme to decompose each video into a set of atomic acts or key-sequences, and (ii) A new video descriptor that captures relative local temporal relations among atomic acts. As a main result, our experiments show that the proposed method reaches remarkable action recognition performance on three popular benchmark datasets. In particular, it is able to reduce recognition errors by more than 20% with respect to alternative state-of-the-art techniques. This demonstrates that the selected acts and their local inter and intra-class temporal relationships capture highly discriminative information.

In terms of the proposed method to extract key-sequences, it demonstrates to be effective extracting meaningful representative acts that capture relevant intra-class patterns from each input video. In particular, when we replace this method by alternative strategies to obtain the key-frames, such as using a fixed uniform policy or a clustering algorithm, we observe a large drop in recognition performance.

In terms of the ITRA descriptor, it is remarkable that using a reduced dimensionality, between 24 and 54 dimensions for the datasets considered here, it provides a representation that offers high levels of discrimination. Considering the most popular visual descriptors in use today, this result provides support to a wider use of feature representations that explicitly exploit relative similarity relations among visual patterns.

In terms of the complete classification strategy, while the training phase requires an intensive dictionary learning step, that scales linearly with the number of key-sequences and target classes, this step can be easily decomposed into independent dictionary learning problems that can be executed in parallel. Furthermore, the reduced dimensionality of the ITRA descriptor provides a fast classification scheme.

As future work, the ITRA descriptor opens the possibility to explore several strategies to concatenate the basic dictionaries to access different relative similarity relationships among the target classes.
References


