Video Classification System Based on Representation Techniques Using Kernel Methods and Bayesian Inference

Jorge Luis Fernández Ramírez

Thesis submitted as a partial requirement to receive the grade of:
Master in Electrical Engineering

Advisor: Andrés Marino Álvarez Meza, Ph.D.

Co-advisor: Álvaro Ángel Orozco Gutiérrez, Ph.D.

Academic Research Group: Automatics Research Group

Universidad Tecnológica de Pereira
Faculty of Engineering - Electric Engineering program
Master in Electrical Engineering
Pereira, Risaralda, Colombia
2019

Acknowledgments

First of all, I want to thank my family for their unconditional support, love and comprehension. I would like to express my gratitude to professors Andrés Marino Álvarez and Álvaro Orozco for their orientation during this research. Besides, I thank all my peers in the Automatics Research group for their support, advisement and fellowship. I recognize that this research would not have been possible without the financial assistance provided by the Universidad Tecnológica de Pereira through its project of postgraduate scholarships. I would like to thank the project "Desarrollo de un sistema de identificación de estructuras nerviosas en imágenes de ultrasonido para la asistencia del bloqueo de nervios periféricos, aplicación al tratamiento de dolor agudo traumático y prevención del dolor neuropático crónico" with code 1110-744-55958 funded by COLCIENCIAS. Also, I thank to the Vicerrectoria de Investigaciones, Innovación y Extensión from the same university for partially funding this research under the project E6-19-2.

Jorge L. Fernández Ramírez
2019

Abstract

In this work, we proposed different feature representation strategies for video processing. Our main goal is to reveal discriminant patterns from video data for enhancing the Computer Vision task, Human Action Recognition. To this end, we proposed to use a Kernel-based relevance analysis for recognizing the most relevant descriptors related to action recognition. Moreover, the proposal allows computing a linear projection matrix for mapping video samples into a new space, were class separability is preserved and representation dimensionality reduced. Additionally, a new data encoding framework is presented to improve the usual pipeline for performing action recognition. The methodology based on the Infinite Gaussian Mixture Model allows revealing a set of discriminant local spatiotemporal features that enable the precise codification of visual information. Furthermore, by automatically inferring every parameter in the encoding process, our approach reduces the computational complexity of the recognition system by avoiding exhaustive search on model parameters. As a final result, human behavior analysis, which is a particular case of action recognition, is studied. With this task, there is a need for high-level semantic features to allow the proper transcription of the human activity. Besides, the presence of unusual human behaviors introduces the data imbalance challenge. To assess the proposed video processing methodologies, we use real-world datasets. Attained results are presented in terms of supervised measures and compared against state of the art approaches. From these results, it is shown that the use of representation techniques using kernel methods and Bayesian inference favors human actions recognition tasks, obtaining promising performance, and the basis for exciting future work.

Keywords:

Kernel methods, Bayesian Inference, Human Action Recognition, Behavior analysis, video processing, Relevance analysis, data imbalance.

Contents

Chapter 1

Introduction

1.1 Motivation

Human Action Recognition (HAR) has become an important research area in the computer vision field due to its wide range of applications, including automatic video analysis, video indexing and retrieval, video surveillance, and virtual reality [1]. As a result of the increasing amount of video data available both on internet repositories and personal collections, there is a strong demand for understanding the content of complex real-world data. However, different challenges arises for action recognition in realistic video data [2]. First, there is large intra-class variation caused by factors such as the style and duration of the performed action, scale changes, and sudden motion. Second, background clutter, occlusions, and low-quality video data are known to affect robust recognition as well. Finally, for large-scale datasets, the video data processing represents a crucial computational challenge to be addressed [3].

Behavior analysis is a subdivision of HAR that seeks to monitor complex individual and collective human behaviors for creating Intelligent surveillance systems. In which the lack of sociological context features hampers the proper transcription of human activity [4]. Automatic behavior recognition comprises several hierarchical layers of processing, ranging from low-level features extraction to high-level semantic interpretation [5]. The low-level processing stages include pedestrian tracking, motion description, and gaze estimation to characterize humans in the scene. However, this information does not allow the understanding of their behavior as the scene context is not considered [6]. Therefore, the high-level semantic interpretation is required to identify human action descriptors, that enable introducing information about space layout, social environment, and non-verbal behavioral interactions [7]. Furthermore, the inclusion and combination of these factors imply an in-depth analysis of their importance on understanding different sociological contexts to describe behavioral patterns [8].

1.2 State of the Art

The most popular framework for action recognition is the Bag of visual Words (BOW), and its variations [9, 10]. The BOW pipeline contains three main processing stages: feature extraction, data encoding, and classification. Besides, there are several pre-processing stages including relevance analysis and normalization approaches, to enhance data separability, interpretability, and overall improving recognition performance [11]. For the feature estimation step, the recent success of local space-time features like Dense Trajectories (DT) and Improved Dense Trajectories (iDT) has lead researchers to use them on a wide variety of datasets, obtaining excellent results [2, 10]. The success of this these techniques is based on the codification of human motion information and description of local space. Allowing a more adequate representation of the human activity, than most traditional approaches based on the generalization of image feature extraction techniques.

Regarding the feature encoding step, super-vector based methods such as Fisher Vector (FV) and Vector of Locally Aggregated Descriptors (VLAD) are presented as the state-of-the-art approaches for data quantization in action recognition tasks [1, 9]. The encoding representation is crucial for the performance of every action recognition system, as it influences directly the classifier ability to predict class labels. However, video representations generated by methods such as FV or VLAD provide high dimensional super-vectors descriptors arising a set of difficult processing challenges [1, 2]. On one hand, high dimensional data affects the classifier performance adversely, by using redundant information or even noise. On the other hand, high dimensional representations combined with large number of samples, which is the case of many HAR tasks, increases the computational complexity, number of operations and memory requirements, greatly.

The Dimensionality Reduction (DR), which consists of feature selection and feature embedding methods, is imperative to lighten the burden associated with the encoding stage, eliminate redundant information, and project samples into new spaces for increasing separability [12]. Conventional methods, such as Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) were proposed to decorrelate individual features and reduce representation length [13]. Nevertheless, these methods are specially designated to work with real-valued vectors coming from flat Euclidean spaces. Thus, in modern computer vision due to real-world data and models, there is growing interest to go beyond the extensively studied Euclidean spaces and analyze more realistic non-linear scenarios for better data representation [14].

Both FV and VLAD methods are supported by Gaussian Mixture Models (GMMs) to create a generative model of the data to be quantized [15]. These methods perform a similarity comparison between data samples and the obtained generative model, for encoding the visual information through determining each Gaussian responsibility on explaining data samples [16].

However, working with GMMs trained by optimization based methods, e.g. Expectation Maximization, requires extensive crossvalidation for selecting the number of Gaussian components in the mixture model [17]. Moreover, the initialization required by these training methods tends to make fall models into local minima [18]. Therefore, using conventional optimization methods for training GMMs implies a large number of operations and memory requirements, that are not always available when developing HAR systems.

1.3 Objectives

1.3.1 General objective

To develop a video classification system using local spatio-temporal features and representation frameworks based on Bayesian inference and kernel methods to support Human Action Recognition tasks. The proposed system must be useful to highlight discriminating patterns that favor data separability and interpretability.

1.3.2 Specific objectives

1. To develop a relevance analysis framework for selecting and embedding local spatio-temporal features using kernel methods. The framework must calculate a feature representation space that favors data separability and intepretability in Human Action Recognition tasks. Attained results will be compared against state of the art algorithms by using supervised performance measures.

2. To develop a data encoding framework using Bayesian inference probabilistic models that enable revealing discriminant sets of local spatio-temporal features. Proposed methodology must enable the effective codification of visual information for enhancing Human Action Recognition tasks.

3. To develop a feature representation framework based on high-level semantic features and kernel methods to support Human Behavior Analysis at indoors settings. The proposed methodology must allow the transcription of human behavior by considering the scene context. The performance of the framework will be tested against state of the art approaches by using supervised measures given ground-truth data.

Chapter 2

Kernel-based relevance analysis for selecting and embedding local spatio-temporal features in video-based Human Action Recognition

In this chapter, we introduce a new human action recognition framework using kernel relevance analysis. The methodology, based on a non-linear representation of the super-vector obtained after Fisher Vector (FV) encoding, seeks to reduce dimensionality, enhance separability, and interpretability of video representation. Specifically, our approach includes a centered kernel alignment (CKA) technique to recognize relevant descriptors related to action recognition. Hence, we match trajectory-aligned descriptors with the output labels (action categories) through non-linear similarity comparisons [11]. Also, the CKA-algorithm allows to compute a linear projection matrix, where the columns represent those features whose eigenvalues are larger that the mean of them all. Therefore, by projecting video samples into the CKA generated space, the class separability is preserved, and the number of dimensions is reduced. Attained results on the UCF50 database demonstrate that our proposal favors the interpretability of the commonly employed descriptors in action recognition, and presents a system able to obtain competitive performance using a drastically reduced feature space dimensionality.

2.1 Descriptor selection and feature embedding

Let $\{\boldsymbol{Z}_n \in \mathbb{R}^{T \times D}, y_n \in \mathbb{N}\}_n^N$ be an input-output pair set holding N video samples, each of them represented by T trajectories generated while tracking a dense grid of pixels. The local space of each trajectory is characterized by a descriptor of dimensionality D, as presented in [2]. Here, the samples are related to a set of human action videos meanwhile the descriptor in turn is one of the following trajectory-aligned measures: trajectory positions (Trajectory), Histogram of Oriented Gradients (HOG), Histogram of Optical Flow (HOF), Motion Boundary Histograms (MBHx and MBHy), yielding a total of $F = 5$ descriptors. Likewise, the output label y_n denotes the human action being performed in the corresponding video representation. From $\boldsymbol{Z}_n$, we aim to encode T described trajectories concerning a Gaussian Mixture Model (GMM), trained to be a generative model of the descriptor in turn. Then, the Fisher Vector (FV) feature encoding technique is employed, as follows [19]:

Let $\boldsymbol{Z}_n$ be a matrix holding T described trajectories $\boldsymbol{z}_t \in \mathbb{R}^D$, and υ^λ be a GMM. The GMM has parameters $\lambda = \{w_i \in \mathbb{R}, \boldsymbol{\mu}_i \in \mathbb{R}^D, \sigma_i^2 \boldsymbol{I} \in \mathbb{R}^{D \times D}\}_{i=1}^K$, which are respectively the mixture weight, mean vector, and covariance matrix of K Gaussians. We assume that $\boldsymbol{z}_t$ is generated independently by υ^λ. Therefore, the gradient of the log-likelihood describes the contribution of the parameters to the generation process:

$$\boldsymbol{x}_n^\lambda = \frac{1}{T} \sum_{t=1}^{T} \nabla_\lambda \log \upsilon_\lambda(\boldsymbol{z}_t) \tag{2.1}$$

where ∇_λ is the gradient operator w.r.t λ. Mathematical derivations lead $\boldsymbol{x}_n^{\mu,i}$ and $\boldsymbol{x}_n^{\sigma,i}$ to be the D-dimensional gradient vectors w.r.t the mean and standard deviation of the Gaussian i, that is:

$$\boldsymbol{x}_n^{\mu,i} = \frac{1}{T\sqrt{w_i}} \sum_{t=1}^{T} \boldsymbol{\gamma}_t(i) \left(\frac{\boldsymbol{z}_t - \boldsymbol{\mu}_i}{\sigma_i} \right), \tag{2.2}$$

$$\boldsymbol{x}_n^{\sigma,i} = \frac{1}{T\sqrt{2w_i}} \sum_{t=1}^{T} \boldsymbol{\gamma}_t(i) \left[\frac{(\boldsymbol{z}_t - \boldsymbol{\mu}_i)^2}{\sigma_i^2} - 1 \right] \tag{2.3}$$

where $\boldsymbol{\gamma}_t(i)$ is the soft assignment of trajectory $\boldsymbol{z}_t$ to the Gaussian i, that is:

$$\boldsymbol{\gamma}_t(i) = \frac{w_i \upsilon_i(\boldsymbol{z}_t)}{\sum_{j=1}^{K} w_j \upsilon_j(\boldsymbol{z}_t)} \tag{2.4}$$

The final gradient vector $\boldsymbol{x}_n^{\lambda}$ is a concatenation of the $\boldsymbol{x}_n^{\mu,i}$ and $\boldsymbol{x}_n^{\sigma,i}$ vectors for $i=1,\ldots,K$. Yielding a $2KD$-dimensional representation of the initial matrix $\boldsymbol{Z}_n$.

Assuming that the same procedure is performed for each descriptor, the concatenation of the resulting vectors generates the feature set $\{\boldsymbol{x}_n{\in}\mathbb{R}^{2K(D_1+\cdots+D_F)}, y_n{\in}\mathbb{N}\}_n^N$. Afterwards, a *Centered Kernel Alignment* (CKA) approach is performed to compute a linear projection matrix, and to determine the relevance weight from each trajectory-aligned descriptor individual feature, as follows [11]:

Let $\kappa_X{:}\mathbb{R}^S{\times}\mathbb{R}^S \to \mathbb{R}$, where $S{=}2K(D_1{+}\cdots{+}D_F)$, be a positive definite kernel function, which reflects an implicit mapping $\phi{:}\mathbb{R}^S \to \mathcal{H}_X$, associating an element $\boldsymbol{x}_n{\in}\mathbb{R}^S$ with the element $\phi(\boldsymbol{x}_n){\in}\mathcal{H}_X$, that belongs to the Reproducing Kernel Hilbert Space (RKHS), $\mathcal{H}_X$. In particular, the Gaussian kernel is preferred since it seeks an RKHS with universal approximation capability, as follows [20, 21]:

$$\kappa_X(\boldsymbol{x}_n, \boldsymbol{x}_{n'}; \sigma) = \exp\left(-\upsilon^2(\boldsymbol{x}_n, \boldsymbol{x}_{n'})/2\sigma^2\right);\ n, n'{\in}\{1, 2, \ldots, N\}, \tag{2.5}$$

where $\upsilon(\cdot,\cdot){:}\mathbb{R}^S{\times}\mathbb{R}^S \to \mathbb{R}$ is a distance function in the input space, and $\sigma{\in}\mathbb{R}^+$ is the kernel bandwidth that rules the observation window within the assessed similarity metric. Likewise, for the output labels space $\mathcal{L}{\in}\mathbb{N}$, we also set a positive definite kernel $\kappa_L{:}\mathcal{L}{\times}\mathcal{L} \to \mathcal{H}_L$. In this case, the pairwise similarity distance between samples is defined as $\kappa_L(y_n, y_{n'}){=}\delta(y_n - y_{n'})$, being $\delta(\cdot)$ the Dirac delta function. Each of the above defined kernels reflects a different notion of similarity and represents the elements of the matrices $\mathbf{K}_X, \mathbf{K}_L{\in}\mathbb{R}^{N\times N}$, respectively. In turn, to evaluate how well the kernel matrix $\mathbf{K}_X$ matches the target $\mathbf{K}_L$, we use the statistical alignment between those two kernel matrices as [11]:

$$\hat{\rho}(\mathbf{K}_X, \mathbf{K}_L) = \frac{\langle \bar{\mathbf{K}}_X, \bar{\mathbf{K}}_L \rangle_{\mathrm{F}}}{\sqrt{\langle \bar{\mathbf{K}}_X \bar{\mathbf{K}}_X \rangle_{\mathrm{F}} \langle \bar{\mathbf{K}}_L \bar{\mathbf{K}}_L \rangle_{\mathrm{F}}}}, \tag{2.6}$$

where the notation $\bar{\mathbf{K}}$ stands for the centered kernel matrix calculated as $\bar{\mathbf{K}} = \tilde{\boldsymbol{I}}\mathbf{K}\tilde{\boldsymbol{I}}$, being $\tilde{\boldsymbol{I}} = \boldsymbol{I} - \mathbf{1}^\top\mathbf{1}/N$ the empirical centering matrix, $\boldsymbol{I}{\in}\mathbb{R}^{N\times N}$ is the identity matrix, and $\mathbf{1}{\in}\mathbb{R}^N$ is the ones vector. The notation $\langle\cdot,\cdot\rangle_{\mathrm{F}}$ represents the matrix-based Frobenius norm. Hence, Eq. (2.6) is a data driven estimator that allows to quantify the similarity between the input feature space and the output label space [11]. In particular, for the Gaussian kernel κ_X, the Mahalanobis distance is selected to perform the pairwise comparison between samples:

$$\upsilon_{\boldsymbol{A}}^2(\boldsymbol{x}_n, \boldsymbol{x}_{n'}) = (\boldsymbol{x}_n - \boldsymbol{x}_{n'})\boldsymbol{A}\boldsymbol{A}^\top(\boldsymbol{x}_n - \boldsymbol{x}_{n'})^\top,\ n, n' \in \{1, 2, \ldots, N\}, \tag{2.7}$$

where the matrix $\boldsymbol{A}{\in}\mathbb{R}^{S\times P}$ holds the linear projection in the form $w_n{=}\boldsymbol{x}_n\boldsymbol{A}$, with $w_n{\in}\mathbb{R}^P$, and being $\boldsymbol{A}\boldsymbol{A}^\top$ the corresponding inverse covariance matrix in Eq. (2.7), assuming $P \leq S$.

Therefore, intending to compute the projection matrix $\boldsymbol{A}$, the formulation of a CKA-based optimizing function can be integrated into the following kernel-based learning algorithm:

$$\hat{\boldsymbol{A}} = \arg\max_{\boldsymbol{A}} \log\left(\hat{\rho}(\mathbf{K}_X(\boldsymbol{A};\sigma), \mathbf{K}_L)\right), \tag{2.8}$$

where the logarithm function is employed for mathematical convenience. The optimization problem from Eq. (2.8) is solved using a recursive solution based on the well-known gradient descent approach. After the estimation of the projection matrix $\hat{\boldsymbol{A}}$, we assess the relevance of the S input features. To this end, the most contributing features are assumed to have the higher values of similarity relationship with the provided output labels. Specifically, the CKA-based relevance analysis calculates the relevance vector index $\varrho \in \mathbb{R}^S$, holding elements $\varrho_s \in \mathbb{R}^+$ that allows to measure the contribution from each of the s-th input features in building the projection matrix $\hat{\boldsymbol{A}}$. Hence, to calculate those elements, a stochastic measure of variability is utilized as follows: $\varrho_s = \mathbb{E}_P\{|a_{s,p}|\}$; where $p \in \{1, 2, \dots P\}$, $s \in \{1, \dots, S\}$, and $a_{s,p} \in \hat{\boldsymbol{A}}$.

2.2 Experimental set-up

Database. To test our *video-based human action recognition using kernel relevance analysis* (HARK), we employ the UCF50 database [22]. This database contains realistic videos taken from Youtube, with large variations in camera motion, object appearance and pose, illumination conditions, scale, etc. For concrete testing, we use $N = 5967$ videos concerning the 46 human action categories in which the human bounding box file was available [2]. The video frames size is 320×240 pixels, and the length varies from around 70-200 frames. The dataset is divided into 25 predefined groups. Following the standard procedure, we perform a leave-one-group-out cross-validation scheme and report the average classification accuracy overall 25 folds.

HARK training. Initially, for each video sample in the dataset we employ the Improved Dense Trajectory feature estimation technique (iDT), with the code provided by the authors in [2], keeping the default parameter settings to extract $F = 5$ different descriptors: Trajectory (x, y normalized positions along 15 frames), HOG, HOF, MBHx, MBHy. The iDT technique is an improved version of the previously realized Dense Trajectory technique from the same author, which removes the trajectories generated by the camera motion and the inconsistent matches due to humans. The human detection plays an important role in this technique, because people in action datasets appear in many different poses, and can be only partially visible due to occlusion or out-of-scene. These five descriptors are extracted along all valid trajectories and the resulting dimensionality D_f is 30 for the trajectory, 96 for HOG, MBHx and MBHy, and 108 for HOF.

We then randomly select a subsample of $5000{\times}K$ trajectories from the training set to estimate a GMM codebook with $K = 256$ Gaussians, and the FV encoding is performed as explained in Section 2.1. Afterwards, we apply to the resulting vector a Power Normalization (PN) followed by the L2-Normalization ($||\text{sign}(x)|x|^{\alpha}||$, where $0 \leq a \leq 1$ is the normalization parameter). The above procedure is performed per descriptor, fixing $\alpha = 0.1$. Next, all five normalized FV representations are concatenated together, yielding $S = 218112$ encoding dimension. The linear projection matrix $\hat{\boldsymbol{A}} \in \mathbb{R}^{S \times P}$ and the relevance vector index $\boldsymbol{\varrho} \in \mathbb{R}^{S}$ are computed as explained in section Section 2.1. P=104.8, is the average dimensions through 25 leave-one-out iterations, obtained from those features whose eigenvalues are larger than the mean of them all.

For the classification step, we use a one-vs-all Linear SVM with regularization parameter equal to 1. Fig. 2.1 summarizes the HARK pipeline.

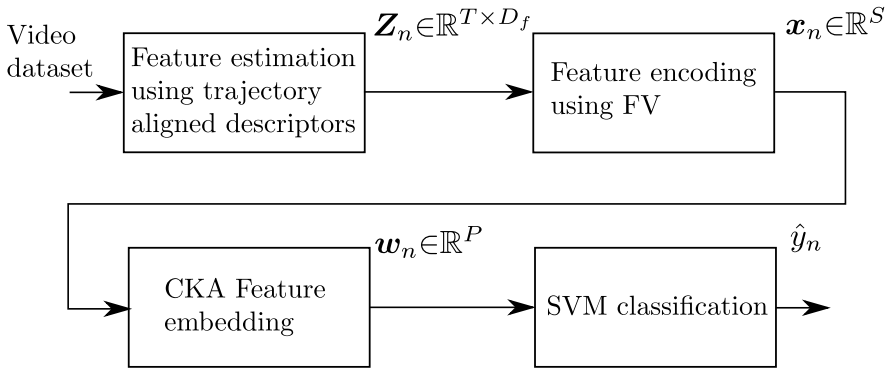


Figure 2.1: Sketch of the proposed HARK-based action recognition system.

2.3 Results and discussions

Fig. 2.2, shows a visual example of feature estimation and assignments to a GMM. Different colors represent memberships to different Gaussians. Also, the sample sizes represent the scale in which the trajectory was generated. It is worth noting that due to human segmentation, the encoding points are mainly grouped around the player whereabouts, which constrains the zone of interest. This strategy helps reducing uncertainty from the video representation, as the background influence is decreased.

Fig. 2.3a shows the normalized relevance value from the provided descriptors. This figure is generated by averaging each descriptor individual component from $\boldsymbol{\varrho} \in \mathbb{R}^{S}$. As seen, the HOG descriptor exhibits the highest relevance value regarding our HARK criteria. This descriptor quantifies the local appearance and shape within the trajectory-aligned window,

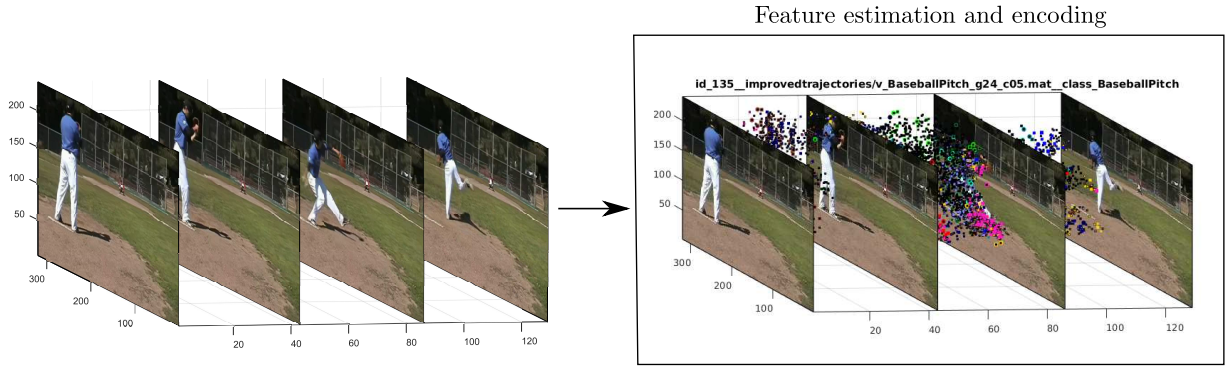


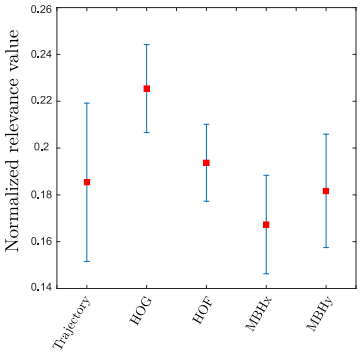
Figure 2.2: Feature estimation and encoding using trajectory-aligned descriptors and BOW.

through the distribution of intensity gradients. Notably, all the others descriptors mainly quantify human local motion (Trajectory normalized positions, HOF, MBHx, MBHy), and are represented closely regarding their relevance value. Trajectory-aligned descriptors match similarly human actions labels concerning the CKA criteria. Therefore, the employed local measures of appearance, shape, and motion are equally important to support action recognition. The relevance value in Fig. 2.3a mainly depends upon the discrimination capability of the Gaussian kernel in Eq. 2.5, and the local measure being performed by the descriptor.

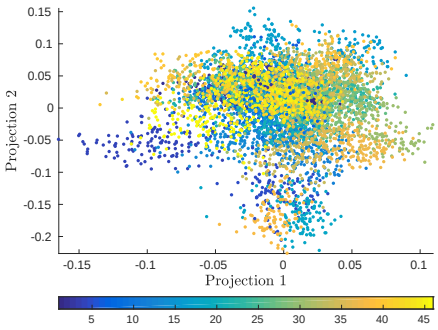
As seen in Fig. 2.3b, the CKA embedding space in its first two projections provides an insight into data overlapping. This situation can be attributed to similar intra-class variations in several categories. Videos in realistic scenarios have inherent attributes such as background clutter, scale changes, dynamic viewpoint and sudden motion, that are affecting adversely the class separability.

Fig. 2.3c shows a confusion matrix using a linear SVM over the CKA feature embedding set. The proposal obtained $87.92 \pm 2.94\%$ of accuracy while classifying human actions. From the matrix, classes 22 and 23 are similar, because the system had trouble classifying between them. These classes correspond to Nunchucks and Pizza tossing respectively. Notably, our approach requires only 104.8 dimensions on average to classify the 46 human actions from the UCF50 dataset, with competitive accuracy.

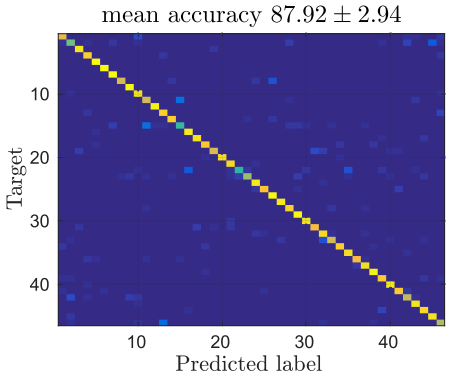
In turn, Tab. 2.1 presents a comparative study of the results achieved by HARK and other similar approaches for human action recognition on the UCF50 database. To build this comparative analysis, approaches with similar experimental set-up are employed. Specifically, those approaches using iDT representation and similar descriptors. Primarily, the compared results exhibit a trade-off between data dimension and accuracy. More elaborate procedures such as the one presented in [1], uses Time Convolutional Networks (TCN) and Spatial Convolutional Networks (SCN) descriptors along with iDT descriptors, and Spatio-temporal VLAD (ST-VLAD) encoding to enhance the class separability. Thus, the mentioned approach



(a) Descriptor relevance values



(b) 2D input data projection using CKA



(c) Confusion matrix using linear SVM

Figure 2.3: Human action recognition on the UCF50 database

obtain very high mean accuracy 97.7%. However, the data dimensionality is considerably high as well. On the other hand, the approach presented in [2], enhances the spatial resolution of the iDT descriptors by using a strategy called spatiotemporal pyramids (STP) along with Spatial Fisher Vector encoding (SFV). Obtained results regarding the accuracy of [2] are comparable to ours. Nonetheless, the data dimension is drastically higher.

2.4 Conclusions

We introduced a video-based human action recognition framework using kernel relevance analysis (HARK). Our approach highlights the primary descriptors to predict the output

Table 2.1: Comparison with similar approaches in the state-of-the-art on the UCF50 dataset.

Reference	**Descriptors**	**Feature encoding**	**Data dimension**	**Classification method**	**Accuracy** [%]
J. Uijlings *et al* [9].	HOG+HOF+MBHx+MBHy	FV	36,864	Linear SVM	81.8
H. Wang *et al* [2].	HOG+HOF+MBHx+MBHy	SFV + STP	811,008	Linear SVM	91.7
I. C. Duta *et al* [1].	HOG+HOF+MBHx+MBHy+SCN+TCN	ST-VLAD	258,816	Linear SVM	97.7
HARK	Traj+ HOG+HOF+MBHx+MBHy	FV + CKA	**104.8**	Linear SVM	**87.9**

labels of human action videos using trajectory-based representations. Therefore, HARK quantifies the relevance of $F = 5$ trajectory-aligned descriptors towards a CKA-based algorithm, that matches the input space with the output labels, to enhance the descriptor interpretability, as it allows to determine the importance of local measures (appearance, shape, and motion) to support action recognition. Also, the CKA-algorithm allows to compute a linear projection matrix, through a non-linear representation, where the columns represent those features whose eigenvalues are larger than the mean of them all. Hence, by projecting the video samples into the generated CKA space, the class separability is preserved, and the number of dimensions is reduced. Attained results on the UCF50 database show that our proposal correctly classified the 87.92% of human actions samples using an average input data dimension of 104.8 in the classification stage, through 25 folds under a leave-one-group-out cross-validation scheme. In particular, according to the performed relevance analysis, the most relevant descriptor is the HOG which quantifies the local appearance and shape through the distribution of intensity gradients. Remarkable, HARK outperforms state-of-art results concerning the trade-off between the accuracy achieved and the required data dimension (Tab. 2.1). As future work, authors plan to employ other descriptors such as the deep features presented in [1]. Also, a HARK improvement based on the enhancement of spatial and temporal resolution, as the one presented in [2], could be an exciting research line.

Chapter 3

Data encoding framework using Bayesian inference probabilistic models to support video-based Human Action Recognition

We introduce a novel data encoding framework using Infinite Gaussian Mixture Models (IGMMs), to extend the conventional Fisher Vector encoding technique. The methodology, based on Bayesian inference and Dirichlet processes seeks to reveal a set of discriminant local spatio-temporal features that enable the precise codification of visual information from Human Action Recognition tasks (HAR). Specifically, it is much simple to handle the infinite limit from the IGMM, than working with traditional Gaussian Mixture Models with unknown sizes, that will require extensive crossvalidation. Under this premise, we developed a fully automatic video data encoding methodology for HAR that avoids the need of specifying the number of Gaussians in the mixture model. This parameter is known to greatly affect the recognition performance in many Computer Vision tasks, and its inference with conventional methods implies high computational burden. In fact, the Markov Chain Monte Carlo implementation of the hierarchical IGMM effectively avoids local minima which tend to plague mixtures trained by optimization based methods. Attained results on the UCF50 database demonstrate that our proposal favors data representation obtaining promising recognition results according to supervised classification measures, without the need of exhaustive search.

3.1 Infinite Gaussian Mixture Model

As expressed previously, we have the input-output pair set $\{\boldsymbol{Z}_n \in \mathbb{R}^{T\times D}, y_n \in \mathbb{N}\}_{n=1}^{N}$, holding N described video samples. In this chapter, we aim to train a generative model using IGMM of the descriptor in turn from feature matrix $\{\boldsymbol{Z}_n\}_{n=1}^{N}$. Then, with the obtained codebook, we perform the FV feature encoding as presented in the previous chapter. The IGMM model is defined as follows [23]:

The complete feature matrix $\boldsymbol{Z} \in \mathbb{R}^{NT\times D}$ holds $NT = N * T$ described trajectories, $\boldsymbol{z}_t \in \mathbb{R}^D$. Let the complete likelihood from sample $\boldsymbol{z}_t$ to a Gaussian mixture model with k_{rep} components, be:

$$p(\boldsymbol{z}_t, \boldsymbol{c}_t | \{\boldsymbol{\mu}_j, \boldsymbol{S}_j, \pi_j\}_{j=1}^{k}) = \sum_{j=1}^{k_{\text{rep}}} \left[\pi_j \mathcal{N}(\boldsymbol{\mu}_j, \boldsymbol{S}_j^{-1})\right]^{c_{t,j}} \tag{3.1}$$

where $\boldsymbol{\mu}_j \in \mathbb{R}^D$ are mean vectors, $\boldsymbol{S}_j \in \mathbb{R}^{D\times D}$ are precision matrices, and π_j are the mixing proportions. Variable k_{rep} is number of classes that have data associated with them in the current iteration (represented classes) [24]. On the other hand, $\boldsymbol{c}_t \in \mathbb{R}^k$ is a binary latent variable with notation 1 of k, where k is not limited to represented classes.

In Bayesian inference priors and posteriors on component parameters and hyper-parameters are required for Gibbs sampling, a Markov Chain Monte Carlo method (MCMC). Generally, priors are chosen according to their modeling properties and mathematical convenience (conjugate priors) [25].

3.1.1 Component parameters

The component means $\boldsymbol{\mu}_j$, are given by Gaussian priors:

$$p(\boldsymbol{\mu}_j | \boldsymbol{\lambda}, \boldsymbol{R}) \sim \mathcal{N}(\boldsymbol{\lambda}, \boldsymbol{R}^{-1}) \tag{3.2}$$

Whose mean $\boldsymbol{\lambda} \in \mathbb{R}^D$, and precision $\boldsymbol{R} \in \mathbb{R}^{D\times D}$, are hyper-parameters common to all components. The conditional posterior on means are obtained by multiplying the likelihood from Eq. 3.1 and the prior from Eq. 3.2:

$$p(\boldsymbol{\mu}_j|\boldsymbol{\lambda},\boldsymbol{R},\{\boldsymbol{z}_t : c_{t,j}=1\},\boldsymbol{S}_j) \propto \prod_{t:c_{t,j}=1} p(\boldsymbol{z}_t|\boldsymbol{\mu}_j,\boldsymbol{S}_j) \times p(\boldsymbol{\mu}_j|\boldsymbol{\lambda},\boldsymbol{R})$$
$$= \mathcal{N}\left(\frac{n_j\,\overline{\boldsymbol{z}}_j\,\boldsymbol{S}_j + \boldsymbol{\lambda}\boldsymbol{R}}{n_j\,\boldsymbol{S}_j + \boldsymbol{R}}, \frac{1}{n_j\,\boldsymbol{S}_j + \boldsymbol{R}}\right) \tag{3.3}$$

where, n_j is the occupation number, defined as the number of observations belonging to class j. $\overline{\boldsymbol{z}}_j$ is the average of these observations.

$$\overline{\boldsymbol{z}}_j = \frac{1}{n_j}\sum_{t:c_{t,j}=1} \boldsymbol{z}_t, \quad n_j = \sum_{t=1}^{NT} c_{t,j} \tag{3.4}$$

The component precisions $\boldsymbol{S}_j$, are given by Wishart priors:

$$p(\boldsymbol{S}_j|\beta,\boldsymbol{W}) \sim \mathcal{W}(\beta,\boldsymbol{W}^{-1}) \tag{3.5}$$

where $\boldsymbol{W}\in\mathbb{R}^{D\times D}$ is a precision matrix, and β the degrees of freedom. Both hyperparameters are common to all components.

The Wishart density for the precisions is given by:

$$p(\boldsymbol{S}_j|\beta,\boldsymbol{W}) = \frac{|\boldsymbol{S}_j|^{\frac{\beta-D-1}{2}}\,|\boldsymbol{W}|^{\beta/2}\exp\{-\frac{1}{2}\,\beta\,\mathbf{tr}(\boldsymbol{\Sigma}_j\boldsymbol{W})\}}{(\frac{2}{\beta})^{\frac{\beta\,D}{2}}\;\Gamma_D(\frac{\beta}{2})}, \tag{3.6}$$

where,

$$\Gamma_D(\frac{\beta}{2}) = \pi^{\frac{D(D-1)}{4}}\prod_{d=1}^{D}\Gamma\left(\frac{\beta-(j-1)}{2}\right). \tag{3.7}$$

The conditional posterior on precisions are obtained by multiplying the likelihood from Eq. 3.1 and the prior from Eq. 3.5:

$$p(\boldsymbol{S}_j|\beta,\boldsymbol{W},\{\boldsymbol{z}_t : c_{t,j}=1\},\boldsymbol{\mu}_j) \propto \prod_{t:c_{t,j}=1} p(\boldsymbol{z}_t|\boldsymbol{\mu}_j,\boldsymbol{S}_j) \times p(\boldsymbol{S}_j|\beta,\boldsymbol{W})$$
$$= \mathcal{W}(\beta+n_j, [\frac{1}{\beta+n_j}(\beta\,\boldsymbol{W}+\boldsymbol{A})]^{-1}) \tag{3.8}$$

where

$$\boldsymbol{A} = \sum_{t:c_{t,j}=1} (\boldsymbol{z}_t - \boldsymbol{\mu}_j)(\boldsymbol{z}_t - \boldsymbol{\mu}_j)^\top \tag{3.9}$$

3.1.2 Hyper-parameters

For the mean hyper-parameters $\boldsymbol{\lambda}, \boldsymbol{R}$, and the precision hyper-parameter $\boldsymbol{W}$, the priors are defined as follows:

$$p(\boldsymbol{\lambda}) \sim \mathcal{N}(\boldsymbol{\mu}_Z, \mathbf{cov}_Z) \quad p(\boldsymbol{R}) \sim \mathcal{W}(1, \mathbf{cov}_Z^{-1}) \quad p(\boldsymbol{W}) \sim \mathcal{W}(1, \mathbf{cov}_Z) \tag{3.10}$$

where $\boldsymbol{\mu}_Z \in \mathbb{R}^D$ and $\mathbf{cov}_Z \in \mathbb{R}^{D \times D}$, are respectively the mean and covariance of the observations. The degrees of freedom that are set to unity, correspond to a very broad distribution. In fact, a wide variety of reasonable priors will lead to similar results. Following the procedure exposed in Sec. 3.1.1, the posterior distributions on hyper-parameters are obtained straight forward using the mean prior, Eq.3.2, and the precision prior, Eq.3.5, as likelihoods in each case.

It is of interest the parameter β, which remains scalar after conjugacy. According to Rasmussen in [26], beta has gamma prior of the form:

$$g = \beta - D + 1 \tag{3.11}$$

$$p(g^{-1}) \sim \mathcal{G}(1, \frac{1}{D}) \quad \rightarrow \quad p(g) \propto g^{-\frac{3}{2}} \exp\{-\frac{D}{2\,g}\} \tag{3.12}$$

For the parameter β, the precision prior, Eq.3.5, plays the role of likelihood. Thus, the posterior distribution takes the form:

$$p(g|\{\boldsymbol{S}_j\}_{j=1}^{k_{\text{rep}}}, \boldsymbol{W}) \propto \prod_{j=1}^{k_{\text{rep}}} p(\boldsymbol{S}_j|\beta, \boldsymbol{W}) \times p(g) \tag{3.13}$$

$$\propto (\frac{\beta}{2})^{\frac{k_{\text{rep}}\,\beta\,D}{2}}\, g^{-\frac{3}{2}}\, \Gamma_D(\frac{\beta}{2})^{-k_{\text{rep}}}\, \exp\{-\frac{D}{2\,g}\} \prod_{j=1}^{k_{\text{rep}}} |\boldsymbol{W}\,\boldsymbol{S}_j|^{\frac{\beta}{2}} \exp\{-\frac{1}{2}\,\beta\,\mathbf{tr}(\boldsymbol{W}\,\boldsymbol{S}_j)\} \tag{3.14}$$

The later density does not have standard form. However, the distribution $p(\log(g|\{\boldsymbol{S}_j\}_{j=1}^{k_{\text{rep}}}, \boldsymbol{W})$ is log-concave, so we generate independent samples using the Adaptive Rejection Sampling technique (ARS), and transform these to get values of β.

3.1.3 Mixing proportions and latent variables

In this section, the number of components k, is not limited to represented classes. Thus, for the mixing proportions π_j, the prior is a symmetric Dirichlet distribution with concentration parameter α/k,

$$p(\{\pi_j\}_{j=1}^k|\alpha) \sim \text{Dir}(\{\alpha/k\}_{j=1}^k) = \frac{\Gamma(\alpha)}{\Gamma(\alpha/k)^k} \prod_{j=1}^k \pi_j^{\alpha/k-1}, \tag{3.15}$$

where $\Gamma(\cdot)$ is the gamma function, and the mixing proportions must be positive and sum to one. Given the mixing proportions, the joint distribution for the latent variable $\boldsymbol{c}_t$ is:

$$p(\{c_{t,j}\}_{j=1}^k|\{\pi\}_{j=1}^k) = \prod_{j=1}^k \pi_j^{c_{t,j}}, \quad \{\forall t : \prod_{j=1}^k \pi_j^{n_j}\}. \tag{3.16}$$

Using the Dirichlet integral type I, we marginalize the mixing proportions and write the prior directly in terms of the latent variable.

$$\begin{aligned} p(\{c_j\}_{j=1}^k|\alpha) &= \int p(\{c_j\}_{j=1}^k|\{\pi_j\}_{j=1}^k)\ p(\{\pi_j\}_{j=1}^k) d\pi_1 \cdots d\pi_k \\ &= \frac{\Gamma(\alpha)}{\Gamma(\alpha/k)^k} \prod_{j=1}^k \frac{\Gamma(n_j + \alpha/k)}{\Gamma(\alpha/k)}. \end{aligned} \tag{3.17}$$

To use Gibbs sampling for the latent variable $\boldsymbol{c}_t$, we use the conditional prior for a single indicator given all the others. This is obtained from Eq 3.17, keeping all but a single indicator fixed:

$$p(c_{t,j} = 1|\boldsymbol{c}_{-t}, \alpha) = \frac{n_{-t,j} + \alpha/k}{NT - 1 + \alpha}, \tag{3.18}$$

where the subscript $-t$ indicates all the indexes except t and $n_{-t,j}$ is the number of observations, excluding $\boldsymbol{z}_t$, that are associated with component j.

In the limit where $k \to \infty$, the conditional prior is:

$$\begin{aligned} \text{Components where } n_{-t,j} > 0: \quad p(c_{t,j} = 1 | \boldsymbol{c}_{-i}, \alpha) &= \frac{n_{-i,j}}{NT - 1 + \alpha}, \\ \text{else:} \qquad p(\boldsymbol{c}_t \neq \boldsymbol{c}_{t'}, \{\forall t \neq t'\} | \boldsymbol{c}_{-t}, \alpha) &= \frac{\alpha}{NT - 1 + \alpha}. \end{aligned} \tag{3.19}$$

The tractability of integral 3.17, allow us to work with the finite number of latent variables, rather than the infinite number of mixing proportions. The posterior is obtained by multiplying the complete likelihood from $\boldsymbol{z}_t$, Eq. 3.1, and the latent variables prior, Eq. 3.19:

$$\begin{aligned} \text{Components where } n_{-t,j} > 0: \quad & p(c_{t,j} = 1 | \boldsymbol{c}_{-i}, \boldsymbol{\mu}_j, \boldsymbol{S}_j, \alpha) \\ & \propto \frac{n_{-i,j}}{NT - 1 + \alpha} \, |\boldsymbol{S}_j|^{\frac{1}{2}} \exp\{-\frac{1}{2} (\boldsymbol{z}_t - \boldsymbol{\mu}_j) \, \boldsymbol{S} \, (\boldsymbol{z}_t - \boldsymbol{\mu}_j)^\top\}, \end{aligned} \tag{3.20}$$

$$\begin{aligned} \text{else:} \quad & p(\boldsymbol{c}_t \neq \boldsymbol{c}_{t'}, \{\forall t \neq t'\} | \boldsymbol{c}_{-t}, \boldsymbol{\lambda}, \boldsymbol{R}, \beta, \boldsymbol{W}, \alpha) \\ & \propto \frac{\alpha}{NT - 1 + \alpha} \int p(\boldsymbol{z}_t | \boldsymbol{\mu}_j, \boldsymbol{S}_j) \, p(\boldsymbol{\mu}_j, \boldsymbol{S}_j | \boldsymbol{\lambda}, \boldsymbol{R}, \beta, \boldsymbol{W}) \, d\boldsymbol{\mu}_j \, d\boldsymbol{S}_j. \end{aligned} \tag{3.21}$$

The likelihood for components with observations other than $\boldsymbol{z}_t$ is Gaussian with parameters $\boldsymbol{\mu}_j$ and $\boldsymbol{S}_j$. The likelihood for currently unrepresented classes, with no parameters yet, is obtain through marginalization of their prior distribution. This integral is not analytically tractable. Thus, following the procedure from [27], the parameters for unrepresented classes are obtained by sampling from the priors. When an unrepresented class is chosen, a new class is introduced to the model. Likewise, when a class becomes empty, the class is removed from the model.

3.2 Experimental set-up

Database. To test our *Infinite Gaussian Fisher Vector encoding to support Human Action Recognition* (IGFV), we employ the UCF50 database [22]. For concrete testing, we use $N = 5967$ videos concerning the 46 human action categories in which the human bounding box file was available [2]. The video frames size is 320×240 pixels, and the length varies

from around 70-200 frames. The dataset is divided into 25 predefined groups. Following the standard procedure, we perform a leave-one-group-out cross-validation scheme and report the average classification accuracy overall 25 folds.

IGFV training. Initially, for each video sample in the dataset we employ the Improved Dense Trajectory feature estimation technique (iDT), with the code provided by the authors in [2]. we extract $F = 4$ different trajectory aligned descriptors: HOG, HOF, MBHx, MBHy. The trajectory descriptor was removed from the experiments in this chapter because of its large variation in relevance value, this result was obtained in chapter 2. All descriptors are extracted along all valid trajectories and the resulting dimensionality D_f is 96 for HOG, MBHx and MBHy, and 108 for HOF.

Initially, we randomly select a subsample of $1000 \times N_c$ trajectories from the training set, being N_c the number of classes. The samples are represented by their most relevant features according to a variability criterion (PCA preserving 90% of data variance), and then divided into groups using the spatio-temporal pyramid technique. For each spatio-temporal cell, we estimate a IGMM codebook (see Sec.3.1). The model is started with a single component, then 300 Gibbs iterations are performed for updating all parameters and hyper-parameters from their conditional posterior distribution (with 250 "burn in iterations"). From repetitions, the employed mixture model is chosen according to the BIC model selection criterion. Afterwards, the FV encoding is performed using the conventional algorithm. To the resulting super-vector, we apply a Power Normalization (PN) followed by the L2-Normalization. The above procedure is performed per descriptor. Next, all five normalized IGFV representations are concatenated together.

For the classification step, we use a one-vs-all Linear SVM with regularization parameter equal to 1. Fig. 2.1 summarizes the HARK training pipeline. It is worth noting that the feature extraction was performed in C++, and the remaining experiments in the Matlab software.

3.3 Results and discussions

Fig. 3.2, shows an example of data distribution estimation using IGMM. The figures from column one, Fig. 3.2a and Fig. 3.2c, were obtained using the conventional IGMM formulation for representing both Gaussian and Non-Gaussian distributed data. As seen, for Non-Gaussian data, the model estimates its distribution through a large number of components. Moreover, when the model is restricted to diagonal precisions, Fig. 3.2b and Fig. 3.2d, it requires an even larger amount of components, as correlated data can not be completely represented with only one diagonal component. This situation is of interest, as in many HAR tasks, challenges such as occlusions and partially out-of-scene humans can be mitigated by enhancing

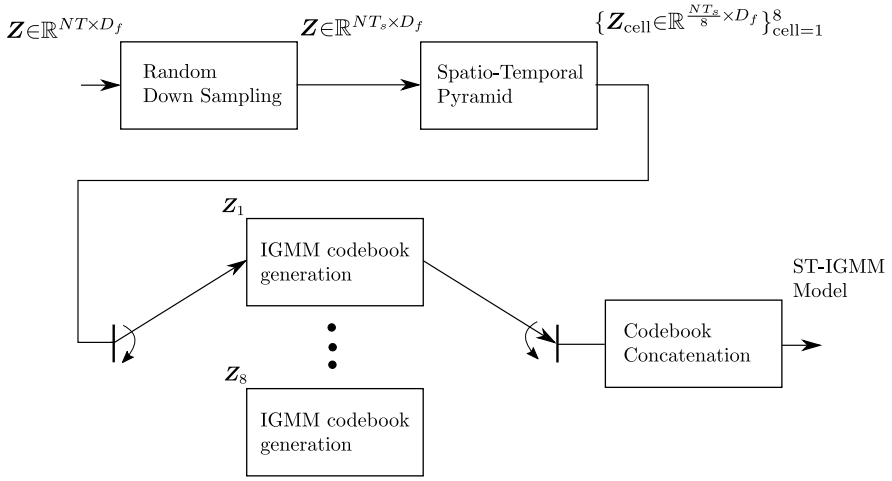
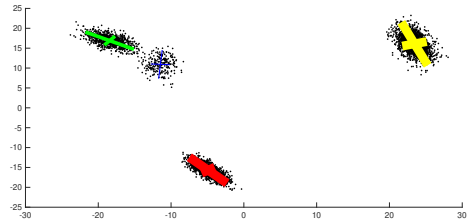


Figure 3.1: Sketch of the proposed IGFV data encoding technique.

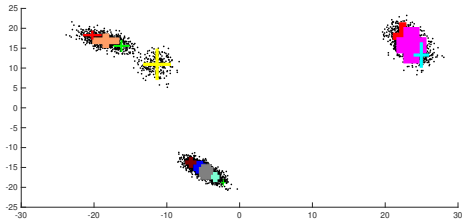
the model resolution, which in this case, is performed by increasing the number of Gaussian components.

Fig. 2.3c shows a confusion matrix using a linear SVM over the IGFV encoding representation. The proposal obtained $85.54 \pm 4.07\%$ of mean accuracy while classifying human actions. From the matrix, again classes 22 and 23 are similar, because the system had trouble classifying between them. These classes correspond to Nunchucks and Pizza tossing respectively. Notably, this approach obtained a similar result to the presented in chapter 2, while avoiding the need of exhaustively searching for the number of Gaussian components.

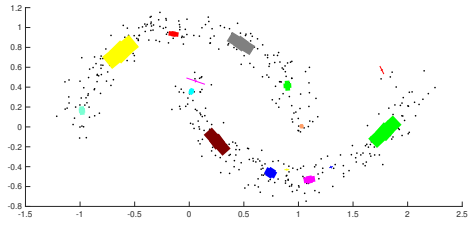
In turn, Tab. 3.1 presents a comparative study among the result obtained using our IGFV encoding and other similar approaches for human action recognition on the UCF50 database. As seen, our approach favors HAR tasks by providing a fully automatic system that drastically reduces experimental setup. The proposed framework decreases memory requirements and number of operations, as within one crossvalidation run, the system is completely determined. Meanwhile, conventional approaches require multiple crossvalidation runs for determining the number of components parameter. The obtained recognition performance is comparable to those presented in the state-of-the art. Acknowledging that the best result [1], uses Time Convolutional Networks (TCN) and Spatial Convolutional Networks (SCN) descriptors, which were not considered in this analysis. On the other hand, the approach presented in [2], required the overwhelming largest dimensional representation for obtaining their performance. When our data encoding IGFV, is employed together with the kernel-based relevance analysis presented in Chap. 2, the resulting data representation slightly reduces is precision, but the reduction in dimension length is considerable.



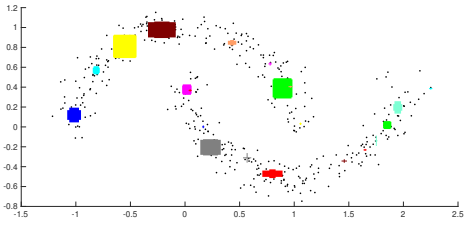
(a) IGMM on Gaussian distributed data



(b) DIGMM on Gaussian distributed data



(c) IGMM on Non-Gaussian distributed data



(d) DIGMM on Non-Guassian distributed data

Figure 3.2: Data distribution estimation

Table 3.1: Comparison with similar approaches in the state-of-the-art on the UCF50 dataset.

Reference	**Descriptors**	**Feature encoding**	**Data dimension**	**Accuracy** [%]
J. Uijlings *et al* [9].	HOG+HOF+MBHx+MBHy	FV	36,864	81.8
H. Wang *et al* [2].	HOG+HOF+MBHx+MBHy	SFV + STP	811,008	91.7
I. C. Duta *et al* [1].	HOG+HOF+MBHx+MBHy+SCN+TCN	ST-VLAD	258,816	97.7
HARK-IGFV	Traj+ HOG+HOF+MBHx+MBHy	IGFV	**56,197**	**85.5**
HARK-IGFV	HOG+HOF+MBHx+MBHy	IGFV + CKA	**2,022**	**83.3**

3.4 Conclusions

We introduced a novel Infinite Gaussian Fisher Vector data encoding framework to support video-based Human Action Recognition (IGFV). Our approach is fully automatic, allowing every parameter in the model to be updated hierarchically through a MCMC method, Gibbs sampling. The IGFV encoding allows revealing a set of discriminant local spatio-temporal features that enable the precise codification of visual information from HAR tasks, with promising recognition results. Specifically, the use of MCMC effectively avoids local minima, a problem that all optimization based-methods have. Moreover, the infinite limit on the number of Gaussian components evades the need of inferring this parameter through extensive crossvalidation, which reduces memory requirements and number of operations. Attained results on the UCF50 database show that our proposal correctly classified the 85.5% of human

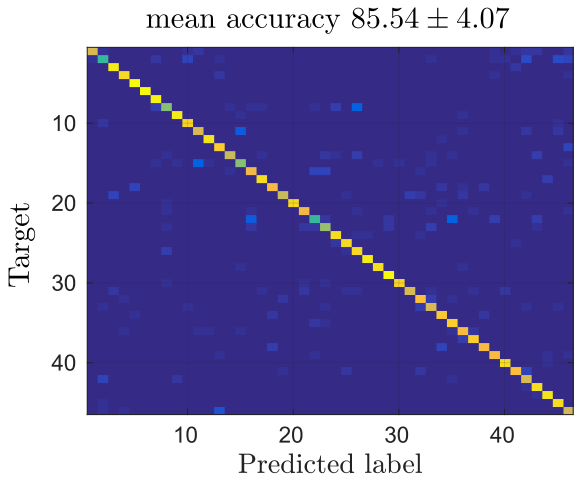


Figure 3.3: Confusion matrix using linear SVM

actions samples, with a IGFV representation of $56,197$ dimensions, through 25 folds under a leave-one-group-out cross-validation scheme. When used with the relevance analysis from Chap. 2, the recognition performance slightly drops 83.3, but the reduction in data dimensions is considerable $2,022$. The obtained dimension correspond to the number of features required to preserve 90% of data variability. Remarkable, our IGFV encoding approach obtained promising results comparable with state-of-art approaches. Furthermore, it drastically reduces the experimental set-up of HAR tasks, by avoiding exhaustive search.

Chapter 4

Feature representation framework based on high-level semantic features and kernel methods

We present a kernel-based relevance analysis for video data to support social behavior recognition. Our approach, termed KRAV, is twofold: (i) A feature ranking based on centered kernel alignment–(CKA) is carried out to match social semantic features with the output labels (individual and group behaviors). The employed method is an extension of the conventional CKA to mitigate the imbalance effect of unusual human behaviors. (ii) A classification stage based on k-nearest neighbors to perform the behavior prediction. For concrete testing, the Israel Institute of Technology social behavior database is employed to assess KRAV under a 10-fold cross-validation scheme. Attained results show that the proposed approach for the individual recognition task obtains 0.7481 $F1$ measure using 21% of the input features. Likewise, for the group recognition task obtains 0.7611 $F1$ measure using 57% of the input features, which in both cases outperforms state-of-the-art results concerning the classification performance and number of employed features. Also, our video-based approach would assist further social behavior analysis from the set of features selected regarding the recognition of individual profiles and group behaviors.

4.0.1 Social behavior dataset

To identify individual and group behavior performed by pedestrians, we employ the IIT (Israel Institute of Technology) dataset that holds several still camera videos recorded in shopping-mall scenarios [28]. Though this social context scenario comprises three videos,

Table 4.1: Israel Institute of Technology dataset statistics for video-based social behavior recognition.

Frames Annotated	Annotation duration	Elapsed Time (IP)	Elapsed Time (GB)	IPs distribution	GBs distribution	Average Individuals per frame	Average Inidividuals per group
80894 (97.3)%	02:22:49 (hh:mm:ss)	203.5 (s) Dist 35.3 (s) Exp 12.8 (s) Int 4.2 (s) Dis	30.7 (s) EI 23 (s) BI 100.3 (s) UI 83.7 (s) CHAT	869 total 45 Dist 776 Exp 41 Int 7 Dis	255 total 193 EI 27 BI 28 UI 7 CHAT	3.5	1.8 (max: 9)

only the annotated one is used that lasts nearly one hour, holding 83155 frames with resolution 512×384 and 25 *fps*. The video displays a pedestrian in, at least, 97.3% of frames and 3.5 persons per frame on average, posing a challenge for activity recognition tasks. As seen in Table 4.1, the annotation procedure[1] labels two cases of interest: individual activities (termed individual profiles - IP) and group activities (group behaviors - GB). Following, the individual profiles and the group behaviors are described:

Individual profiles:

- *Exploring (Exp)*: No specific interest is revealed, but movement and gaze are coherent with the scene structure and context.
- *Interested (Int)*: Explicit interest in a scene object is manifested.
- *Distracted (Dist)*: No specific interest is revealed, resulting in unstructured movement and gaze variability.
- *Disoriented (Dis)*: A mixed interest is disclosed, exposed as high variability of movement and gaze (unstructured flow).

Group behaviors:

- *Equally Interested (EI)*: A group presents coherent behavior as below: i) *interested*: Individuals show interest in the same object; ii) *exploring*, individuals explore the environment with similar gaze-direction and close to each other.

[1]performed by the INESC TEC, Group of Portugal and supervised by the lab of social-psychology of the University of Porto–http://sigarra.up.pt/fpceup

- *Balanced Interests (BI)*: Individuals within a group do not reveal the same level of interest, preserving the same behavior and verifying that they explore the environment with similar gaze-direction, but being slightly separated from each other (that is, IPs are *exploring*).
- *Imbalanced Interests (UI)*: A group reveals, at the same time, distinct behaviors so that there are different individual profiles, varying their gaze and distance among them.
- *Chatting (CH)*: There is a free-standing conversational group (FCG), holding individuals that are talking with each other (moving persons are not considered). By default, all IPs are set to *distracted*.

In the annotation procedure, the ground truth low-level information for pedestrian detection, tracking, and full-oriented gaze direction $[0°, 360°]$, was marked together with the group formation and dispersion, and the scene objects of interest, namely, candy box, toy cars, and electric stairs (see Fig. 4.1b). The above information is projected onto the ground plane, using camera calibration and detection of vanishing lines (see Fig. 4.1a). Afterwards, the data is used to extract individual and group relational features, such as trajectories, position, and attention among individuals, people in groups, and scene objects.

(a) Calibration (b) Groundplane estimation

Figure 4.1: (a) Detected chessboard points for camera calibration; (b) Horizontal vanishing line (blue), ground plane's projection area (green), ground points (red), and objects of interest (purple).

4.0.2 IP recognition

For this one video database, the challenge of detecting transitions between different behaviors arises due to the changing nature of the individual and collective action. This is addressed

by working with video mini-batches under the assumption of constant behavior and sufficient discriminative information. Hence, the IP recognition comprises the following stages:

Feature extraction

Let $\{\boldsymbol{Z}_n \in \mathbb{R}^{H_n \times W_n \times F_n}, l_n \in \mathbb{Z}\}_{n=1}^{N}$ be an input-output pair set holding N pedestrian mini-batches (as explained in Section 4.0.1), each one represented by a bounding box of size $H_n \times W_n$ pixels, captured throughout F_n frames. The output label l_n denotes the social behavior performed on the corresponding video sequence. For IP identification, the ground truth-tracking and gaze information are used to extract the following *Individual features* [4]:

- The *Trajectory* that measures the angular variation of an individual along the frames.
- The *Distance* that quantifies the extent of space between an individual and object of interest from the scene.
- The *Speed* that estimates the instantaneous velocity of an individual.
- The *Gaze* that computes the steady intent look direction.

Features extracted from the mini-batches are encoded into a multi-scale histogram with $R \in \mathbb{N}$ granularity levels. For an arbitrary extracted feature, its multi-scale representation, sizing R, is given by the vector $\boldsymbol{x} = [h^1, \ldots, h^R]$, where each entry $h^r \in \mathbb{R}^{2^{r+1}}$, is a normalized histogram of 2^{r+1} bins, $\forall r = \{1, \ldots, R\}$. Thus, the pedestrian mini-batch descriptor is represented by the fixed-length vector $\boldsymbol{x} = [\boldsymbol{x}^1, \ldots, \boldsymbol{x}^{N_f}]$, where N_f is the number of extracted features. Therefore, the input set is represented by the feature set $\{\boldsymbol{X} \in \mathbb{R}^{N \times P}, \boldsymbol{l} \in \mathbb{Z}^N\}$, where P is the resulting number of bins from the concatenation of all feature histograms. Note that N samples are related to individuals in a shopping mall video and $l_n \in \boldsymbol{l}$.

Relevance analysis extension to mitigate the effect of unusual human behaviors

In practice, the feature matrix $\boldsymbol{X}$ stores a vast number of variables, which would introduce noise and complexity for further IP recognition stages. Thus, following the kernel-based relevance analysis presented in Chap. 2, we select and embed features by training a projection matrix using the *Centered Kernel Alignment* (CKA) approach. However, the human behavior dataset employed in this section exhibit the challenge of data imbalance. This situation arises given the presence of unusual human behavior, to which conventional processing strategies will tend to forsake. Therefore, in this section we introduce an extension of the conventional CKA formulation to mitigate the imbalance effect of unusual human behaviors.

To make the classes with fewer samples more relevant, the kernel matrix $\boldsymbol{K_l} \in \mathbb{R}^{N \times N}$, holding the pairwise similarity between labels $l_n, l_{n'} \in \boldsymbol{l}$ $(n, n' \in [1, N])$, is modified from binary to a weight matrix. Large values are assigned to classes with fewer samples. The positive definite kernel function for the labels κ_L, that measures the pairwise similarity between IP labels, and addresses the class imbalance, is the following:

$$\kappa_L(l, l') = \frac{1}{N_c}\delta(l - l') \tag{4.1}$$

where $\delta(\cdot)$ is the Dirac Delta function and N_c is the number of samples per class.

4.0.3 GB recognition

Since the IIT database provides us with the bounding box of the group formation, we employ it and the IP recognition labels to create an input-output pair set holding all detected groups. In this case, we use the ground truth information about group formation and dispersion, along with individual features and predicted IP labels (see Fig. 4.2) to extract the following *Group features* [4]:

- The *Group's Speed* that measures the average instantaneous velocities of all individuals within a group.
- The *Group's Distance* as the average distance between a pair of individuals, considering all the pair-wise relations within a group.
- The *Speed Variance* that quantifies the variance of instantaneous velocities of all individuals within a group.
- The *Looking At Each Other* that Expresses the minimum angle difference between an individual gaze direction and the displacement vector between him and other individuals. By considering only the people who fall inside each field of view, we determine this measure as the mean square error (MSE) of all the differences.
- *Profiles Information* that codifies all IPs within a group.

Like the IP recognition framework discussed in Section 4.0.2, we concatenate the group features obtained for each provided group to built an input-output pair set holding the given group samples and the target labels (Group Behaviors). Next, the kernel-based relevance analysis approach introduced in Section 4.0.2 is also applied.

Fig. 4.2 shows the proposed video-based social behavior recognition (IP and GB) pipeline based on kernel relevance analysis. Note that the obtained IP labels feed the GB recognition.

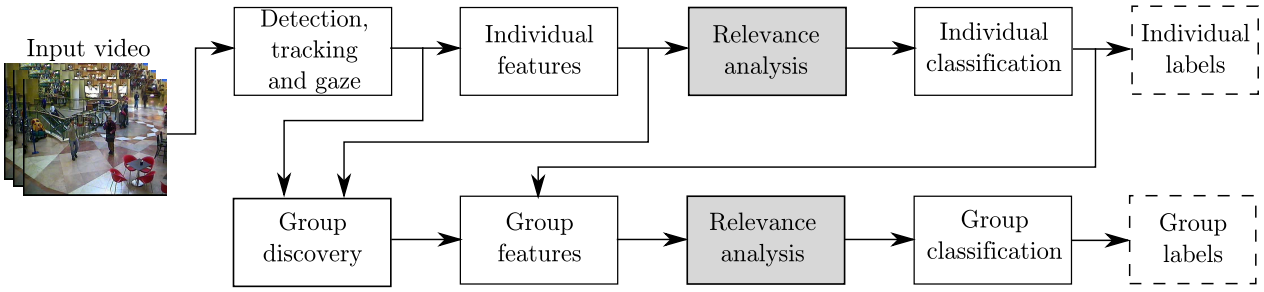


Figure 4.2: Sketch of the proposed methodology for video-based social behavior recognition (IP and GB) utilizing kernel-based relevance analysis.

4.1 Experimental setup

The proposed methodology, termed *Kernel-based Relevance Analysis for Video data* (*KRAV*), is used to obtain the vector $\boldsymbol{\varrho}$ holding the relevance value for every feature in the original space $\boldsymbol{X}$. To this end, we select the human mini-batch length to be 25 frames, which according to [4] achieves a good trade-off between classification performance and length. As starting point for the optimization problem in Eq. 2.8, the matrix $\boldsymbol{A}^o$ is computed by PCA to retain 90% of the original space variance. In addition, to estimate the Gaussian kernel bandwidth σ from the same equation, we use the information theoretic learning framework proposed in [29].

For both IP and GB recognition tasks, the classification stage comprises a *k-nn* classifier through a 10-fold nested cross-validation scheme. The number of neighbors was heuristically found within the range $\{1, 3, 5, 7, 9, 11\}$. Since the *Disoriented* class has considerably few samples (seven in total), we forced the 75% of samples to appear in the training folds and the remaining 25% for testing in each iteration.

4.1.1 Validation

The validation of *KRAV*, for both IP and GB recognition, is carried out by the following three tests:

1) Feature ranking: For both individual and group cases, we use *KRAV* to determine the most relevant high-level semantic features according to their capability in discriminating among different social behaviors. To this end, we calculate the relevance vector $\boldsymbol{\varrho}$ ranking the original feature space in $\boldsymbol{X}$ as explained in Section 4.0.2. The proposed *KRAV* is compared with two baseline feature relevance methods. The first method is a variance-based relevance analysis (termed *VRA*) that computes the relevance vector according to a variability criterion [30].

The parameter related to the percentage of retained variance in *VRA* was set to 90%. The second method called *Relief-f*, calculates the relevance vector by looking for the closest class samples using a *k-nn* classifier [31]. The *Relief-f* parameter related to the number of neighbors was set to 1.

2) Feature selection: we calculate a performance curve using a *k-nn* classifier, through a 10 fold cross-validation scheme. The curve is obtained by adding in succession features ranked according to their relevance value in $\boldsymbol{\varrho}$. The classification performance for this experiment is assessed using the F_1 measure, which jointly considers the Precision and Recall. The best subset of relevant features is selected according to their classification performance and generates a new feature space denoted by $\boldsymbol{X}_s$.

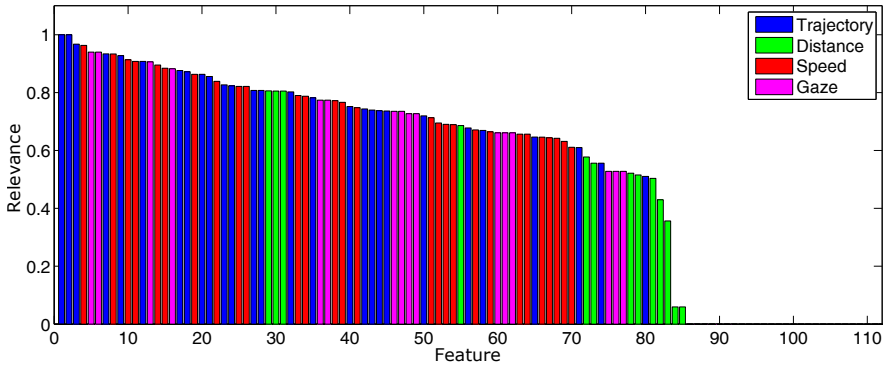
3) KRAV as feature embedding (KRAV-E): we evaluate the impact of the CKA-based embedding into the classification performance using the projection model $\boldsymbol{X}_e{=}\boldsymbol{X}_s\widetilde{\boldsymbol{A}}$ (see Section 4.0.2).

4.2 Results and discussions

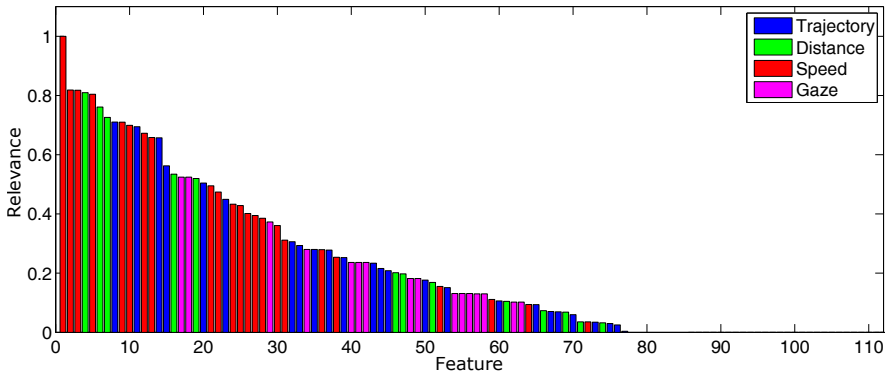
4.2.1 Feature ranking analysis

The relevance vectors $\boldsymbol{\varrho}$ obtained using *VRA*, *Relief-f*, and the proposed *KRAV* for IP behavior recognition are shown in Fig. 4.3. In each case, vectors $\boldsymbol{\varrho}$ are normalized to the interval $[0, 1]$, and sorted in decreasing order regarding their relevance value. As seen, each method highlights different sets of features as relevant for IP classification task. Particularly, for the *VRA* method, most features provide similar information as shown in Fig. 4.3a. Besides, the bins related to the *Distance* descriptor are the overall less important characteristics. The *VRA* definition can explain these results as it seeks a linear combination of features to maximize the variability among data samples, regardless of the label information. On the other hand, the *Relief-f* and our *KRAV* method incorporate supervised information to rank the original input features with a more discriminative order. For *Relief-f* criteria, the most relevant features are the ones related to the *Speed* bins (see Fig. 4.3b). Whereas, for the *KRAV*'s criteria, the most relevant features are related to the *Distance* bins (see Fig. 4.3c). The difference between the obtained $\boldsymbol{\varrho}$ for these methods can be attributed to the fact that *KRAV* considers the high-class imbalance for this classification problem, while the *Relief-f* gives the same importance to all the samples disregarding the imbalance issue.

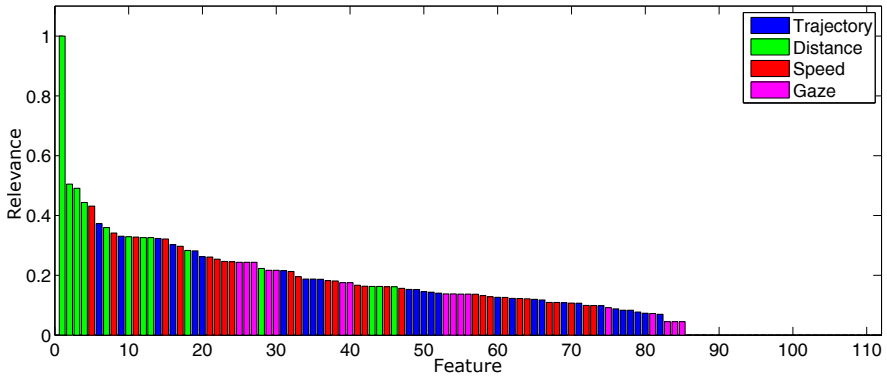
Now, for the GB relevance vectors shown in Fig. 4.4, we see that *VRA* behaves as it did for the IP recognition, giving similar importance to most of the features. Moreover, the *Profiles Information* is not relevant for the method. In contrast, as seen in Fig. 4.4b and Fig. 4.4c,



(a) *VRA*



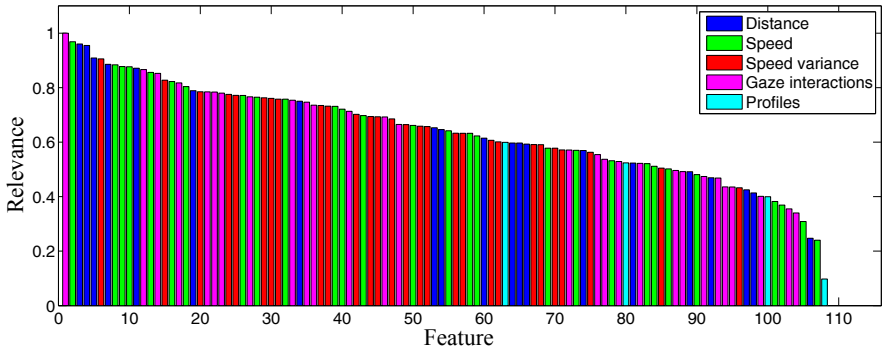
(b) *Relief-f*



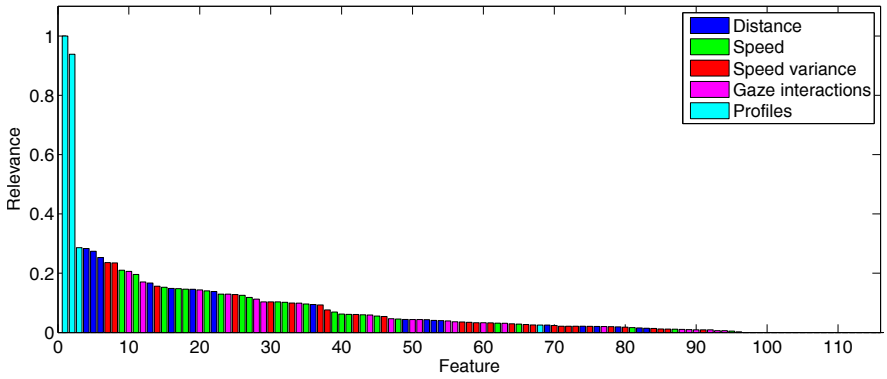
(c) *KRAV*

Figure 4.3: IP feature ranking results.

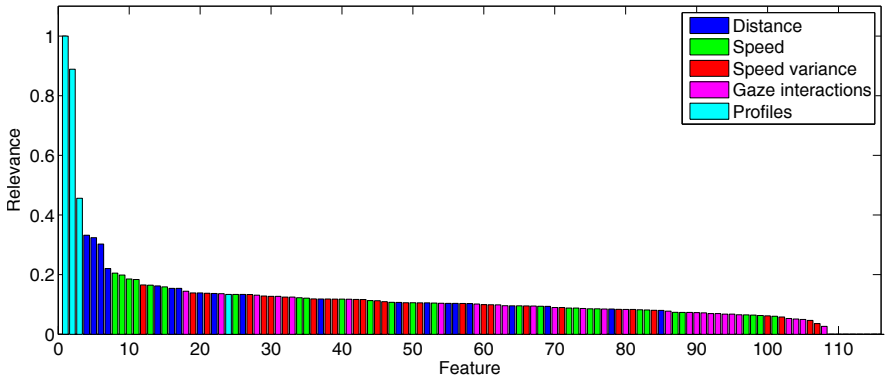
the *Relief-f* and the *KRAV* methods identify that the most relevant features are related to the *Profiles Information*. Unlike for IP, here these methods behave similarly since the class imbalance effect is not as notorious. Some differences can be seen only after the 7th feature, where *Relief-f* includes *Speed Variance* bins while *KRAV* recognizes the *Speed* as more relevant.



(a) *VRA*



(b) *Relief-f*



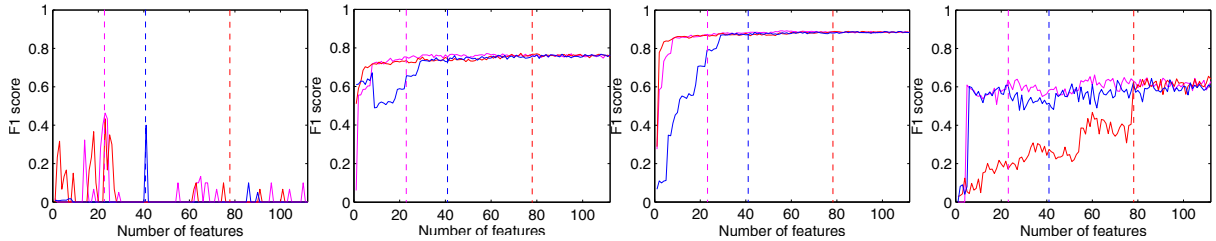
(c) *KRAV*

Figure 4.4: GB feature ranking results.

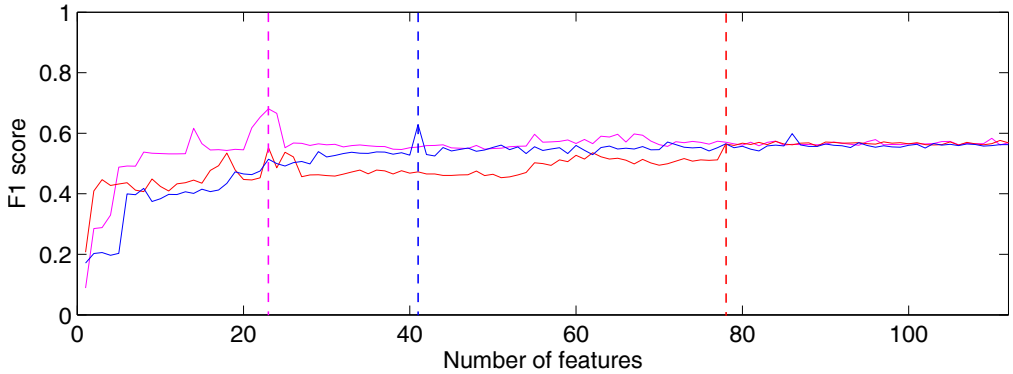
4.2.2 Feature selection results

Fig. 4.5 shows the F_1 classification curve for the IP recognition task from the vector $\boldsymbol{\varrho}$ of *VRA* (red), *Relief-f* (blue), and *KRAV* (pink). The dashed lines indicate the selected subset of relevant features to conform $\boldsymbol{X}_s$, which correspond to $M_s = 78$ for *VRA*, $M_s = 41$ for

Relief-f, and $M_s = 23$ for *KRAV*. The threshold selection criteria for M_s was set to be the highest value of the F_1 classification curve. Fig. 4.5e, shows that the proposed *KRAV* method obtains the highest classification performance 0.6807, with the lowest number of employed features. While analyzing individually the class performance, is noticeable that given the high imbalance, the *Disoriented* class obtains the lowest classification performance, which only spikes when the selection of some arbitrary subset of features is obtained (see Fig. 4.5a). For the *Distracted* and *Exploring* classes, the highest classification performance are 0.77 and 0.83, respectively. These values are obtained with the selection of a small set of features. Lastly, for the *Interested* class it can be seen that the *VRA* method has the worst performance, which can be explained by the low relevance value given to the features related to the *Distance* feature.



(a) Disoriented (b) Distracted (c) Exploring (d) Interested

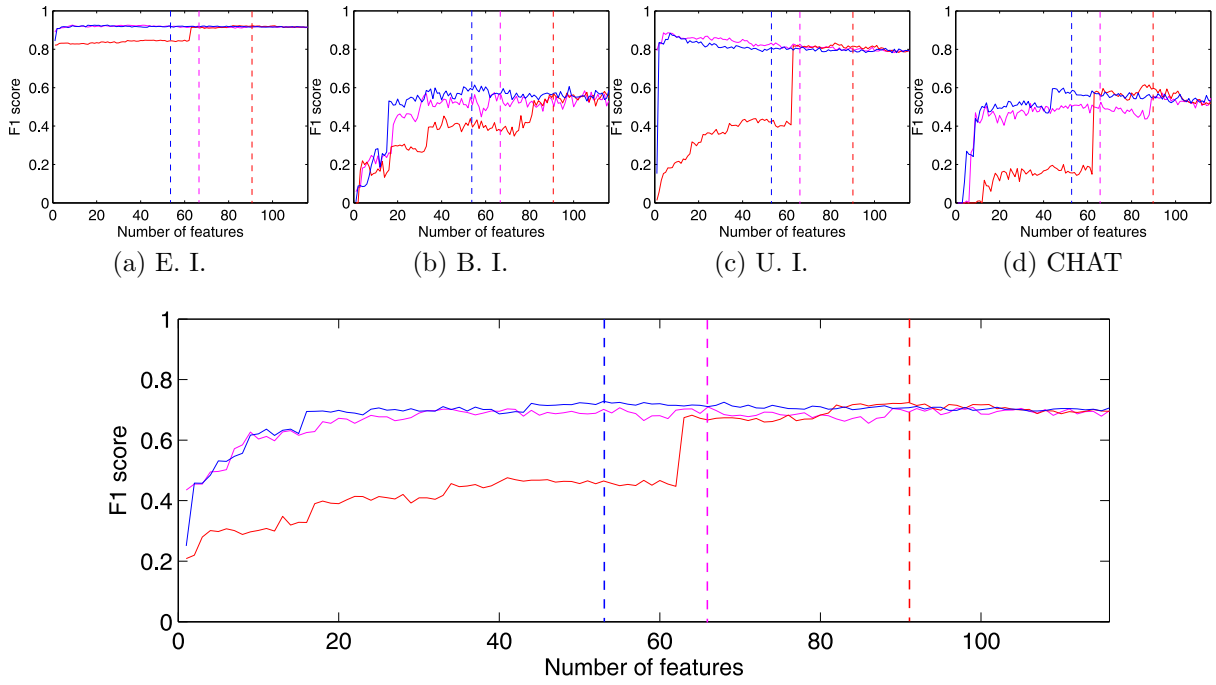


(e) average

Figure 4.5: IP recognition results while adding relevant features. —*Relief-f*, —*VRA* and —*KRAV*. The dashed lines indicate the selected M_s for each method.

Accordingly, Fig. 4.6 shows the best subset of relevant features for the GB recognition task, with respect to the F_1 classification performance curve, from the relevance vector $\boldsymbol{\varrho}$ of *VRA* (red), *Relief-f* (blue), and *KRAV* (pink). As exposed, the dashed lines indicate the selected subset, which corresponds to $M_s = 91$ for *VRA*, $M_s = 56$ for *Relief-f*, and $M_s = 66$ for *KRAV*. The average classification result shown in Fig. 4.6e, reveals that the performance curve for *Relief-f* and *KRAV* have similar behavior. For both, the most relevant features

are the *Profiles Information* bins, and the highest classification performance is 0.7239 and 0.7095, respectively. On the other hand, the *VRA* method does not obtain a good performance until the 63-th feature is added, which corresponds to a *Profiles Information* bin (see Fig. 4.4a). The above result demonstrates that the *Profiles Information* is relevant for the GB classification, as expected from the labels explanation in Section 4.0.1. Also, it can be seen that for the *E.I.* and *U.I.* classes, the classification performance reaches high ratings from the beginning of the succession. This remarks that the *Profiles Information* features are discriminative enough to separate both classes from the others. Differently, for the *B.I.* and *CHAT* classes, a larger number of features is required to achieve relatively high F_1 measure results. The latter can be related to the fact that these two classes have fewer samples than the other two. Thus, there is not enough information to learn patterns and discriminate them properly.



(a) E. I. (b) B. I. (c) U. I. (d) CHAT

(e) Average

Figure 4.6: GB classification results while adding relevant features. —*Relief-f*, —*VRA* and —*KRAV*. The dashed lines indicate the selected M_s for each method.

4.2.3 Feature embedding results

Table 4.2 shows the average F_1 measure for the IP and GB classification tasks. The results are obtained using *KRAV* as a feature selection method as in Section 4.2.2, termed *KRAV-S*,

and *KRAV* as a feature embedding tool, termed *KRAV-E*. Also, the average results obtained by the *VRA* and *Relief-f* feature selection methods are presented as well. In general, the average F_1 measure for GB is similar for all the considered methods, while for IP the results differ. This can be attributed to the class imbalance problem, which is not considered by *VRA* either *Relief-f*. By analyzing individually the results, *VRA* obtains the lowest performance for IP (0.5761), partially because it does not consider supervised information to rank the input features. Remarkably, *KRAV-E* obtains the best results for both IP and GB tasks, e.g., 0.7481 and 0.7611 $F1$ measure, respectively.

Table 4.2: $F1$ results and percentage of relevant characteristics for the IP and GB recognition using *VRA*, *Relief-f*, *KRAV-S*, and *KRAV-E*.

	IP		**GB**	
	Avg F_1	#feat	Avg F_1	# feat
VRA	0.5761	70%	0.7239	78%
Relief-f	0.6288	37%	0.7283	46%
KRAV-S	0.6807	21%	0.7095	57%
KRAV-E	**0.7481**	**21%**	**0.7611**	**57%**

4.3 Conclusions

In this paper, we introduced a kernel relevance analysis for video data to support social behavior recognition tasks, termed KRAV. Our approach highlights the primary semantic features to predict the output labels of the individual (IP) and group (GB) social behavior videos. Specifically, KRAV quantifies the relevance of several bins from a multi-scale feature representation towards a CKA-based algorithm, that matches the input space with the output labels. Remarkably, our method mitigates the imbalance effect of unusual human behaviors introducing information about the number of samples per class, making those with fewer samples more relevant. Also, the CKA algorithm allows computing a linear projection matrix, through a non-linear representation, where the columns quantify the required number of dimensions to preserve the 90% of input data variability. By projecting the video samples into the generated CKA space, the class separability is enhanced. Results on the IIT social behavior dataset show that for the IP recognition task our proposal obtains 0.7481 $F1$ measure using 21% of the input features, through a 10 fold cross-validation scheme. Likewise, for the G.B recognition task obtains 0.7611 $F1$ measure using 57% of the input features, through the same cross-validation scheme. In particular, according to the performed relevance analysis for IP recognition, the most relevant features are the bins related to the *Distance* descriptor which introduces information about the space layout quantifying the distance between individuals and objects of interest. Similarly, the relevance analysis for GB

recognition yield that the most relevant features are the bins related to the *Profiles Information* which introduces information about the social environment within a group. Remarkably, KRAV outperforms state-of-art results concerning the classification performance for both IP and GB recognition tasks. Besides, our video-based method allows interpreting the set of features selected regarding the IP and GB recognition, which would be useful for further social behavior analysis.

Chapter 5

Conclusions and future work

5.1 Conclusions

In this work, we studied the use of kernel methods and Bayesian inference to support video-based human action recognition. Proposed approaches intend to perform a feature relevance analysis and select a set of local samples that enable the precise codification of visual information. In chapter 2, we presented the most popular approach for human action recognition, which is based on the Bag of Visual Words methodology, and introduced our proposed kernel-based relevance analysis for ranking local descriptors. Our approach, termed HARK, also allow us to compute a linear projection matrix for mapping video sample into a CKA generated space, where the class separability is preserved and the representation dimensionality reduced. Attained results by the HARK methodology, revealed that the incorporation of a kernel-based relevance analysis allowed identifying the most essential video descriptors, effectively decreasing the computational burden. As it is known, that the number of operations for further processing stages increases exponentially on the representation length.

Aiming at further improving performance of the recognition system, while decreasing the computational complexity, we proposed an extension to the conventional Fisher Vector encoding technique. This novel framework presented in chapter 3 is based on the Infinite Gaussian Mixture Model, an aims at revealing a set of discriminant local spatiotemporal features that enable the precise codification of visual information from HAR tasks. Encoding visual data using the FV technique requires a Gaussian Mixture codebook, and seeks to determine each mixture component responsibility on explaining samples, in this case, videos. For creating this codebook, traditional approaches rely on optimization-based GMM and extensive cross-validation for determining model parameters. Meanwhile, our approach takes advantage of the IGMM formulation for developing a fully automatic data encoding framework, that in

one cross-validation run can specify every parameter in the model, including the number of Gaussian components. The Markov Chain Monte Carlo implementation of the hierarchical IGMM effectively prevent falling into local minima, which tend to plague mixtures trained by optimization based methods. Performed experiments showed that the proposed encoding framework obtained promising recognition performance and computational requirements savings. Further tests on the model ability to specify the form of the Gaussian covariances are required. As demonstrated, when the covariance form is limited, the model requires a more considerable amount of components for describing data distribution, which increases its resolution. This situation is of great interest because an increased resolution can address a couple of difficult Computer Vision challenges (occlusions and partially out of scene humans).

Finally, in chapter 4, we present a methodology for performing human behavior analysis. This task is a particular case of Human Action Recognition in which the scene context and non-verbal interactions must be considered for allowing a proper transcription of the human activity. Therefore, we proposed an approach to extract high-level semantic features from low-level video description (Human detection, tracking, and gaze direction), and later perform a relevance analysis for the selection and combination of these higher level features. Moreover, the employed database introduced the challenge of addressing data imbalance, as there is presence of unusual human behaviors. Conventional processing strategies tend to forsake these classes, reducing their ability to perform behavior prediction. Thus, we extend the traditional Centered Kernel Alignment technique by introducing information about the number of samples per class, so that in the CKA generated space, unusual behaviors become more relevant. Obtained results demonstrated that our approach outperforms current frameworks in both individual and group behavior recognition tasks.

5.1.1 Conferences and articles

13^{th} International Symposium on Visual Computing *November 19-21 2018 - Las Vegas, NV*

- **Fernández J**, Álvarez A, Orozco Á. *"Video-Based Human Action Recognition Using Kernel Relevance Analysis"*. In: Advances in Visual Computing. LNCS, vol. 11241, pp.116-125. Springer, Cham (2018).

6^{th} International Workshop on Pattern Recognitionand Artificial Intelligence *September 24-26, 2018 - Havana, CU*

- **Fernández J**, Álvarez A, Quintero H, Echeverry J, Orozco Á. *"Multilayer-Based HMM Training to Support Bearing Fault Diagnosis"*. In: Progress in Artificial Intelligence and Pattern Recognition. LNCS, vol. 11047, pp. 43-50. Springer, Cham (2018).
- Hoyos K, **Fernández J**, Martinez B, Henao Ó, Orozco Á, Daza G. *"Imbalanced Data Classification Using a Relevant Information-Based Sampling Approach"*. In: Progress in Artificial Intelligence and Pattern Recognition. LNCS, vol. 11047, pp. 280-287. Springer, Cham (2018).
- **Fernández J**, Rojas A, Daza G, Gómez D, Álvarez A, Orozco Á. *"Student Desertion Prediction Using Kernel Relevance Analysis"*. In: Progress in Artificial Intelligence and Pattern Recognition. LNCS, vol. 11047, pp.263-270. Springer, Cham (2018).

1^{er} Simposio de Investigación Comfamiliar *September 06, 2018 - Pereira, CO*

- **Fernández J**, Álvarez A. *"Sistema de videovigilancia orientado a la detección de comportamientos anormales en grupos de personas utilizando técnicas de visión por computador"*.

Fernández J, Álvarez A., Pereira E., Orozco Á., Castellanos G. *"Video-based social behavior recognition based on kernel relevance analysis"*. In: The Visual Computer Journal. Springer (2019). (*Under review*)

5.2 Future work

From the attained results and the drawbacks found along the process, the following theoretical and experimental topics could be explored.

- Regarding the proposed HARK framework, the employment of convolutional neural networks to both describe spatial and temporal video characteristics could be explored. Attained results in [1] demonstrated that inclusion of theses descriptors increases the recognition performance significantly.
- Regarding the proposed IGFV data encoding, the employment of clustering-based downsampling methods would be an exciting research line. As reducing the presence of outliers in the IGMM codebook generation could represent a closer estimation of the data distribution. Which ultimately enables a more precise codification of visual information.
- A more efficient implementation could be developed. The usage of multithread paradigm will be desirable to improve the system scalability and reduce processing time.

Chapter 6

Appendix

6.1 Diagonal Infinite Gaussian Mixture Model

In this case, we assume that all precision matrices are diagonal. Mathematical derivations led to the following equations:

For the component precision $\boldsymbol{S}_j$, the prior becomes:

$$\begin{aligned} p(\boldsymbol{S}_j/\beta, \boldsymbol{W}) &\sim \mathcal{W}(\beta, \boldsymbol{W}^{-1}) \\ &\propto \prod_{d=1}^{D} s_{d,j}^{\frac{\beta}{2}-1} \exp\{-\frac{1}{2}\beta\, s_{d,j}\, w_d\} \\ &\sim \prod_{d=1}^{D} \mathcal{G}(\beta, [w_d]^{-1}) \end{aligned} \tag{6.1}$$

where $s_{d,j}$ and w_d are the elements (d, d) from matrix $\boldsymbol{S}_j$ and $\boldsymbol{W}$, respectively. The conditional posterior on the diagonal precisions is obtained by multiplying the complete likelihood from $\boldsymbol{z}_t$, Eq. 3.1, and the diagonal precision prior, Eq. 6.1:

$$\begin{aligned} p(\boldsymbol{S}_j|\beta, \boldsymbol{W}, \boldsymbol{\mu}_j, \{\boldsymbol{z}_t : c_{t,j} = 1\}) &\propto \prod_{t:c_{t,j}=1} p(\boldsymbol{z}_t|\boldsymbol{S}_j, \boldsymbol{\mu}_j) \times p(\boldsymbol{S}_j|\beta, \boldsymbol{W}) \\ \sim \prod_{d=1}^{D} \mathcal{G}(\beta + n_j, \Big[&\frac{\beta\, w_d + \sum_{t:c_{t,j}=1} (z_{t,d} - \mu_{j,d})^2}{\beta + n_j}\Big]^{-1}) \end{aligned} \tag{6.2}$$

For the precision matrix $\boldsymbol{R}$, the prior becomes:

$$\begin{aligned} p(\boldsymbol{R}) &\sim \mathcal{W}(1, \mathbf{cov}_Z^{-1}) \\ &\propto \prod_{d=1}^{D} r_d^{\frac{1}{2}-1} \exp\{-\frac{1}{2} r_d \, [\mathbf{cov}_Z]_{d,d}\} \\ &\sim \prod_{d=1}^{D} \mathcal{G}(1, [\mathbf{cov}_Z]_{d,d}^{-1}) \end{aligned} \tag{6.3}$$

Accordingly, the posterior is:

$$\begin{aligned} p(\boldsymbol{R}|\{\boldsymbol{\mu}_j\}_{j=1}^{k}, \boldsymbol{\lambda}) &\propto \prod_{j=1}^{k} p(\boldsymbol{\mu}_j/\boldsymbol{\lambda}, \boldsymbol{R}) \times p(\boldsymbol{R}) \\ &\sim \prod_{d=1}^{D} \mathcal{G}(k+1, \Big(\frac{\sum_{j=1}^{k}(\mu_{j,d} - \lambda_d)^2 + [\mathbf{cov}_Z]_{d,d}}{k+1}\Big)^{-1}) \end{aligned} \tag{6.4}$$

For matrix $\boldsymbol{W}$, the prior becomes:

$$\begin{aligned} p(W) &\sim \mathcal{W}(1, \mathbf{cov}_Z) \\ &\propto \prod_{d=1}^{D} w_d^{\frac{1}{2}-1} \exp\{-\frac{1}{2} \, w_d \, [\mathbf{cov}_Z^{-1}]_d\} \\ &\sim \prod_{d=1}^{D} \mathcal{G}(1, ([\mathbf{cov}_Z^{-1}]_{d,d})^{-1}) \end{aligned} \tag{6.5}$$

likewise, the posterior is:

$$\begin{aligned} p(\boldsymbol{W}|\beta, \{\boldsymbol{S}_j\}_{j=1}^{k}) &\propto \prod_{j=1}^{k} p(\boldsymbol{S}_j|\beta, \boldsymbol{w}) \times p(\boldsymbol{W}) \\ &= \prod_{d=1}^{D} \mathcal{G}(k\beta+1, \Big[\frac{\beta \, \sum_{j=1}^{k} s_{d,j} + [\mathbf{cov}_Z^{-1}]_{d,d}}{k\beta+1}\Big]^{-1}) \end{aligned} \tag{6.6}$$

Bibliography

[1] I. C. Duta, B. Ionescu, K. Aizawa, and N. Sebe, "Spatio-temporal vlad encoding for human action recognition in videos," in *International Conference on Multimedia Modeling*. Springer, 2017, pp. 365–378.

[2] H. Wang, D. Oneata, J. Verbeek, and C. Schmid, "A robust and efficient video representation for action recognition," *International Journal of Computer Vision*, vol. 119, no. 3, pp. 219–238, 2016.

[3] A. M. Álvarez-Meza, S. Molina-Giraldo, and G. Castellanos-Dominguez, "Background modeling using object-based selective updating and correntropy adaptation," *Image and Vision Computing*, vol. 45, pp. 22–36, 2016.

[4] E. M. Pereira, L. Ciobanu, and J. S. Cardoso, "Cross-layer classification framework for automatic social behavioural analysis in surveillance scenario," *Neural Computing and Applications*, vol. 28, no. 9, pp. 2425–2444, 2017.

[5] F. Zhao, Y. Huang, L. Wang, T. Xiang, and T. Tan, "Learning relevance restricted boltzmann machine for unstructured group activity and event understanding," *International Journal of Computer Vision*, vol. 119, no. 3, pp. 329–345, 2016.

[6] A. B. Mabrouk and E. Zagrouba, "Abnormal behavior recognition for intelligent video surveillance systems: A review," *Expert Systems with Applications*, vol. 91, pp. 480–491, 2018.

[7] S. Vishwakarma and A. Agrawal, "A survey on activity recognition and behavior understanding in video surveillance," *The Visual Computer*, vol. 29, no. 10, pp. 983–1009, 2013.

[8] D. D. Dawn and S. H. Shaikh, "A comprehensive survey of human action recognition with spatio-temporal interest point (stip) detector," *The Visual Computer*, vol. 32, no. 3, pp. 289–306, 2016.

[9] J. Uijlings, I. C. Duta, E. Sangineto, and N. Sebe, "Video classification with densely extracted hog/hof/mbh features: an evaluation of the accuracy/computational efficiency trade-off," *International Journal of Multimedia Information Retrieval*, vol. 4, no. 1, pp. 33–44, 2015.

[10] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Dense trajectories and motion boundary descriptors for action recognition," *International journal of computer vision*, vol. 103, no. 1, pp. 60–79, 2013.

[11] A. M. Alvarez-Meza, A. Orozco-Gutierrez, and G. Castellanos-Dominguez, "Kernel-based relevance analysis with enhanced interpretability for detection of brain activity patterns," *Frontiers in neuroscience*, vol. 11, p. 550, 2017.

[12] S. Ai, T. Lu, and Y. Xiong, "Improved dense trajectories for action recognition based on random projection and fisher vectors," in *MIPPR 2017: Pattern Recognition and Computer Vision*, vol. 10609. International Society for Optics and Photonics, 2018, p. 1060915.

[13] K. Guo, P. Ishwar, and J. Konrad, "Action recognition from video using feature covariance matrices," *IEEE Transactions on Image Processing*, vol. 22, no. 6, pp. 2479–2494, 2013.

[14] M. Harandi, M. Salzmann, and R. Hartley, "Dimensionality reduction on spd manifolds: The emergence of geometry-aware methods," *IEEE transactions on pattern analysis and machine intelligence*, 2017.

[15] S. Wang, Y. Hou, Z. Li, J. Dong, and C. Tang, "Combining convnets with hand-crafted features for action recognition based on an hmm-svm classifier," *Multimedia Tools and Applications*, vol. 77, no. 15, pp. 18 983–18 998, 2018.

[16] W. Fan, N. Bouguila, and X. Liu, "A nonparametric bayesian learning model using accelerated variational inference and feature selection," *Pattern Analysis and Applications*, vol. 22, no. 1, pp. 63–74, 2019.

[17] M. Field, D. Stirling, Z. Pan, M. Ros, and F. Naghdy, "Recognizing human motions through mixture modeling of inertial data," *Pattern Recognition*, vol. 48, no. 8, pp. 2394–2406, 2015.

[18] P. V. K. Borges, N. Conci, and A. Cavallaro, "Video-based human behavior understanding: A survey," *IEEE transactions on circuits and systems for video technology*, vol. 23, no. 11, pp. 1993–2008, 2013.

[19] F. Perronnin, S. Jorge, and T. Mensink, "Improving the Fisher Kernel for Large-Scale Image Classification," pp. 143–156, 2010.

[20] A. J. Brockmeier, L. G. S. Giraldo, M. S. Emigh, J. Bae, J. S. Choi, J. T. Francis, and J. C. Principe, "Information-theoretic metric learning: 2-d linear projections of neural data for visualization," in *EMBC*. IEEE, 2013, pp. 5586–5589.

[21] Y. Wang, X. She, Y. Liao, H. Li, Q. Zhang, S. Zhang, X. Zheng, and J. Principe, "Tracking neural modulation depth by dual sequential monte carlo estimation on point processes for brain–machine interfaces," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 8, pp. 1728–1741, 2016.

[22] K. K. Reddy and M. Shah, "Recognizing 50 human action categories of web videos," *Machine Vision and Applications*, vol. 24, no. 5, pp. 971–981, 2013.

[23] T. Chen, J. Morris, and E. Martin, "Probability density estimation via an infinite gaussian mixture model: Application to statistical process monitoring," *Journal of the Royal Statistical Society. Series C: Applied Statistics*, vol. 55, no. 5, pp. 699–715, 2006.

[24] T. Priya, S. Prasad, and H. Wu, "Superpixels for spatially reinforced bayesian classification of hyperspectral images," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 5, pp. 1071–1075, 2015.

[25] R. Sicre and H. Nicolas, "Improved gaussian mixture model for the task of object tracking," in *International Conference on Computer Analysis of Images and Patterns*. Springer, 2011, pp. 389–396.

[26] C. Rasmussen, "The infinite gaussian mixture model," 2000, pp. 554–559.

[27] R. Neal, "Markov chain sampling methods for dirichlet process mixture models," vol. 9, no. 2, pp. 249–265, 2000.

[28] A. Adam *et al.*, "Robust real-time unusual event detection using multiple fixed-location monitors," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 30, no. 3, pp. 555–560, 2008.

[29] A. Álvarez-Meza *et al.*, "Unsupervised kernel function building using maximization of information potential variability," in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*. Springer, 2014, pp. 335–342.

[30] G. Daza-Santacoloma *et al.*, "Dynamic feature extraction: an application to voice pathology detection," *Intelligent Automation & Soft Computing*, vol. 15, no. 4, pp. 667–682, 2009.

[31] M. Robnik-Šikonja and I. Kononenko, "Theoretical and empirical analysis of relieff and rrelieff," *Machine learning*, vol. 53, no. 1-2, pp. 23–69, 2003.