Modeling Semi-Bounded Support Data using Non-Gaussian Hidden Markov Models with Applications

Rim Nasfi

A Thesis
in
The Concordia Institute
for
Information Systems Engineering

Presented in Partial Fulfillment of the Requirements
For the Degree of
Doctor of Philosophy (Information and Systems Engineering) at
Concordia University
Montréal, Québec, Canada

April 2022

CONCORDIA UNIVERSITY
School of Graduate Studies

This is to certify that the thesis prepared

By: **Rim Nasfi**

Entitled: **Modeling Semi-Bounded Support Data using Non-Gaussian Hidden Markov Models with Applications**

and submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy (Information and Systems Engineering)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

_____ Chair
Dr. Zachary Patterson

_____ External Examiner
Dr. Hamid Bouchachia

_____ External to Program
Dr. Fuzhan Nasiri

_____ Examiner
Dr. Abdessamad Ben Hamza

_____ Thesis Supervisor
Dr. Nizar Bouguila

Approved by _____
Dr. Zachary Patterson, Graduate Program Director

May 31st, 2022 _____
Dr. Mourad Debbabi, Dean
Gina Cody School of Engineering and Computer Science

Abstract

Modeling Semi-Bounded Support Data using Non-Gaussian Hidden Markov Models with Applications

Rim Nasfi, Ph.D.

Concordia University, 2022

With the exponential growth of data in all formats, and data categorization rapidly becoming one of the most essential components of data analysis, it is crucial to research and identify hidden patterns in order to extract valuable information that promotes accurate and solid decision making. Because data modeling is the first stage in accomplishing any of these tasks, its accuracy and consistency are critical for later development of a complete data processing framework. Furthermore, an appropriate distribution selection that corresponds to the nature of the data is a particularly interesting subject of research. Hidden Markov Models (HMMs) are some of the most impressively powerful probabilistic models, which have recently made a big resurgence in the machine learning industry, despite having been recognized for decades. Their ever-increasing application in a variety of critical practical settings to model varied and heterogeneous data (image, video, audio, time series, etc.) is the subject of countless extensions. Equally prevalent, finite mixture models are a potent tool for modeling heterogeneous data of various natures.

The over-use of Gaussian mixture models for data modeling in the literature is one of the main driving forces for this thesis. This work focuses on modeling positive vectors, which naturally occur in a variety of real-life applications, by proposing novel HMMs extensions using the Inverted Dirichlet, the Generalized Inverted Dirichlet and the Beta-Liouville mixture models as emission probabilities. These extensions are motivated by

the proven capacity of these mixtures to deal with positive vectors and overcome mixture models' impotence to account for any ordering or temporal limitations relative to the information. We utilize the aforementioned distributions to derive several theoretical approaches for learning and deploying Hidden Markov Models in real-world settings. Further, we study online learning of parameters and explore the integration of a feature selection methodology. Extensive experimentation on highly challenging applications ranging from image categorization, video categorization, indoor occupancy estimation and Natural Language Processing, reveals scenarios in which such models are appropriate to apply, and proves their effectiveness compared to the extensively used Gaussian-based models.

Acknowledgments

First, I would like to show my deepest gratitude to my supervisor Dr. Nizar Bouguila whose unfaltering support and encouragement have been endless throughout this journey. Although these words could never give him the credit he truly deserves, I would like to express how honored I feel to be guided by such a committed person whose insightful knowledge and undoubted talent have been steering me and illuminating my path. I will always be grateful not only for the valuable remarks and feedback that helped hone my research skills, but also for pushing me to think outside the box and explore the perks of leaving my comfort zone as a researcher. I could not ask for a better mentor but also for a better friend and supporter.

Second, my most sincere appreciation goes to all the committee members, who have allotted their valuable time to review my reports and provide me with pertinent remarks and enriching discussions that highly contributed to the improvement of my thesis. I would also like to express how lucky I am for having been surrounded by my dearest friends Fatma, Omar, Nuha, Basim, Huda, Kamal, Muhammad and Eddy, to name a few, during this roller-coaster journey. They have always been there during good times to celebrate my small wins, but also to show empathy and reassurance during tougher times.

Words cannot describe how thankful I am to my life companion and husband Zied for being the most caring and supportive partner a person could ever ask for. Thank you for believing in me when I sometimes fail to believe in myself. Thank you for endlessly giving me the rounds of applause I sometimes feel unworthy of. Thank you for pushing me harder

to do better everyday. I will forever be indebted to your love and affection. You have been and always will be my lifelong companion, my most fervent cheerleader, my best friend and my solid rock.

Last but most importantly, I'm deeply thankful to my family for their unwavering support and constant thoughtfulness. Even though they could not be physically present, their inspirational words have always succeeded in fueling me with the determination and self-confidence I needed to do better every day.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Artificial intelligence, data processing, and machine learning advancements have ushered in a new era of automation where machines match or outperform human performance in a variety of work activities involving cognitive talents. This was made possible thanks to computer vision, natural language processing, unsupervised learning and many other artificial intelligence domains. When deciphering the human cognitive system, we understand that it is endowed with one unique and powerful aspect: the ability to speculate, i.e., synthesize and process events and objects that are not necessarily linked to the current observed reality. Humans manipulate what they perceive and plan their actions or expect imminent events based on a very intuitive and complex procedure. They can examine hypotheses about the dynamics of the world, as well as understand facts and predict actions in the absence of explicit help or supervision (guessing weather conditions, analyzing facial expressions, etc.). Exploiting data to achieve engineering goals like these, is the essence of machine learning.

Machine learning models are divided into two main categories, discriminative and generative. Particularly used for supervised learning, discriminative models learn the boundaries between classes in a data set. They create instances using probability estimates and maximum likelihood without generating new data points. Separating one class from another is

the ultimate goal of discriminative models. This category includes models such as logistic regression, Support Vector Machines (SVM), neural networks and random forests. In this thesis, we focus on the second category which is generative models.

Generative models are considered to be one of the machine learning methods that aspire to empower machines with the fundamental capacity to examine objects, events or observations and speculate the upcoming aftermath. Drawing an analogy with the human cognitive system, generative models would constantly absorb perceptual data (images, sounds, etc.) and use its features to be trained and learn how it would be to draw reasoning from a combination of data features, depending on the assigned task, e.g., classification, clustering, prediction, etc. Furthermore, there is a great need to develop systems that can handle uncertainty and missing or incomplete data as well as undertake predictions to fill in these informational gaps.

A generative model produces the sequence of future observable events conditioned upon previously occurred observable events: $p(Obs_{t:T}|Obs_{1:t})$. Thus, if we have a system with variables $Obs_1, ..., Obs_T$, a system could be specified through a joint probability distribution over all significant variables within it $p(Obs_1, ..., Obs_T)$ [131]. Given a probability distribution, we are able to generate samples of various configurations of the system, thereby the latter is referred to as a generative probabilistic model [131]. Hidden Markov Models (HMMs) [203, 204], mixture models and Bayesian networks are among the most exploited generative probabilistic models in pattern recognition and event prediction [31]. These models rely on their capacity to identify conditional independencies between variables. In order to specify the prior knowledge needed for their structure, along with a parametric specification of prior distributions, observations are later combined with this prior knowledge to grant efficient model training. Thanks to the progress that they have brought to the machine learning field, generative probabilistic models have been extensively used to model a wide variety of data efficiently (text, image, video, audio, time series, etc.)

[60, 195, 95]. The majority of probabilistic models fall into the generative category by modeling how the data have been generated via distributions. The Gaussian distribution is the most commonly used in the literature. Notwithstanding, this type of distribution is not well adapted to semi-bounded data. A better practice is to choose the distributions depending on the nature of the processed data. In this thesis, we focus on Hidden Markov Models as the main used generative model.

1.1 Hidden Markov Models

1.1.1 Related Work

Hidden Markov Models have proven to be very useful in a huge variety of applications in machine learning. They are one of the most powerful tools to conduct prediction and pattern recognition tasks. Used for decades and adopted for their predictive aspect, HMMs are still in a state of development. HMMs have been introduced in their full generality in 1966 when Baum et al. [19] developed and investigated a maximum likelihood (ML) method in order to estimate HMM parameters of a training observation sequence. Earlier in 1948, an opening to this reasoning axis was discovered by Shannon when he developed a model for the English language, referred to as a Markov Source [225].

HMMs have long been referred to as dynamic probabilistic methods and have, till date, been used in numerous applications such as signature verification [138, 193, 18], speech processing [204, 203], anomaly detection [79], as well as in various pattern recognition tasks such as gesture and texture recognition [4, 201, 81]. The highly influential tutorial by Rabiner [204], based on tutorials by Jack Ferguson in the 1960s, pioneered in presenting the idea of the three fundamental problems to be considered when characterizing a hidden Markov model. These problems, as well as the complete structure of HMMs presented in [204], will be thoroughly explained later in this thesis. Various forms of extensions have

been tested to model diversified and heterogeneous data (image, video, audio, time series) in numerous important practical situations. Although their common structure and flow of steps seem to be quite standardized and usually performed in the same fashion, numerous experiments have been carried out contributing to new adaptations and extensions. The former has been driven by multiple aspects such as the need to raise the ability of a system to manipulate different forms of data and to simplify the task of modeling time series in order to embed randomness in the temporal nonstationary, spatially variable, but regular, learnable patterns of real-life events [250]. Figure 1 illustrates an example of an HMM-based decision system in the context of automated cars.

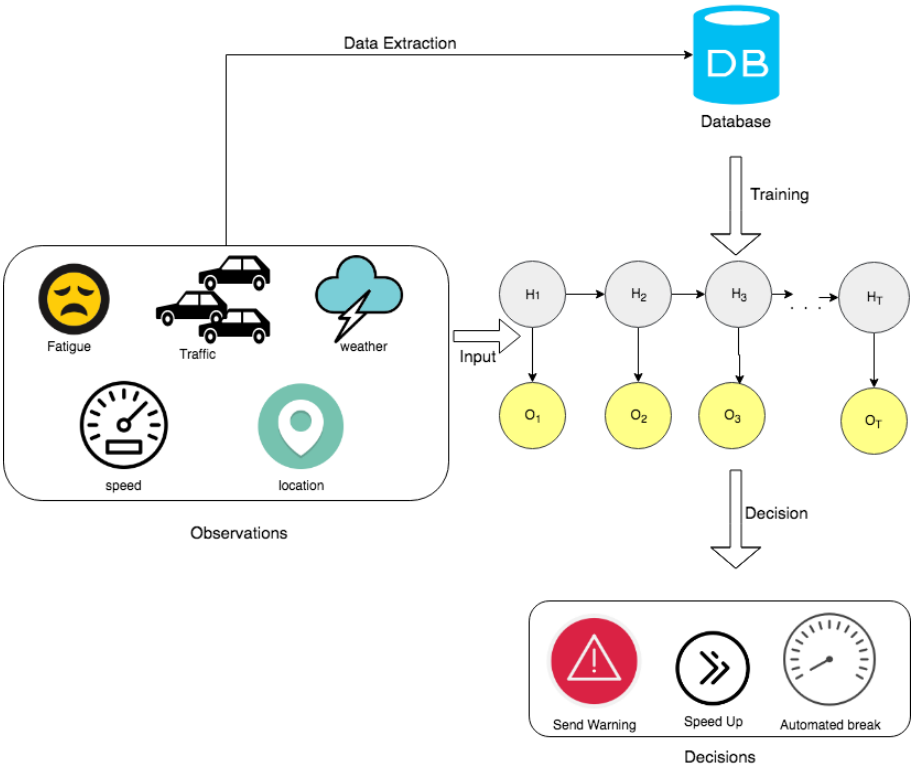


Figure 1: HMM-based decision system for automated cars

The use of HMMs in speech processing has been very extensive due to the convenience

of handling one-dimensional data [136]. In [106], authors develop an HMM extension suited for complex large vocabulary speech recognition tasks. This applies to broadcast news transcription and large portions of dictation speeches. The work in [106] introduced the outcomes of the potential dynamic Bayesian Networks use as well as the integration of single Gaussian diagonal covariance into the traditional HMM framework. The developed approach allowed the automated training of the model and ensured an improved noise compensation and multi-pass system combination thanks to covariance modelling and feature projection. Research using the same models with different techniques proceeded in the scope of speech recognition to get a hold of the psychological aspect by focusing on emotions related to speech syntheses [187, 220, 168]. In their work, Mao et al. [168] investigated three different HMM extensions, namely Gaussian mixture model (GMM) based HMMs, the subspace-based Gaussian mixture model-based HMMs and the hybrid deep neural network HMMs. They proved that these extensions of hidden Markov models are among the most powerful tools to be used in speech emotion recognition by incorporating various advanced techniques from the automatic speech recognition field to further enhance state-of-the-art accuracies.

1.1.2 Model Specification

Hidden Markov Models are described according to Rabiner [204], as: "A doubly stochastic process with an underlying stochastic process that is hidden and can only be observed through another set of stochastic processes that produce the sequence of observed symbols". To put it differently, an HMM is a learnable random process characterized by double random processes that have the following properties: The first is a finite set of unknown states (events). Each of these states is represented by a probability distribution. The transitions between them obey a set of probabilities called transition probabilities. The second is a set of observable states (events) that are analyzed in order to determine the hidden ones

that have emitted them. These models are a generalization of mixture models [111, 31]. In fact, the probability density functions over all observable states which are defined by an HMM are considered as a mixture of densities defined by each state. With HMMs, we are aiming to represent probability distributions over sequences of observations, with the assumption that the observations are discrete. An observation at time t is denoted by the variable O_T.

Two main properties delineate the hidden Markov model. First, it assumes that the observation at time t was generated by some process whose state q_t is hidden from the observer. Second, it assumes that the state of this hidden process satisfies the Markov property; that is, given the value of q_{t-1}; the current state q_t is independent of all the states prior to $t-1$. The task of the learning algorithm is to find the best set of state transitions and emission probabilities between the states of the model. Therefore, an output sequence or a set of these sequences is given.

A HMM is characterized by an ordered observation sequence $O = \{O_1, ..., O_T\}$ generated by hidden states $H = \{h_1, ..., h_T\}; h_j \in [1, K]$ where K is the number of the states, we have:

A transition probabilities matrix: $B = \{B_{ij} = P(h_t = j | h_{t-1} = j')\}$ and the emission probabilities matrix (Continuous case): $C = \{C_{ij} = P(m_t = i | h_t = j)\}; i \in [1, M]$ where M is the number of mixture components associated with state j (which can be assumed to be the same for all states without loss of generality). We define the intial probability: π_j which is the probability to start the observation sequence from the state j.

We denote an HMM as $\lambda = \{B, C, \varphi, \pi\}$ where φ is the set of mixture parameters depending on the chosen type of mixture.

1.2 Mixture Models

Although HMMs were mainly developed for discrete and Gaussian data [204], the diversity of applications in contexts and domains such as anomaly detection, diseases diagnosis and dynamic forecasting, increased the necessity of modifying the underlying HMM algorithm so that it efficiently suits those new data types [221]. From this perspective, proportional data modeling has been a study focus in [53] which used Dirichlet mixtures as emission probabilities for the first time, applied in synthetic proportional data, which has been extended in [81] afterwards and applied to real-world data. Later on, Epaillard et al. [79] adapted HMMs to embed separately the Beta-Liouville distribution as well as the generalized Dirichlet mixtures as emission probabilities. Further, [8] extended this work using variational inference on the aforementioned distributions.

While HMMs are considered a powerful, extensively used probabilistic tool, mixture models have been in their turn equally used, thanks to their ability to accurately represent multimodal data that a single distribution cannot. In fact, almost all natural distributions like audio and image datasets are multimodal. Therefore, in order to overcome the modeling limitation of most unimodal closed-form densities, the latter is replaced with a mixture over densities with distinct parameters. Mixture models have demonstrated a strong modeling capacity in numerous applications such as discrete data clustering [37], objects and text categorization [166, 36], image color analysis [52], etc.

Despite the widespread use of the Gaussian mixture models due to their mathematical convenience [128, 102], researchers lean towards using various other distributions. This choice is often due to the unrealistic description that GMM sometimes provides of specific types of data which can not be symmetric in all cases, not to mention the unbounded support that might not as well be accurate to describe all data natures. This idea has been discussed in [39], and, as an alternative, the Generalized Dirichlet mixture models [55, 242] have been adopted mainly for their compact support [0,1] suitable for data originating from

videos, images or texts, and also relying on its flexibility in the approximation of both symmetric and asymmetric distributions. Similarly, [24] adopted, for the first time, the Inverted Dirichlet (ID) mixture for positive vector clustering motivated by the significant flexibility of this distribution which permits multiple symmetric and asymmetric modes. The ID distribution does not only grant higher convenience in the modeling of previously stated types of data, but it also shows in the aforementioned work greater capabilities in modeling both symmetric and asymmetric high-dimensional data [234]. Thus, GD and ID mixture models proved that they can outperform the GMM by discovering more efficiently useful patterns [24, 53]. In this thesis, we focus on investigating mainly 3 distributions, i.e. the Inverted Dirichlet, the Generalized Inverted Dirichlet (GID) and the Inverted Beta Liouville (IBL).

1.3 Contributions

This thesis aims to propose several novel approaches and extensions of the widely used HMM. Each of these approaches is finely tuned to handle the semi-bounded nature of our data and provide better modeling capacities when dealing with positive vectors. We also pay particular attention to several aspects such as feature selection and the nature of learning. The overall contributions of this thesis are as follows

- In Chapter 2, we propose a novel method using inverted Dirichlet mixtures to model HMM emission probabilities, which to the extent of our knowledge, is the first integration of this powerful distribution to HMMs framework. This extension is motivated by the capacity of inverted Dirichlet mixtures to deal with positive vectors, and overcome any limitations when adopting a Gaussian mixture. We detail the complete inference and parameter estimation and showcase the performance of this method

through applications in the context of image categorization, but also in a study case of indoor occupancy detection. This work is also publicly available as:
Nasfi, Rim, Manar Amayri, and Nizar Bouguila. "A novel approach for modeling positive vectors with inverted Dirichlet-based hidden Markov models." Knowledge-Based Systems 192 (2020): 105335. [178].

- In Chapter 3, we explore, as a first-time experiment in state of the art to the best of our knowledge, the use of Generalized Inverted Dirichlet mixtures as HMM emission probabilities to further push the modeling capabilities when working with positive vectors. We also propose the complete integration framework of embedded feature selection into HMMs. Applications in fields such as facial expression recognition and scenes categorization are explored and comparable to high-performance results are presented. This research has also been published as:
Nasfi, Rim, and Nizar Bouguila. "A novel feature selection method using generalized inverted Dirichlet-based HMMs for image categorization." International Journal of Machine Learning and Cybernetics (2022): 1-17. [180].

- In Chapter 4, we investigate the use of Inverted Beta-Liouville mixtures as HMM emission probabilities along with an embedded online parameter estimation setting to tackle the problem of anomaly detection in crowd scenes. This work has been accepted as a book chapter in the upcoming volume entitled *Hidden Markov Models and Applications* to be published in the book series *Unsupervised and Semi-Supervised Learning*, Springer, under the title *"Online learning of Inverted Beta-Liouville HMMs for Anomaly Detection in Crowd Scenes"* [181].

- In Chapter 5, we further improve the recognition capacities of the aforementioned models by considering hybrid generative-discriminative approaches for each of the studied distributions. We conduct validations on different applications ranging from

indoor activity recognition to sentiment analysis. This work has been submitted for publication in the form of two conference papers under the titles:

- *Indoor Activity Recognition Using a Hybrid Generative-Discriminative Approach with Hidden Markov Models and Support Vector Machines* [179] (Accepted)
- *Sentiment Analysis from User Reviews Using a Hybrid Generative-Discriminative HMM-SVM Approach* [182]

- Finally a general conclusion closes this thesis and discusses future work propositions in Chapter 6.

Chapter 2

A Novel Approach for Modeling Positive Vectors with Inverted Dirichlet-Based Hidden Markov Models

Hidden Markov Models (HMMs) are among the most remarkably powerful probabilistic models, that although been acknowledged for decades have recently made a huge resurgence in the machine learning field. Their ever-growing use to model diversified and heterogeneous data (image,video, audio, time series) in numerous important practical situations is the subject of all forms of perpetual extensions. This work presents what we believe to be the first integration of the Inverted Dirichlet (ID) Mixture Models into the framework of HMMs. The proposed method uses the inverted Dirichlet mixtures to model the emission probabilities also known as observation probabilities. This extension (IDHMM), is motivated by the proven capacity of these mixtures to deal with positive vectors and overcome mixture models' capability to take into account any ordering or temporal constraints relative to information. The complete inference and parameter estimation are detailed in this work. Applications in the context of image categorization and indoor occupancy detection

demonstrate higher performance compared to the extensively used Gaussian mixture-based Hidden Markov Model (GHMM) and the Dirichlet mixture-based hidden Markov Model (DHMM).

2.1 Introduction

HMMs are powerful versatile statistical models that have proven to be not only useful but also efficient in various machine learning-based applications. They were introduced in their full generality in 1966 when Baum et al. [20] developed and investigated a maximum likelihood (ML) method, in order to estimate HMM parameters of a training observation sequence. Earlier in 1948, an opening to this reasoning axis was discovered by Shannon when he developed a model for the English language, it was referred to as a Markov Source [225]. HMMs have long been referred to as dynamic probabilistic methods and have, till date, been used in numerous applications such as signature verification [138, 193, 18], speech processing [203], anomaly detection [79], as well as in various pattern recognition tasks such as gesture and texture recognition [4, 201, 81]. Recently, it has also started to be used for occupancy estimation for smart buildings [5, 12].

The whole idea behind the use of HMMs is that these models showed real cogency in dealing with characterizing real-world signals as they are capable of modeling different types and natures of data, namely discrete and continuous. Above all, they have been proved to be very efficient when dealing with non-observable or missing data by providing a great deal of information without having the physical event, hence helping interpret and predict scenarios prior to their occurrence and disclosing latent variables that direct observations could not reveal. A few works attempted to bring some improvements in the HMM structure by mending and tuning the initialization process with regard to compactness and parameter settings [150, 29]. For its part, the training process of HMMs maintains the same standardized form. However, the choice of emission probability distribution functions is still not

a common discussion topic since it's usually Gaussian Mixture Models (GMMs) that are adopted by default [29, 54], and the motive behind their use is often left without strong justifications given their mathematical and practical convenience. However, this easing aspect is potentially insufficient to achieve the best results, particularly because of the unrealistic description that GMM sometimes provides of specific types of data which can not be symmetric in all cases, not to mention the unbounded support that might not as well be accurate to describe all data natures. This idea has been discussed in [39], and as an alternative, the Dirichlet Mixture Model (DMM) has been adopted mainly for its compact support [0,1] suitable for data originating from videos, images or texts, and also relies on its flexibility in the approximation of both symmetric and asymmetric distributions. Thus, DMM proved that it can outperform the GMM by discovering more efficiently useful patterns [24, 53].
In this paper, we are focusing on the modeling and clustering of different forms of positive vectors, and we are seeking flexibility in the modeling of this type of data. Therefore, in the present paper, we propose to explore and evaluate the performance of HMMs by setting Inverted Dirichlet (ID) as emission probability distribution and comparing them to the Gaussian-based HMMs. As a matter of fact, since their early appearance, HMMs suffered several drawbacks such as their limitation to fit numerous applications. For example, the first-order Markov Property is specifically limiting in the context of speech recognition in view of the fact that sound dependencies sometimes do extend through several states. Therefore, despite their strong adaptivity to a large range of real-life applications, researchers continuously attempted to improve the structure of HMMs, namely for dynamical non-gaussian systems. The latter requires non-gaussian emission probabilities and thus a major modification of the HMM structure mainly in terms of parameter estimation by finding a tractable solution to the inference problem.

As a matter of fact, being confronted with various types of information such as voice,

music, images, videos, etc. is a real challenge in terms of sequence prediction and pattern recognition. In practical terms, the previously mentioned types of information are rigorously tied to the temporal ordering aspect that defines their nature (For example voice sequences depend strictly on a given order of sequence). HMMs can explicitly handle the states' ordering sequence and provide a suitable solution for the automatic recognition of temporal events (activity recognition, signature verification, etc.). Therefore, these models are used in this work to improve the recognition process along with a powerful distribution which is the Inverted Dirichlet distribution that previously proved its effectiveness to handle positive vectors. The latter is the main focus of our work. We aim to show that the combination of HMMs and ID can provide us with better results when it comes to handling positive data under temporal constraints.

To the best of our knowledge, the integration of the inverted Dirichlet distribution into the HMM framework is unprecedented. This choice is driven by the flexibility of the inverted Dirichlet distribution which permits a high modeling capability when dealing with positive vectors as well as multiple symmetric and asymmetric modes. Moreover, when dealing with positive data vectors, which is the case of extracted visual features in many computer vision applications, ID distribution demonstrated huge flexibility and outperformed the Gaussian mixture [26]. As part of this study, we concentrate on the learning process of the model not only in terms of understanding the relationship between the model's performance and the number of both states and mixture components but also in terms of how to estimate and adjust the model parameters to best suit a specific observation sequence. Indeed, the number of components has its share of effect on the modeling accuracy. To handle this problem and tackle the parameter estimation task, a Maximum Likelihood approach is adopted using the traditional Expectation-Maximization (EM) [63] framework.

To outline the major contributions of this work, we cite: First, we put forward a complete

derivation of the equations for the integration of the Inverted Dirichlet mixture into the HMM framework. Second, we apply this new framework to four different scenarios related to image categorization and real case occupancy estimation.

The paper is organized as follows: Section 2 presents the related work done exploiting HMMs in various application contexts with a particular interest in the originality of the model extension and the nature of used mixture models. Section 3 briefly recalls the structure and the general functioning of HMMs. In section 4 we specify the different equations framing the proposed Inverted Dirichlet HMM ($\mathcal{ID}HMM$) model including the estimation of the distribution parameters. Section 5 presents the different developed applications as well as their respective adopted preprocessing methodologies and experimental results. Section 6 provides the summary, conclusion and potential future research.

2.2 Related Work

Since their promotive resurgence in the famous work of Rabiner and Juang in [203], HMMs have been the subject of many adaptations and extensions driven by multiple aspects such as the need to raise the ability of a system to manipulate different forms of data, and to simplify the task of modeling time series in order to embed randomness in the temporal nonstationary, spatially variable, but regular, learnable patterns of real-life events [250]. The use of HMMs in speech processing has been very extensive due to the convenience of handling one-dimensional data. Without delay, numerous similar works have been conducted ever since, in image classification and object recognition, always using HMMs, varying, nevertheless, techniques and methods to refine results. One-dimensional HMMs were even explored in face recognition even though the latter is an inherently 2-dimensional problem. In their work, [216] used a 1-dimensional left-to-right HMM and treated faces as 2D objects divided into two parts in a fixed succession of regions, from each region a 1D observation sequence is obtained for the sake of the automatic extraction

of features. The acquired vectors of pixel intensities form the different elements of the observation sequence. A dual 1D-HMM model was put into training, and a database with small changes in head poses and facial expressions was used. As a result, an 8-state HMM was produced matching somehow the frequent 8 distinct regions appearing in the face image, and the resulting HMM was afterward used to train another 5-state HMM where the transitions between the states have been restricted to adjacent states. This work achieved successful recognition results despite the minimal changes in the orientation of heads in the database of the image. Later refinements have been then performed in [218] as a consequence of the marginal results obtained in [216]. Efforts were later devoted to the purpose of applying a truly 2-D HMM for image classification, where this type of HMM has been used by [154] to the attempt of improving classification by context. The authors proposed a model that considers feature vectors statistically dependent through a fundamental state process considered to be a Markov Mesh, with transition probabilities conditioned on the neighboring states (both horizontal and vertical) which allows the dependency in two dimensions to be simultaneously obvious. An EM algorithm was applied to estimate the HMM parameters and the classification process implies that classes with maximum a posteriori probability are identified jointly for all the neighboring state blocks. The work on [154] used a Viterbi training along with a suboptimal algorithm in the quest of achieving polynomial-time complexity. Although HMMs were mainly developed for discrete and Gaussian data [203], diversity of applications in contexts and domains such as anomaly detection, disease diagnosis and dynamic forecasting, increased the necessity of modifying the underlying HMM model so that it efficiently suits those new data types [221].

By the same token, a dynamic texture rarely obeys Gaussian distributions. It could definitely be represented via mixed distributions (Gaussian or not), and given the convenience of latent states in dynamic texture classification and segmentation, HMMs were put to use in [201] where the latent variables were considered discrete but following an arbitrary

emission probability distribution (without further specifications of the emission probability nature). Authors proposed a model relying on the conventional Baum-Welch [19] algorithm and on the assumption that a random emission probability distribution, along with a higher-order dependency within the hidden states, will better represent the structure of a dynamic texture and encode the appearance information of the dynamic texture with the observed variables. The proposed model applied the notorious maximum-likelihood method to serve the parameters estimation process. This work has recently been refined in [202] using rather a multivariate HMM to model the neighboring pixels changing along the time, based on the notion that the texture is a region property. The recent work yielded higher classification accuracy than the traditional HMM one.

HMMs have also been approached in the goal of simultaneously modeling multiple processes, namely in a setting governed by longitudinal data. In [10], Altman presented a new class of models, Mixed HMMs (MHMMs) providing an application to data on lesion counts in multiple sclerosis patients. The proposed model extends existing HMMs in such a way that it handles data heterogeneity (different sources for data), and also allows the integration of covariates as well as the addition of random variables (in the literature *random effects*) which are themselves a natural extension of these models. MHMMs provide an efficient estimation of parameters that processes have in common and they offer flexibility in modeling correlations given the fact that they relax the assumption that the observations are independent given the hidden states. A real-life example is provided in [200] analyzing data on criminal activities in Italy. Authors attempt to model different features related to times and types of criminal activities, taking into account the effect of territorial roots in specific Italian areas on organized crimes. Different identified levels of safety conditions are represented as hidden states.

Equally esteemed adaptations of HMMs were fulfilled in the context of Facial Expressions Recognition (FER). As early as 1993, HMMs were adopted for the first time by

Samaria and Young [217] to solve a face identification problem, whereby they drew their inspiration from Rabiner [204] and his seminal speech recognition work. They applied this powerful model to extract facial features from a set of training data and use the latter to identify a set of test images. The method used a context-dependent classifier based on HMMs, where faces are deemed to be two-dimensional objects. The segmentation is done by extracting statistical facial features while face images are segmented into horizontal and statistically similar regions identified as "facial bands" [215]. Each facial band corresponds to one state in the model, and a separate left-to-right HMM is trained for each of the 20 distinct subjects considered in the experimentation. Results showed that the HMM-based approach had better results than other purely statistical methods. Thereupon, several experiments targeting FER were conducted using HMMs, namely the work of [194], in which authors picked a Gaussian density-based left-to-right HMM with three states to handle the recognition task. The three states corresponded to three different expression groups: neutral face, face in motion and a face in its stable state of each expression. The authors confirmed the effectiveness of the proposed method for some expressions but not for the entire set. Failure in handling high similarity in shape is one of the reasons for which the model needed to be improved. Reference [243] used multi-instance HMM to capture the temporary information in facial segments where images are segmented into labeled sequences (bags). Each may itself be composed of several segments regarded as instances in the bag. The training is a multi-instance learning consisting of maximizing the conditional likelihood of an image sequence giving its corresponding sequence level. The achieved results showed high effectiveness in sequence labeling and good frames locating multi-peaks sequences. In the context of an interactive computer game environment, [262] made use of Nefian et al. HMM [184] in real-time FER. The embedded HMM uses observation vectors composed of 2-dimensional Discrete Cosine Transform (2D-DCT) coefficients to form the

observation vectors rather than use pixel intensities. The proposed method helped to reduce both the training and recognition complexities of the task. Overall, none of the stated methods have attempted to specifically adapt their HMM model to handle positive vectors which constitute the main representation unit of images. In our work, the ID distribution is used for the first time as the emission probability to best suit our extracted positive vectors in the context of the FER task.

In a similar fashion, HMMs have been extensively used in the context of occupancy detection tasks, most commonly to serve the purpose of auto-controlled ventilation, energy management, security, and lately for sustainable green buildings. Starting from the fact that, in an HMM, astate is not directly visible and only the outputs (environmental manifestations e.g. CO2 level, temperature) are, [68] attempted to exploit this statistical model inits conventional form, to solve the problem of detecting occupancy levels whose counterpart is the number of hidden states. Likewise, [5, 12, 11] have estimated occupancy in the context of smart buildings by putting HMMs into practice. Authors in [5] assayed an auto-regressiveHMM to determine the number of some research laboratory attendants by analyzing the data generated from a previously deployed wireless sensors network. Results Showed that the auto-regressive HMM is more effective than the conventional HMMalgorithm, particularly with a frequent occupancy level fluctuation.

Another motive to further extend traditional HMMs is the will to efficiently model state durations. Since HMMs are modeled based on iterations in which a change of states is possible, there is a need to model the duration of these iterations based on the respective state. At each iteration, an HMM has the chance either to switch or to stay in the same state. The traditional HMMs state durations have naturally fixed geometric distributions, which, although they sometimes represent physical reality fairly well, they tend to be utterly inadequate in a wide variety of applications. [101] first introduced the variable duration HMM

whose state durations could be modeled by different types of probability distributions. Several works were then conducted in the context and [66] introduced non-stationary HMMs with state transition probabilities in a form of functions of time, modeling state durations by a given probability mass function.

Always in an attempt to ensure HMMs' best adaptivity, proportional data modeling has been a study focus in [53] which used Dirichlet mixtures as emission probabilities for the first time applied in synthetic proportional data, which has been extended in [81] afterward and applied to real-world data. Later on, [79] adapted HMMs to embed separately the Beta-Liouville distribution as well as the generalized Dirichlet mixtures as emission probabilities. Improved results have been achieved since these types of probabilities allowed the HMM to have a more flexible covariance structure [39].

Although there have been a few, works using inverted Dirichlet for machine-learning applications are yet valuable. Recently interesting results have been achieved in the field of pattern recognition using this distribution in particular to model positive vectors [24, 22, 23]. The ID distribution does not only grant higher convenience in modeling the previously stated types of data, but it also shows in the aforementioned works greater capabilities in modeling both symmetric and asymmetric data [234]. An infinite ID mixture model has been embedded in an accelerated variational framework proposed by [89] and used for an FER task. The model was first proposed in [88] and improved afterward to yield better results than the Gaussian mixture, not to mention the model's capability of handling large amounts of data.

2.3 Hidden Markov Models

Hidden Markov Models are described according to Ghahramani [110], as a ubiquitous tool to model time series data. They have been used for decades in speech recognition

systems as well as artificial intelligence and pattern recognition applications. These models are a generalization of mixture models [111]. In fact, the probability density functions over all observable states defined by an HMM, are considered as a mixture of densities defined by each state. With HMMs, we are aiming to represent probability distributions over sequences of observations, with the assumption that the observations are discrete. An observation at time t is denoted by the variable O_T.

Two main properties delineate the hidden Markov model. First, it assumes that the observation at time t is generated by some process whose state h_t is hidden from the observer. Second, it assumes that the state of this hidden process satisfies the Markov property; that is, given the value of h_{t-1}; the current state h_t is independent of all the states prior to the time $t-1$. A hidden Markov model is governed by a set of parameters that will be specified later in this paper. The task of the learning algorithm is to find the best set of state transitions and emission probabilities between the states of the model. Therefore, an output sequence or a set of these sequences is given.

To illustrate our model, we are first listing various HMM notations and enumerating the upcoming used work script.

2.3.1 HMM Notations

According to [203], for giving an ordered observation sequence $O = \{O_1, ..., O_T\}$ generated by hidden states $H = \{h_1, ..., h_T\}; h_j \in [1, K]$ where K is the number of the states, we have:

A transition probabilities matrix: $B = \{B_{ij} = P(h_t = j | h_{t-1} = j')\}$ and the emission probabilities matrix (Continuous case): $C = \{C_{ij} = P(m_t = i | h_t = j)\}; i \in [1, M]$ where M is the number of mixture components associated with state j (which can be assumed to be the same for all states without loss of generality). We define the intial probability: π_j which is the probability to start the observation sequence from the state j.

We denote an HMM as: $\lambda = \{B, C, \varphi, \pi\}$ where φ is the set of mixture parameters depending on the chosen type of mixture.

2.3.2 Structures and underlying HMM problems:

It is worth mentioning that, for continuous observation vectors, the emission probability distributions are often taken as Gaussian mixtures [203, 153, 15, 29].

Prediction and classification are the most achieved tasks using HMMs. The fulfillment of these tasks is sustained by an observation sequence given a model λ computed using a forward-backward procedure. The training problem is considered to be the most critical for most applications of HMMs, as it allows us to optimally adapt the model parameters for real phenomena [110]. Depending on the context of each application, the quantity that should be optimized during the learning process differs. So there are several optimization criteria for learning. In this paper, we focus on the Maximum Likelihood approach via Expectation-Maximization. This problem can also be considered as one of scoring how well a given model matches a given observation sequence. This may conduct to a choosing process among several competing models in such a way that the solution to the problem would be to determine the model which best matches the observations. The learning algorithm consists of finding the best set of state transitions and emission probabilities by applying the Baum-Welch algorithm in order to estimate the parameters that maximize the probability of a given set of observations, (see figure 2). Finally we need to find out the most probable sequence that generated a set of observation states as well as its mixture components. In solving this problem, a commonly known method namely the Viterbi algorithm is used. It allows the whole state sequence with maximum likelihood to be found [203].

To start off, the initial values of parameters as well as the number of the hidden states related to the studied model are set apriori. Furthermore, as stated earlier, the nature of the data fed to the model as well as the features extracted from the data, usually prompt the

choice of the emission probability distribution which will also be selected as soon as the data type is revealed.

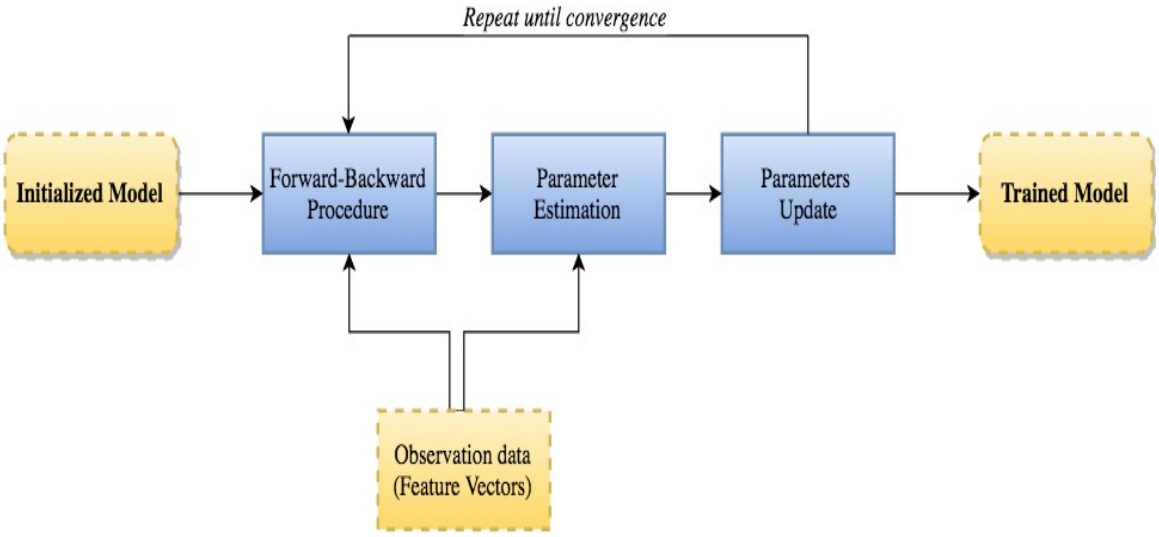


Figure 2: Hidden Markov Model training procedure

2.4 ID mixture models integration into the HMM framework

The main functionality of the observation of emission densities related to each hidden state in a given HMM is approximating randomly complex probability density functions. These emission densities had long been adopted as mixtures as an early extension of HMMs. A finite mixture model is a linear combination of a finite number of weighted standard distributions named mixture components and is defined by:

$$p(X) = \sum_{j=1}^{M} \pi_j p(X|\theta_j) \tag{1}$$

with $p(X|\theta_j)$ is a mixture component with parameter θ_j, π_j are the mixing proportions (or

also weights), and $\pi_j \in [0,1]$, $\sum_{j=1}^{M} \pi_j = 1$. M is the number of components and is a fixed quantity.

GHMMs are the most popular mixture-based HMMs using GMM as emission probabilities in the conventional HMM structure [203]. The vast reaches of Gaussian-based HMMs arise from the high capacity of GMMs to approach unidentified random distributions by embedding a simple and functional EM-based maximum likelihood model fitting framework.

In our work, the primary motivation behind the choice of adopting the inverted Dirichlet distribution as emission probabilities is the capability of this distribution to develop models specific to positive vectors. In fact, ID mixture models are flexible and able to perform in both symmetric and asymmetric modes.

Let a D-dimensional positive vector $\vec{X} = (X_1, X_2, ..., X_D)$ follow an Inverted Dirichlet (ID) distribution, the joint function is given by Tiao & Cuttman [233] as follows:

$$\mathcal{ID}(\vec{X}|\vec{\alpha}) = \frac{\Gamma(|\vec{\alpha}|)}{\prod_{d=1}^{D+1}\Gamma(\alpha_d)} \prod_{d=1}^{D} X_d^{\alpha_d - 1} \left(1 + \sum_{d=1}^{D} X_d\right)^{-|\vec{\alpha}|} \tag{2}$$

where $X_d > 0, d = 1, 2, ..., D, \vec{\alpha} = (\alpha_1, ..., \alpha_{D+1})$ is the vector of parameters and $|\vec{\alpha}| = \sum_{d=1}^{D+1} \alpha_d, \alpha_d > 0, d = 1, 2, ..., D+1$.

$\Gamma(.)$ denotes the Gamma function defined by

$$\Gamma(x) = \int_0^{+\infty} t^{x-1} e^{-t} dt \tag{3}$$

The mean and the variance of the inverted Dirichlet distribution are:

$$E(x_d) = \frac{\alpha_d}{\alpha_{D+1} - 1} \tag{4}$$

$$Var(x_d) = \frac{\alpha_d(\alpha_d + \alpha_{D+1} - 1)}{(\alpha_{D+1} - 1)^2(\alpha_{D+1} - 2)} \tag{5}$$

Choosing to change the type of the emission probability distribution, implies significant modifications in the EM estimation process.

We set notations for the quantities:

$\gamma^t_{h_t,m_t} \triangleq p(h_t, m_t|x_0, ...x_T)$: the estimates of states and mixture components, and $\xi^t_{h_t,h_t+1} \triangleq p(h_t, h_{t+1}|x_0, ...x_T)$: the estimates of local states sequence given the whole observation set, the E-step leads to $\gamma^t_{h_t,m_t}$ and $\xi^t_{h_t,h_t+1}$ for all $t \in [1, T]$ using initial parameters at the expectation step and the result of M-Step afterwards.

2.4.1 Maximum likelihood estimation

There have been a lot of approaches adopted for the sake of estimation of the mixture model. Yet, the most popular among them, and which we are going to adopt here, is the maximum likelihood estimation (MLE). MLE determines the parameters' values that maximize the inverted Dirichlet probability density function of each observed sample, and for convenience, the log-likelihood is maximized instead of the likelihood.

The M-step aims to maximize the data log-likelihood. By denoting Z as hidden variables and X as the data, we can express the data likelihood $\mathcal{L}(\theta|X) = p(X|\theta)$ by:

$$\begin{aligned} E(X,\theta) - R(Z) &= \sum_Z p(Z|X)\log(p(X,Z)) - \sum_Z p(Z|X)\log(p(Z|X)) \\ &= \sum_Z p(Z|X)\log(p(X|\theta)) \\ &= \log(p(X|\theta))\sum_Z p(Z|X)\log(p(X|\theta)) \\ &= \log(p(X|\theta)) = \mathcal{L}(\theta|X) \end{aligned} \tag{6}$$

with θ representing all the HMM parameters, $E(X,\theta)$ is the value of the complete-data log-likelihood with the maximized parameters θ, and $R(Z)$ is the log-likelihood of the hidden data given the observations.

The expected complete-data log-likelihood is:

$$E(X,\theta,\theta^{old}) = \sum_Z p(Z|X,\theta^{old})\log(p(X,Z|\theta)) \tag{7}$$

In the following, we take the case of a unique observation sequence, X, then the complete-data likelihood is expanded as

$$p(X,Z|\theta) = p(h_0)\prod_{t=0}^{T-1} p(h_{t+1}|h_t)\prod_{t=0}^{T} p(m_t|h_t)p(x_t|h_t,m_t) \tag{8}$$

where the different terms of the expression are identified as:

$$p(X, Z|\theta) = \pi_{h_0} \prod_{t=0}^{T-1} B_{h_t,h_{t+1}} \prod_{t=0}^{T} C_{h_t,m_t} ID(x_t|h_t, m_t) \tag{9}$$

We substitute Eq.(1) into Eq.(8)[203] and take the logarithm of the expression. Using the logarithm sum-product property the complete-data log-likelihood is expressed as:

$$\begin{aligned}
\log(p(X, Z|\theta)) = \log(\pi_{h_0}) &+ \sum_{t=0}^{T-1} \log(B_{h_t,h_{t+1}}) + \sum_{t=0}^{T} \log(C_{h_t,m_t}) \\
&+ \sum_{t=0}^{T} \Bigg\{ \log(\Gamma(\sum_{d=1}^{D+1} \alpha_d)) - \log(\prod_{d=1}^{D+1} \Gamma(\alpha_d)) + \sum_{d=1}^{D} (\alpha_d \log x_d) \\
&- \sum_{d=1}^{D} \log X_d - \sum_{d=1}^{D} \alpha_d \log(1 + \sum_{d=1}^{D} X_d) \\
&+ \alpha_{D+1} \log(1 + \sum_{d=1}^{D} X_d) \Bigg\}
\end{aligned} \tag{10}$$

Using Eq.(9) into Eq.(6), we can write the expected complete-data log-likelihood as:

$$\begin{aligned}
E(X, \theta, \theta^{old}) = \sum_{k=1}^{K} \sum_{m=1}^{M} \gamma_{k,m}^{0} \log(\pi_k) + \sum_{t=0}^{T} \sum_{k=1}^{K} \sum_{m=1}^{M} \gamma_{k,m}^{t} \log(C_{k,m}) + \\
\sum_{t=0}^{T-1} \sum_{i=1}^{K} \sum_{j=1}^{K} \xi_{i,j}^{t} \log(B_{i,j}) + \log \mathcal{ID}(\vec{X}|\vec{\alpha})
\end{aligned} \tag{11}$$

2.4.2 Update equations of HMM and ID parameters estimation

Here we maximize the expectation of the complete-data log-likelihood with respect to π, B, and C, taking into account the constraints due to the stochastic nature of these parameters,

on this wise, we introduce Lagrange multipliers and the resulting update equations are:

$$\pi_k^{new} \propto \sum_{d=1}^{D} \sum_{m=1}^{M} \gamma_{k,m}^{0,d} \tag{12}$$

$$B_{i,j}^{new} \propto \sum_{d=1}^{D} \sum_{t=0}^{T_d-1} \xi_{k,k'}^{t,d} \tag{13}$$

$$C_{k,m}^{new} \propto \sum_{d=1}^{D} \sum_{t=0}^{T_d-1} \gamma_{k,m}^{t,d} \tag{14}$$

where $k, k' = \{1, ..., K\}$, and $m = \{1, ..., M\}$.

2.4.3 Estimation of ID parameters

The most common strategy for maximum likelihood inference relies on the Expectation-Maximization (EM) algorithm [63]. We use this algorithm to maximize $\log \mathcal{ID}(\vec{X}|\vec{\alpha})$. In this section, we deploy the maximum likelihood estimates of the parameters of our model and we will afterwards give the estimation algorithm. It is worthwhile to mention that the estimation process in [24] is the first work considering and applying the inverted Dirichlet distribution in a practical setting and for modeling positive vectors. Therefore, the estimation process of the emission probabilities parameters is in itself similar to the one performed in [24], however in our case the integration of the inverted Dirichlet distribution into the framework of hidden Markov models is performed for the first time and its estimation represents an important step in the whole parameter estimation task implied by HMMs.

To solve this problem, we must determine the solution to $\frac{\partial \log \mathcal{ID}(\vec{X}|\vec{\alpha})}{\partial \alpha_{k,m,d}} = 0$, taking into account that we are estimating each ID distribution separately, which simplifies the

equations. We calculate the derivative with respect to $\alpha_{k,m,d}, d = 1, ..., D$ we obtain:

$$\frac{\partial \log \mathcal{ID}(\vec{X}|\vec{\alpha})}{\partial \alpha_{k,m,d}} = \gamma_{k,m}^{t} \Psi_0(\sum_{d=1}^{D} \alpha_{k,m,d}) - \gamma_{k,m}^{t} \Psi_0(\alpha_{k,m,d}) - \log\left(\frac{x_d}{1 + \sum_{d=1}^{D} x_d}\right) \quad (15)$$

where $\Psi_0(.)$ is the digamma function. The derivative with respect to $\alpha_{k,m,D+1}$ is given by

$$\frac{\partial \log \mathcal{ID}(\vec{X}|\vec{\alpha})}{\partial \alpha_{k,m,D+1}} = \gamma_{k,m}^{t} \Psi_0(\sum_{d=1}^{D} \alpha_{k,m,D+1}) - \gamma_{k,m}^{t} \Psi_0(\alpha_{k,m,D+1}) - \log\left(\frac{1}{1 + \sum_{d=1}^{D} x_d}\right) \quad (16)$$

There is no closed-form solution to estimate our parameters. Consequently, we will use an iterative approach, namely the Newton-Raphson method [156]. We give the global estimation equation for a single state:

$$\alpha^{new} = \alpha^{old} - H^{-1} G \quad (17)$$

where H is the Hessian matrix associated to $\log \mathcal{ID}(\vec{X}|\vec{\alpha})$ and G is the first derivatives vector, $G = \left(\frac{\partial \log \mathcal{ID}(\vec{X}|\vec{\alpha})}{\partial \alpha_{k,m,d}}, ..., \frac{\partial \log \mathcal{ID}(\vec{X}|\vec{\alpha})}{\partial \alpha_{k,m,D+1}}\right)^T$. To calculate the Hessian of $\log \mathcal{ID}(\vec{X}|\vec{\alpha})$ we need to calculate the second derivatives:

$$\frac{\partial^2 \log \mathcal{ID}(\vec{X}|\vec{\alpha})}{\partial^2 \alpha_{k,m,D}} = \gamma_{k,m}^{t} \Psi_1(\sum_{d=1}^{D} \alpha_{k,m,D+1}) - \gamma_{k,m}^{t} \Psi_1(\alpha_{k,m,D}) \quad (18)$$

$$\frac{\partial^2 \log \mathcal{ID}(\vec{X}|\vec{\alpha})}{\partial \alpha_{k,m,d_1} \partial \alpha_{k,m,d_2}} = \gamma_{k,m}^{t} \Psi_1(\sum_{d=1}^{D} \alpha_{k,m}) \quad (19)$$

where $\Psi_1(.)$ is the trigamma function.

It is noteworthy to mention that, in our context, each inverted Dirichlet distribution is estimated separately by fixing k and m, which would simplify the equations.

.

$$H = \overline{\gamma} \times \begin{pmatrix} \Psi_1(\sum_{d=1}^{D+1} \alpha_d) - \Psi_1(\alpha_1) & \Psi_1(\sum_{d=1}^{D+1} \alpha_d) & \dots & \Psi_1(\sum_{d=1}^{D+1} \alpha_d) \\ \Psi_1(\sum_{d=1}^{D+1} \alpha_d) & \Psi_1(\sum_{d=1}^{D+1} \alpha_d) - \Psi_1(\alpha_2) & \dots & \Psi_1(\sum_{d=1}^{D+1} \alpha_d) \\ \vdots & \ddots & & \vdots \\ \Psi_1(\sum_{d=1}^{D+1} \alpha_d) & \dots & & \Psi_1(\sum_{d=1}^{D+1} \alpha_d) - \Psi_1(\alpha_D + 1) \end{pmatrix} \tag{20}$$

In that manner we can write the Hessian H as follows:

$$H = D + \delta A A^T \tag{21}$$

where $D = diag[-\overline{\gamma}\Psi_1(\alpha_{D+1})]$: is a diagonal matrix [74], $\delta = \overline{\gamma}\Psi_1(\sum_{d=1}^{D+1} \alpha_d)$, and $A^T = (a_1, ..., a_{D+1}), a_d = 1, d = 1, ..., D+1$.

We then have:

$$H^{-1} = D^{-1} + \delta^* A^{*^T} A^* \tag{22}$$

in such a way that D^{-1} can be easily computed and

$$A^* = \frac{-1}{\overline{\gamma}} \left(\frac{1}{\Psi_1(\alpha_1)}, ..., \frac{1}{\Psi_1(\alpha_D + 1)} \right) \tag{23}$$

$$\delta^* = \overline{\gamma}\Psi_1(\sum_{d=1}^{D+1} \alpha_d) \left[\Psi_1(\sum_{d=1}^{D+1} \alpha_d) \sum_{d=1}^{D+1} \frac{1}{\Psi_1(\alpha_d)} - 1 \right] \tag{24}$$

where $\overline{\gamma} = \sum_{d=1}^{D+1} \sum_{t=1}^{T} \gamma^{d,t}$ is the cumulative sum to the state estimates of the observation sequence, in the case where more than one observation sequence are available. Once we obtained H^{-1} and G, the Newton-Raphson estimation method is applied over (15) in order to update the parameters of the inverted Dirichlet mixture.

2.4.4 Inference on hidden states

Given our $\mathcal{IDHMM}$ model and observed sequences, there is a combination of several interesting inferences that could be done regarding the hidden states. As stated earlier, the estimates of the states (γ_k, ξ_k), obtained after a forward-backward procedure [204], will serve us in the M-Step as we need each time (iteration) to compute the data likelihood quantity and the difference between the former and the current of each iteration of all subsequent samples. The use of the forward algorithm has a huge impact on the computational complexity of the model. Using dynamic programming, it allows storing and reusing the results of partial computations. In fact, this algorithm permits dealing with a complexity $\mathcal{O} = \mathcal{N}^2\mathcal{T}$(with $\mathcal{N}$ the number of states and $\mathcal{T}$ is the length of the observation sequence) instead of dealing with a $\mathcal{O} = \mathcal{N}^{\mathcal{T}}\mathcal{T}$ exponential computing complexity when adopting the non-trivial method.

We inspire from the backward part of the Forward-Backward algorithm to develop our EM algorithm. The estimation part starts by assigning values to the estimates. Thus, good starting values are much needed to help us find the optimal solution in a reasonable time. From this perspective, we chose to test a large number of initial values to avoid falling in a poor local maximum.

As an initialization method, we combined the K-means algorithm and the method of

moments [117]. In our $\mathcal{IDHMM}$, we have $K \times M$ inverted Dirichlet components where K is the number of groups (HMM states) in which we find M inverted Dirichlet components. In a similar way as in [53], we are initializing the parameters in a way where we are first discarding the temporal constraints in order to focus on each single inverted Dirichlet distribution and finding the best transition probabilities afterward.

2.4.5 Model evaluation

To evaluate algorithm convergence, we should iterate until no or little increase in data likelihood is observed. We, therefore, ought to consider defining a certain threshold that has to be met, and at which the algorithm stops, and the obtained parameters values are retained and saved for the HMM. This threshold is a predefined value that will at each iteration be compared to the difference between the former and the current data likelihood. The latter is maximized by the means of its lower band, it's given by $E(X, \theta, \theta^{old}) - R(Z)$:

$$\begin{aligned} p(Z|X) &= p(h_0)p(m_0|h_0)\prod_{t=1}^{T} p(h_t|h_{t-1})p(m_t|h_t) \\ &= p(h_0)\frac{p(m_0,h_0)}{p(h_0)}\prod_{t=1}^{T}\frac{p(h_t,h_{t-1}p(m_t,h_t))}{p(h_{t-1})p(h_t)} \end{aligned} \tag{25}$$

We denote $\eta_t \triangleq p(h_t|\overrightarrow{x})$. In the case of a single observation and after necessary derivations (not stated here, see details in [53]) we have the following expression :

$$\begin{aligned} R(Z) = \sum_{k=1}^{K}\left[\eta_k^0 log(\eta_k^0) + \eta_k^T log(\eta_k^T) - 2\sum_{t=0}^{T}\eta_k^t log(\eta_k^t)\right] \\ + \sum_{t=0}^{T}\sum_{m=1}^{M}\sum_{k=1}^{K}\gamma_{k,m}^t log(\gamma_{k,m}^t) + \sum_{t=0}^{T-1}\sum_{k=1}^{K}\sum_{k'=0}^{K}\xi_{i,j}^t log(\xi_{i,j}^t) \end{aligned} \tag{26}$$

If data are completely observed, a sum operator has to be added to the whole expression

and η's have to be computed for each sample.

The way to determine the values of initial parameters as well as choosing the number of mixture components is distinguished after going through a set of experimental trials before landing on the most effective results. One of them is regression clustering RC combined with EM (LinReg-EM) [257], which we adopted at first as an initialization tool for its ability to simplify complex distributions into simpler ones by applying regression functions to the data in order to get "K subsets" with a much simpler distribution and therefore reduce complexity. Nonetheless, this method appeared to be much more time-consuming when compared to K-means, not to mention its weakness when dealing with high dimensional data where K-means applied along with the method of moments, obviously outperformed RC.

2.5 Experimental Results

In this section, we fulfill a validation of our model, by appraising the performance of the latter throughout different real-life contexts. The first two are, respectively, image texture categorization and dynamic textures classification. The third application is facial expression recognition and the last application is occupancy detection related to smart building energy management. In our applications, we opt for a random initialization for parameters π, B and C, and we base our convergence test in conjunction with our learning algorithm, on the variation of the data likelihood (Eq. 24), setting $\epsilon = 10^{(-3)}$ [1]. In the following, recognition accuracies and recognition rates are computed to evaluate the efficiency of our model, they designate the percentage of the correctly identified images into their natural classes within the dataset.

[1] Tuned to 10^{-6} later on, after several experimental manipulations

2.5.1 Image texture categorization

In the world of computer vision and image processing, the more images represent uniform intensities, the more they tend to be easily interpreted and categorized. Nevertheless, when dealing with images of real objects, those dealt with hardly exhibit regions of uniform intensities. By way of illustration, let us consider an image of a wooden surface, it is not uniform, but we can easily spot variations of intensities forming some repeated patterns named visual texture. Some physical surface properties like roughness, oriented strands and reflectance differences resulting from colour or light, may generate some sort of patterns that often have a tactile quality and recall patterns that draw in some sort of spatial dynamics. It is indeed a challenging task to let a model learn how to assimilate these textures. Therefore, we are using HMMs whose utility has been brought to light since the good old days [199] thanks to their capabilities to unravel latent structures of textures that direct observations could not provide. Seeing that the problem could be approached as such, we chose to adopt one popular pattern representation approach namely the Bag-of-Words (BoW) method, adapted by [249] for scene classification, in which Yang et al. mapped the key points of an image into visual words. Thus, an image could be represented as a "*bag of visual words*" BoVW, and more expressly, as a vector of counts of each visual word in that image that could later serve as a feature vector in the classification task.

2.5.1.1 Adopted methodology

To apply our model and validate its performance we choose to work on highly impacting datasets namely the UIUC and UMD natural textures images datasets, respectively [146, 248] (see Figures 3 and 4), and the very notorious CUReT benchmark [59]. For the first two datasets, we choose to combine the BoVW strategy along with a SIFT [164] descriptor to obtain our feature vectors from the images and build our texton-dictionary [137] [152]. The images are represented as histograms over a textons dictionary. We first started by using

dense SIFT descriptors of 6 x 6 pixels patches and descriptors are sampled every two pixels, at scales $2^{i/3}, i = \{0, 1, 2, ...\}$. Next, for the encoding, we map out the extracted descriptors to prepare the classification task that has been performed by a K-means clustering algorithm for each image to help lower the number of features in a bag of visual words containing 60 elements.

The UIUC texture dataset is composed of 25 texture classes with 40 images per class. Textures are viewed under significant scale and viewpoint changes. We choose to train our model by picking $I = \{10, 20, 25\}$ randomly selected images from each of the classes as a training set. The acquired features are further vector quantized using K-means and c resulting texton cluster centers are thus computed [71]. Each image is getting a feature vector of dimension 8 for one pixel, each concurrent feature is inspected in the dictionary so that the closest one is spotted and the pixel is labeled with that texton. This operation results in a series of G histograms used to represent the image.

Figure 3: Sample images from the UIUC data set

To illustrate how our model $\mathcal{IDHMM}$ is dealing with the treatment of the data, we need to point out that we are exploring this problem in a similar manner as [199] and [81], who both consider that an image can be modeled by a statistical process, namely HMM, with hidden states (here we allude to pixels), that has a certain probability distribution over other observable states neighbouring to the hidden one, and who determines that state. Accordingly, we model each class of texture by an HMM considering its corresponding series of histograms. Varying the number of states K, and mixture components M, we observe how the model will function with respect to the $K \times M$ product and quantify its performance using the nearest neighbour (NN) classification. In this work, we are using a 1-Nearest Neighbour 1-NN classifier. The main idea is to assign each image to its respective class. Since classes are represented as histograms, we are comparing the histogram describing each image to the centroid histograms of all classes and we are choosing only the closest to the tested image. Thus, each image is assigned to the corresponding class. This will grant a lower bias to this operation given the fact that we are fitting our model to the 1-Nearest point. For two histograms $Y1 = (a_1, ..a_g)$ and $Y2 = (b_1, .., b_g)$ comparison, we use the normalized χ^2 (chi-square) distance [189] defined by:

$$D_{(Y1,Y2)} = \frac{\sum_{g=1}^{G} \left[(a_g - b_g)^2/(a_g - b_g)\right]}{2} \tag{27}$$

to measure the similarity between two discriminant histograms.

Figure 4: Sample images from the UMD data set

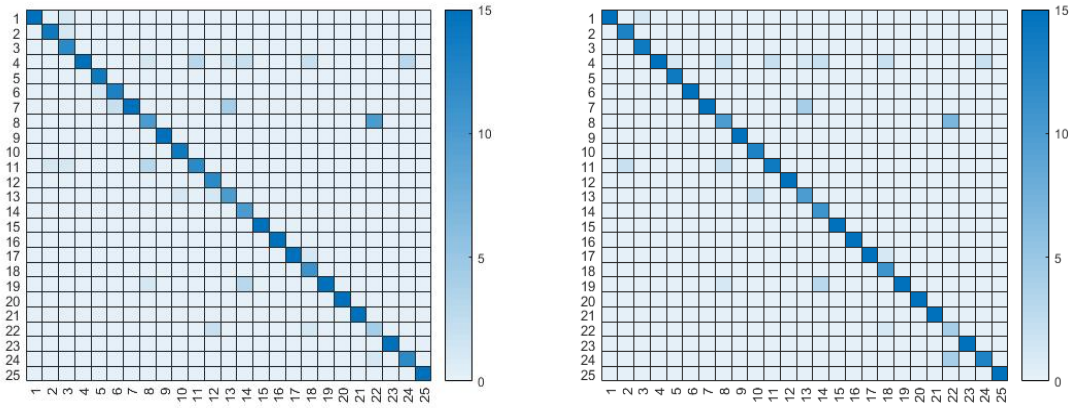
The same pre-processing strategy is adopted for the UMD dataset which is composed of 1000 images of size 1280x960 pixels, also divided into 25 different classes with 40 images in each class. It has been converted to the same resolution as the UIUC one for the sake of results comparison.

Figure 5: Sample images from the CUReT data set

The CUReT database is composed of 61 classes of natural physical textures taken at different levels of illumination, acquired from different viewpoints and varying viewing angles 5. Although each class contains 205 images, only 118 are captured at an angle less than 60 degrees and only 92 out of these from which a region of size 200×200 can be successfully cropped. Therefore we are adopting a commonly used preprocessing methodology, in a similar manner as in [132, 238, 185, 146]. 92 images from each of the 61 classes are used as our subset with a total of $92 \times 61 = 5612$ images are taken into account in this study. The same preprocessing scheme is adopted with the CUReT dataset, with only a slight change in the image description which is here done by two-dimensional texton histograms encoding the joint distribution of all neighboring pixels respectively in a patch of 6×6 pixels and 10×10 pixels.

2.5.1.2 Results and discussion

In Fig.19 we present the confusion matrices for the UIUC dataset. For the conducted tests on respectively 15, 20 and 25 random selected images with, every time 30 algorithm runs, we wanted to spot any increase in the average[2] recognition accuracy presented in Fig.2.5.1.2, along with a comparison with a previously developed HMM with Gaussian mixtures as emission probabilities, and investigate whether or not this could be attributed to the choice of the best K and M combinations. This has been proved true when in all of the experiments, as long as the product $K \times M$ is verifying the following inequation $6 \leqslant K \times M \leqslant 12$, the average accuracy of recognition is identical regardless of the selected number of images I. Average accuracies for both UIUC and UMD datasets are presented in table 1. The $\mathcal{IDHMM}$ model yielded a 97.51% average recognition accuracy whereas the DHMM nailed 95.42% and GHMM achieved 89.91%.



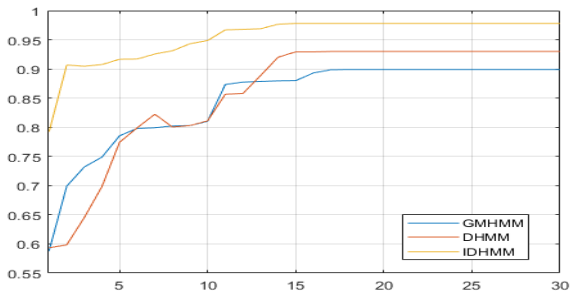
(a) $I = 15$ (b) $I = 25$

Figure 6: Confusion matrices for $\mathcal{IDHMM}$ using respectively $I = 15$ and $I = 25$ random selected images from UIUC dataset

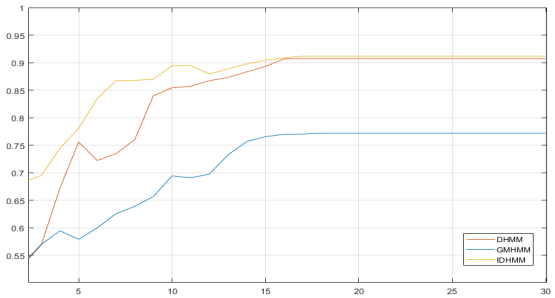
We can thus deduce that a convenient model tuning, particularly selecting the number

[2]Average accuracy from tests of 15, 20 and 25 randomly selected images from each texture.

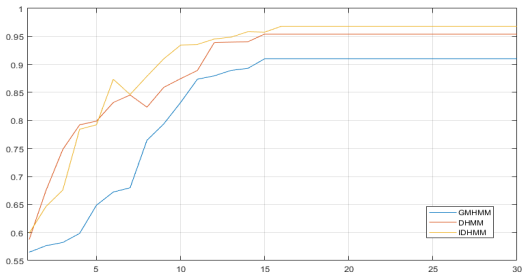
of states along with the number of each inverted Dirichlet mixture component per state, is a suitable way to boost the accuracy of such task [79]. With merely equal importance, we have also noticed that the initialization of the estimation algorithms is subject to big in-depth experimentation to further increase the recognition performance of the proposed model. In fact, since those models are arising from a vast transition parameter space, they require careful training to delay their convergence. We observed that the larger the stop parameter is, the more time is allowed for the algorithm to train and to well estimate the HMM parameters before delivering the maximal results, and for this particular reason we updated our threshold to $\epsilon = 10^{-6}$.



(a) $Database : UIUC$



(b) $Database : UMD$



(c) $Database : CUReT$

Figure 7: Average accuracies for (a) UIUC database, (b) UMD database and (c) CUReT database

Our IDHMM is among the rarest models based on HMMs who attempted to work on the very challenging CUReT dataset. The latter has long been used by researchers in classification contexts as well as image segmentation. However, taking advantage of this database

to validate this work has been itself an achieved goal since there were no available works using HMM that opted for this dataset. Results showed promising average accuracy for our IDHMM. GHMM, DHMM and IDHMM nailed respectively 91.01%, 93.11% and 96.82%. Also, compared to the results yielded in [258] and [146] on both UIUC and CUReT datasets with respectively 97.57% and 95.22% for [258] and 95.22% for the CUReT database for [146], our proposed method achieves state of the art accuracies with 97.51% for UIUC, 91.2% for UMD and 96.82% for CUReT. These results show the effectiveness of the proposed model to handle positive vectors carried by images in the natural texture context. This method permits a targeted interpretation of this type of data which increases the capacity of the model to focus on the positive data above all and thus shows a significant potential power of the IDHMM in modeling positive vectors. Moreover, these results might be further improved with more appropriate features and optimized parameters.

Table 1: Average recognition accuracies for different used HMMs

Method	UIUC dataset	UMD dataset	CUReT dataset
Zhang et al. [258]	**97.57%**	-[3]	95.22%
Lazebnik et al. [146]	95.54%	**96.29%**	-[4]
GHMM	89.91%	77.17%	91.01%
DHMM	93.11%	90.7%	95.42%
$\mathcal{IDHMM}$	97.51%	91.2%	**96.82%**

[3,4]No available record of the corresponding experimentations

2.5.2 Dynamic texture recognition

An equally challenging application is dynamic texture recognition which is an extension of texture to the temporal domain. Dynamic textures are sequences of images moving

containing scenes with certain stationary properties [69](candle, sea waves, foliage, etc.). Every single image in the moving scene is represented as an array of positive numbers that depend on the motion, shape, and pose of the scene. Dynamic or 3D images are also subject to other properties that help define their nature. These properties are reflectance and light distribution. Dynamic textures have been the focus of much research in the field of computer vision to serve tasks such as texture categorization, motion classification, and video registration [87, 105], and model-based methods are the most widely used ones to achieve these tasks [85, 207, 202].

2.5.2.1 Adopted methodology

We are approaching the dynamic texture categorization problem by adopting a BoVW inspired methodology. Thus, for a given video sequence of a single dynamic texture, a class where the video sequence belongs to will be identified. The extraction of feature descriptors from the video sequences is done by applying the dense sampling approach. In fact, in this application, we focus on describing and recognizing motions of objects such as water, where every point is as important as any other point, which explains our choice of applying the dense sampling over the *interest point approach*. To put it differently, instead of selecting certain "interesting" pixels satisfying a particular selected criterion and describe features using a descriptor, we chose to make use of the dense sampling method where a specific video sequence is divided into regular-sized spatiotemporal volumes, each described using a feature descriptor. Therefore, we use all extracted information without ignoring any regions in the video. This technique has been proved to work better on image categorization than the one based on interest points [147].

We divide video sequences into equal-sized spatiotemporal patches. We model those patches using a Linear Dynamic System (LDS) [49] to form a feature descriptor. Once extracted, the descriptors are clustered with a K-means algorithm to form a codebook,

with cluster centers representing codewords in the codebook. Hence a set of histograms of "visual words" are formed with respect to the codewords distribution in the image. For each patch (spatiotemporal volume) an HMM is then trained. This procedure will result in: A codebook of W codewords as a vocabulary to represent each patch and G histograms to represent the image.

A histogram is represented as $Y = [y_1, y_2, ..., y_W]^\top \in \mathbb{R}_+^W$. We designate a codeword w that occurs occ_{iw} times in the ith patch, P the total number of patches and P_w the number of patches in which the codeword w occurs at least once. We thus have the following *Term Frequency (TF)* representation:

$$y_{iw} = \frac{occ_{iw}}{\sum_{w=1}^{W} occ_{iw}} \tag{28}$$

with $w = 1, ..., W$ and $i = 1, ..., P$

In a similar way as in Section 2.5.1.1, a NN classification is used to quantify the performance of the model, along with a χ^2 distance to determine the distance between two histograms.

2.5.2.2 Results and discussion

We choose to test the model on the DynTex dataset [198] (See figure 8). In the original dataset, all dynamic textures are colour image sequences. In this experiment, we consider a subset of 120 image sequences of 6 selected classes from the integrality of the dataset without a special preference. A video sequence is first converted to grayscale intensity, cropped into non-overlapping patches of size 20 x 20 x 25 each modeled using an LDS of order 3. We use a randomly selected half of the dataset for the vocabulary construction and model training and the other half for testing. The initial number of states K and mixture components M is determined using a K-means clustering of the training data, with the number of clusters varying from 2 to 25. As represented in table 2, we can see that

whenever the number of clusters goes underneath 6 or exceeds 12 clusters the recognition accuracy tends to decrease showing that the best results can be achieved when the product $K \times M \in [6, 12]$ with 2 being the maximum tolerated number of states. This can be related to the fact that a higher-order HMM might result in an important loss of information between the states. Thus, as long as the number of states keeps growing, the data is scattered and the model tends rapidly to lose track of pertinent information. The reduction in recognition rates is also conspicuous when larger subsets and hence more hidden states are added, as a result of the growing complexity of the model conducting to a more singular training and classification procedure.

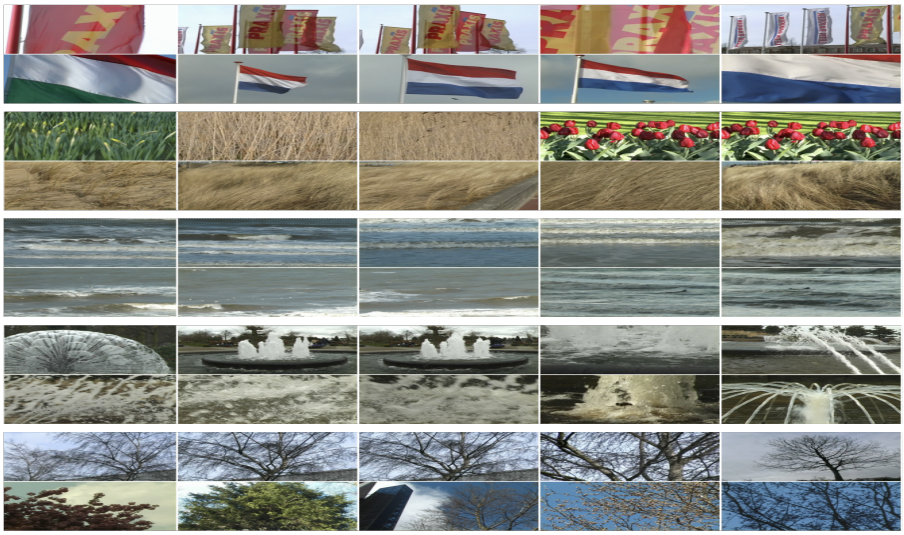


Figure 8: Sample images from the DynTex dataset

Table 2: The average recognition rate for different mixture models

Clusters ($K = 2 \times M$)	$\mathcal{IDHMM}$ recognition rate	DHMM recognition rate	GHMM recognition rate
2	0.36	0.36	0.34
4	0.52	0.47	0.35
6	0.80	0.69	0.61
8	0.84	0.71	0.72
10	**0.98**	**0.81**	**0.79**
12	**0.98**	**0.89**	**0.81**
14	0.76	0.59	0.58
16	0.47	0.38	0.31

After 30 times run, the confusion matrices are computed and the recognition rate for the $\mathcal{IDHMM}$ model is 98% with the best HMM configuration being $K = 2$ states and $M = 5$ mixtures associated with each state. The results obtained with an $\mathcal{IDHMM}$ are indubitably better than those obtained with the GHMM. In fact, the convergence of $\mathcal{IDHMM}$ is faster than the GHMM model.

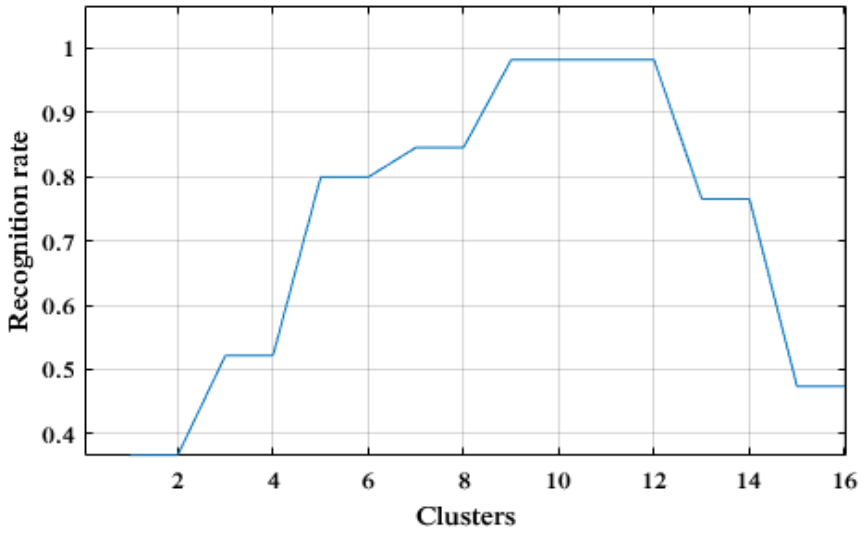


Figure 9: Recognition rate fluctuation with respect to the number of states for the $\mathcal{IDHMM}$

Apart from the fixed number of states experiment (same number of hidden states for each dynamic texture), we also investigate the impact of varying the number of states on the recognition capacity of our model. We make use of the Jaccard similarity index (JI) [231] to evaluate this capacity for the considered subsets. The JI is defined as follows:

$$\mathcal{J}_{s,r} = \frac{\mathcal{R}_{s,r} \cap \mathcal{P}_{s,r}}{\mathcal{R}_{s,r} \cup \mathcal{P}_{s,r}} \tag{29}$$

with $\mathcal{R}_{s,r}$ is the true class of texture r at sequence s (set of spatiotemporal patches defining the type of the texture), and $\mathcal{P}_{s,r}$ is the predicted class of texture at sequence s. We use the mean JI over all dynamic texture classes for all sequences, to assign the same weight to each texture type. The mean Jaccard Index is expressed as follows:

$$\overline{\mathcal{J}} = \frac{\sum_{r=1}^{R} \mathcal{J}_r}{R} \tag{30}$$

with $\mathcal{J}_r$ is the JI of each texture and $r = 1, ..., R$ is the total number of texture types.

We then allow the model to analyze the data by letting it iterate 30 times, with a random number of states from 2 to 8 to be adopted each time. Table 2 shows this impact on the JI and hence the recognition accuracy with respect to the picked number of states.

It is clear that the maximum accuracy is achieved with 2 being the maximum number of hidden states in the HMM. Accordingly, the choice of the number of states is critical to the model efficacy because a very large or very small picked number could drastically affect the HMM and result in bad recognition. Therefore, choosing a fixed number of states grants greater accuracy than a variable state HMM.

2.5.3 Facial expressions recognition

Recognizing facial expressions is firmly linked to the way we interact with other people. It represents the most powerful medium for non-verbal communication in everyday human interaction and plays a major role in understanding people's intentions and feelings. Therefore, substantial efforts have been devoted to automating this recognition and using it as a fundamental step within multiple decision-making systems. Admittedly, the task of interpreting facial expressions might be a natural and effortless task for a human being. However, automating this task remains a complex and challenging process. Facial Expression Recognition is applied in a wide range of contexts and is used in numerous applications such as Human-Computer Interaction, student automatic E-learning, Behavioural Science, psychological studies, image understanding, and synthetic face animation. Among the variety of techniques and methods adopted for face recognition and FER, HMM has made a significant impact in solving this task since the early 1990 [215]. HMMs are used to subtly recognize distinct facial expressions disclosed when experiencing a particular feeling. They offer a representation of the statistical behavior of an observable symbol sequence by specifying the probability distribution over all hidden events that are behind the said

sequence. Furthermore, an HMM is able to yield adequate performances in the spatiotemporal domain especially when it deals with an entire sequence of images describing a group of actions taken by a person when undergoing a certain feeling. The use of HMM in FER owes its success to the analogy between human performance when naturally processing the recognition task and to the stochastic nature of the HMM process inasmuch as it analyses the measurable (observable) actions in order to infer the immeasurable (hidden) feelings of the person.

2.5.3.1 Adopted methodology

The main proceeding when applying HMMs for FER is to match image templates to a chain of states of a doubly embedded stochastic model. In the architecture of the proposed model presented in 10 we use a single HMM to recognize each of the different emotions from the database. Here, for the sake of validation, we choose to apply our IDHMM on the challenging Dollar facial expression database [67] 16.

To diminish the background influence and extract features from the whole face region we start off the preprocessing step by cropping original face images into 110×150 pixels and simply keep the central part of facial extraction. Then, we use a Local Binary Pattern (LBP) descriptor [190] for feature extraction. Here, the LBP histograms are extracted from the local facial region and used as an entity for the regional description as we choose to extract and concatenate LBP features altogether into a single histogram. Thus, we divide each face image into small regions of 18×21 pixels selecting the 59-bin LBP 8.2 operator. Images will be then divided into 42 (6×7) regions and represented by the LBP histograms with a length of 2478(59×42). The procedure is applied as in [223]. Next, to reduce the size of the feature vectors to 128, we apply the probabilistic latent semantic analysis (pLSA) model [124].

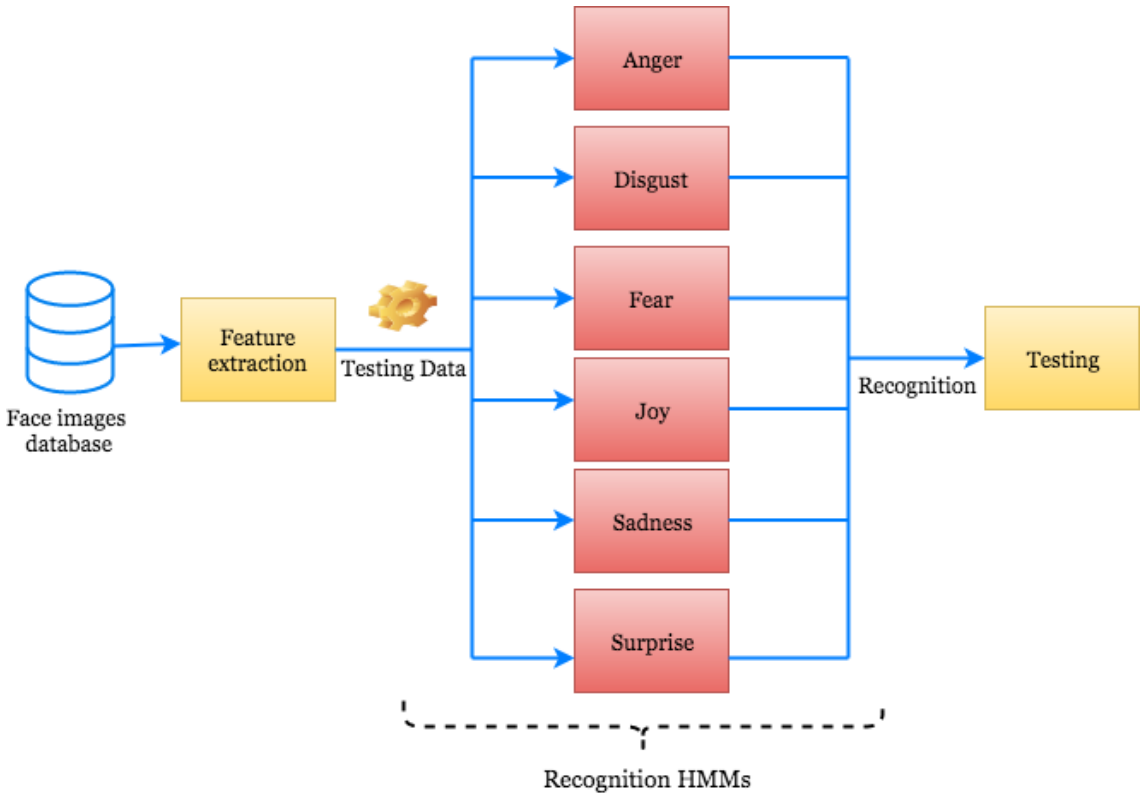


Figure 10: Model architecture for Facial Expression Recognition

2.5.3.2 Results and discussion

The Dollar facial expression data set is composed of 192 image sequences carried out by 2 subjects each expressing 6 different emotions under 2 lighting conditions. We use three peak frames of each sequence for 6-class expression recognition (576 images: Anger, Disgust, Fear, Joy, Sadness, and Surprise). Therefore, the characterization for the initial instance of the proposed model involves a number of states $K = 3$ and a number of ID mixture components $M = 3$ which we will then test during the training process. Tables 3 and 4 bring to light the obtained results. We examine the recognition rate for all expression types combined, and also in each of the expression types separately. The general average recognition rate has been reached after producing 40 trials on the randomly chosen sample. It is worthwhile noting that only 10 trials using all images in each class were performed

for each expression type separately. There is a significant shift between our IDHMM, and both the conventional Gaussian mixture-based HMM but also the DHMM with the same test setting and characterization.

Figure 11: Samples of facial frames from the Dollar facial expressions dataset

Table 3: Average recognition rates for different used HMMs

Method	Average Recognition Rate (%)
GHMM	88.06
DHMM	94.41
IDHMM	**96.11**

As we can see in table 3, the average accuracy for the IDHMM accomplishes considerably better values than the GHMM and DHMM (96.11% for IDHMM against 88.06% for GHMM and 94.41% for DHMM). Again, IDHMM demonstrates that it can definitely outperform the widely used GHMM along with the recently adopted DHMM in the context of image processing applications mainly given the fact that we are dealing with images that could easily be interpreted into histograms, which are nothing but positive vectors, and thus when fed to the ID-based HMM they are interpreted in the most proper manner thanks to the nature of this distribution.

Table 4: Average recognition rates (percentage %) for different Expression types

Expression Type	GHMM	IDHMM	DMM
Anger	88.17	97.15	96.06
Disgust	87.03	93.22	91.01
Fear	89.11	96.31	92.40
Joy	94.02	98.71	95.88
Sadness	92.0	95.2	92.68
Surprise	87.63	96.09	94.55

2.5.4 Estimating occupancy in an office setting

2.5.4.1 Problem statement and adopted methodology

Being a remarkable part of offices and homes automation, indoor occupancy forecasting is a piece of very important input information when it comes to systems' self-remoting of multiple environmental settings such as heating, ventilation, air-conditioning (HVAC) [73, 191] and lightening [45]. Automating these settings has proved to be very efficient since studies showed that around one-third of energy consumed in buildings can be saved using occupancy-based control [82, 44].

There has been an extensive focus on occupancy detection systems and their modeling through all sorts of methodologies seeking mainly to satisfy privacy by avoiding the use of cameras and voice recorders as much as possible as well as the high prediction accuracies in both closed and open spaces. Pyroelectric infrared (PIR) sensors, ultrasonic sensors and a combination of 8 occupant sound-wave detecting microphones, are all techniques used respectively in [163, 236, 237]. There is a subtle preference recently in using environmental parameters to estimate occupants' presence indoors, namely temperature, humidity,

pressure and CO_2 concentration [11], especially since those sensors are non-intrusive and can easily be deployed or integrated into the (HVAC) system. Furthermore, machine learning techniques such as Artificial Neural Network (ANN), Support Vector Machines, and HMMs [45, 68, 12] have been exploited to extract features from the environmental parameters and have been proved efficient in constructing a relationship between those features and the number of occupants.

In this work, we apply our $\mathcal{IDHMM}$ model to estimate occupancy in an office setting and hence be the first to tackle this problem with an inverted Dirichlet-based HMM. We perform the occupancy detection task only by exploiting low-cost non-intrusive environmental sensors and knowledge to provide meaningful estimations.

A. Testbed

The testbed is an office in Grenoble Institute of Technology, housing four people. Visitors often attend meetings and perform presentations in the office throughout the week. The sensors network is composed as follows:

• A network structured by different sensors measuring luminance, CO_2 concentration, relative humidity (RH), temperature, motion, power consumption, window and door position and acoustic pressure from a microphone (or simply microphone). An EnOcean protocol handles data sending.

• The recuperation of the data is performed continuously and the latter is stored in a regular manner in a centralized database with a web application for retrieval.

• 2 video cameras to record real occupancies and activities intended only for validation.

The occupancy in the office, as well as the matching environmental conditions manifestations, are modeled via an HMM where the observable states are the retrieved measurements that allow us to have certain information about the hidden states which are represented by the number of occupants that we desire to determine. Features are attributes from multiple sensors accumulated over a time interval and are grouped into sets forming a feature vector. Thus, the model efficacy will be determined by the effectiveness of the feature selection. Results presented in this work are based on a period of time $T_s = 30$ minutes ($T_s = 1quantum$).

B. Practical study

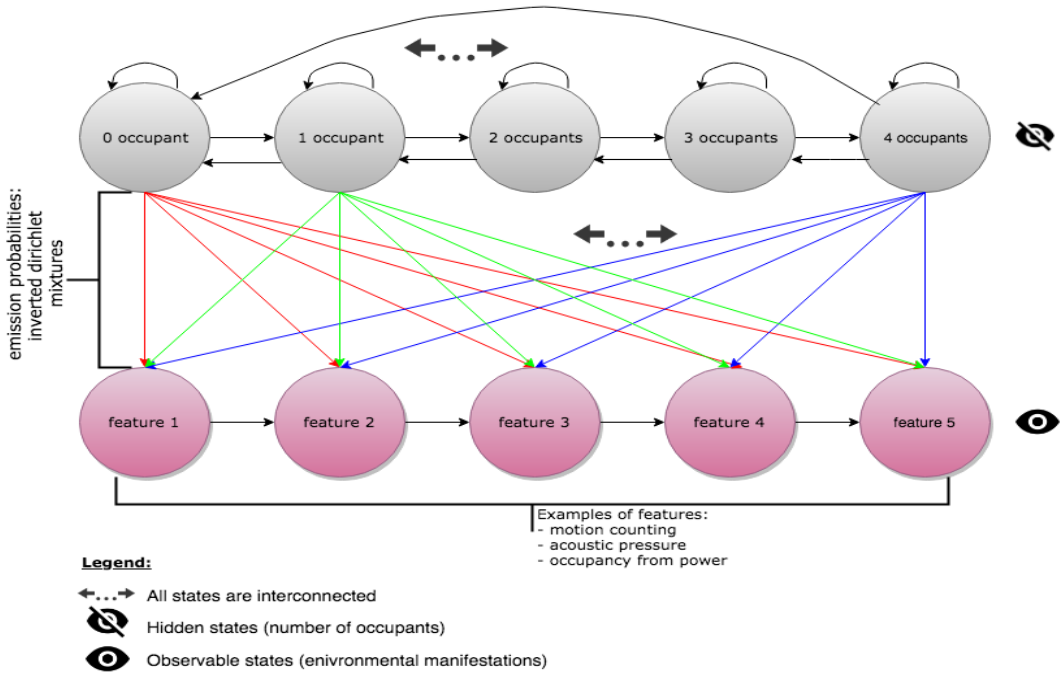


Figure 12: $\mathcal{IDHMM}$ structure according to the case of study

As previously mentioned, the number of occupants has been determined beforehand using a traditional counting of attendants of the room, by means of two cameras intentionally placed in such a way that we can spot each attendant. According to the recorded data, we have the possibility to spot a maximum of 4 occupants $\{0, 1, 2, 3, 4\}$. This being the case, we dispose of ground truth related to the studied data and we used it as an evaluation for our model.

The occupancy estimation is subject to statistical reasoning implemented via our $\mathcal{IDHMM}$ framework. The occupancy is considered to be a hidden state (in this case $K = 5$), and the most important features chosen among the set of available features are observations, see figure 12. Exploiting HMMs will allow a special focus on temporal correlations between occupancy levels and environmental features. This will add to the temporal knowledge and thus improve the prediction potential.

With regard to the choice of the considered features, we noticed that not all measurements are highly effective and that some of the measurements drive the learning process to be slow or reduce its accuracy. In fact, levels of CO_2 do not rise immediately as a visitor comes in as a result of the ventilation present in the area. This preliminary thought led to two different ascertainments. The first one is that a quantitative measurement of the usefulness of features is required and has been conducted in [11] leading to a set of interesting features to work on: {motion counting, acoustic pressure, occupancy from power}. The second is the need to re-evaluate the nature of the used distribution to represent emission probabilities, mainly because of the instability of some of the features, namely levels of CO_2. This experiment has been previously conducted by Amayri et al. calculating the information gain, which depends on the concept of entropy explained in detail in [11]. Whilst the cited work used a conventional HMM, we are working to solve the same problem and improve the obtained results by adopting our $\mathcal{IDHMM}$ whose emission probability distribution describes better the fluctuating nature of features taken into consideration and

handles perfectly the positive nature of our feature vectors. The set of interesting features to take into consideration in this work is {motion counting, acoustic pressure, occupancy from power}

2.5.4.2 Results and discussion

We dispose of a dataset containing observations collected in a time frame of 17 days where measurements are noted to the record every *1quantum* (30 minutes). We choose to carry out our investigation in such a way that we use the data collected on days from May 4^{th}, 2015 to May 13^{th}, 2015 included, to train the model and the rest of the data for testing. For the sake of evaluating, a GHMM is also trained with the same raw data. Before running the $\mathcal{IDHMM}$ model, data have been converted into normalized feature vectors for the sake of simplifying our data, figure 13 shows the results of $\mathcal{IDHMM}$ where real occupancy is compared to the estimated occupancy. The idea is that for our estimation we only used knowledge but in terms of practical exploitation of the model, no cameras will be allowed. The achieved results are very close to the ground truth values generated thanks to cameras only used here in the best interest of model validation.

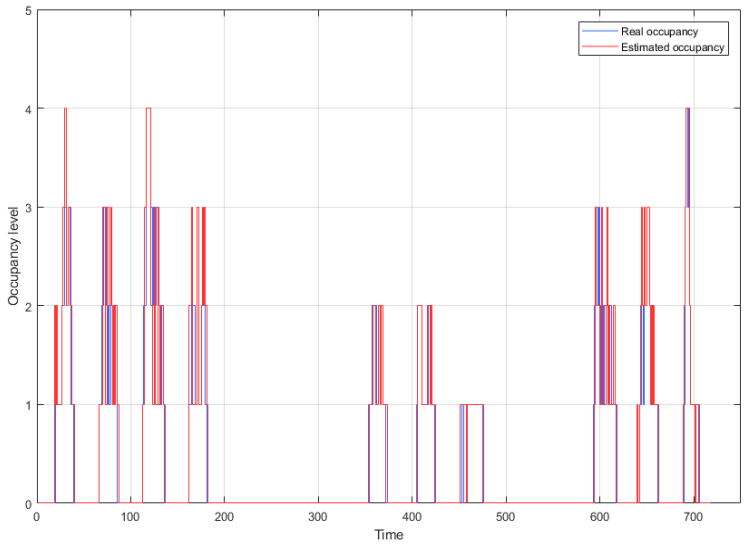


Figure 13: Occupancy estimation using $\mathcal{IDHMM}$

The occupancy estimation accuracy gives more information on the model efficacity when combined with an average error rate. The latter provides us with a better perspective as to changes in the estimated values.

Comparison results with reference to previously realized work by [11] and compared to GHMM are presented in table 5. An average accuracy of 90.8% is achieved by the $\mathcal{IDHMM}$ model with an error rate of 0.11 persons compared respectively to 83.27% and 0.29 persons for GHMM against 89% and 0.1 achieved by [11] but also instead of DHMM with 88.16% and 0.13. These results suggest that $\mathcal{IDHMM}$ handles occupancy estimation better than both methods.

Table 5: Occupancy estimation comparison between $\mathcal{IDHMM}$ and GHMM

Model	Accuracy	Error rate
$\mathcal{IDHMM}$	**90.8%**	**0.11**
GHMM	83.27%	0.29
DHMM	88.16%	0.13
Amayri et al. [11]	89%	0.1

There is no doubt that occupancy estimation without any human supervision or manual labeling is a challenging task. An inverted Dirichlet hidden Markov model has been used for the first time to estimate human presence in a real office context. Results showed that the estimation accuracy is higher than a Gaussian-based HMM with a very small error rate. Throughout all the conducted experiments we were fairly intrigued and there were some raised questions with regard to the possibility of extending the model to predict occupancy in an open space office, or even in a house or apartment context. Moreover, this task could also be refined if we focus on the choice features to further enhance the model efficacity. Therefore, a feature selection refinement could be taken into consideration for future work.

2.6 Conclusion

In this paper, we proposed a new extension of the broadly used HMMs, by adapting them to handle positive vectors and to prove their capacity to outperform the traditionally GM-based HMMs in various contexts of real-life applications without an obligation to perform major modifications on their underlying conventional structure. The particular interest in adopting inverted Dirichlet as emission probability distributions is encouraged by their excellent capacities to easily approximate and model many shapes of data. Contrary to the Gaussian distribution, it can be symmetric, asymmetric and skewed to either left or right

and give a much wider margin to model data. With reference to HMMs, ID grants two main tasks: estimation of emission probabilities and determining the appropriate number of clusters, that is the number of the hidden states. We made sure that the derivations provided for these models could easily be used to obtain HMMs extensions with other exponential probability distributions. Thus, a wider range of data types would be reached by adapting the appropriate distribution to match each data's nature. Moreover, we have brought to light the capacity of the ID distribution to model non-Gaussian data. The other part of this work is the application of this new extension to the purpose of dealing with highly trending real-life applications. Our method is indeed very efficient in categorizing images of texture, dynamic texture and recognizing facial expressions with very high accuracy within a real-time period without the need for high computational resources. Besides, we proved that our method is also very effective in estimating occupancy in an office setting implemented in the context of smart building development. Nonetheless, there are many future work considerations that have been raised during the experimentations. In fact, the extension of this model to the generalized ID form could help us deal with the strictly positive covariance structure imposed by the inverted Dirichlet form. We also presume that to further enhance the accuracy of the model training in the recognition applications, we could integrate a feature selection approach qualified to discard irrelevant but compromising features and thus improve the classification task.

Chapter 3

A novel Feature Selection method using Generalized Inverted Dirichlet-based HMMs for image categorization

Hidden Markov Models (HMMs) have consistently been a powerful tool for performing numerous challenging machine learning tasks such as automatic recognition. The latter perceives all objects of the universe through information carried by their characteristics or features. However, not all available data is always valuable for distinguishing between the different objects, scenes, and scenarios; referring analogically to states. More often than not, automatic recognition is accompanied by a feature selection to reduce the number of collected features to a relevant subset. Although sparse, the majority of literature resources available on feature selection for HMMs, presuppose either a single Gaussian or employ a Gaussian mixture model (GMM) as emission distribution. The proposed method builds upon the feature saliency model introduced by Adams, Cogill, and Beling [3], and is adjusted to handle complex multidimensional data by using as a novel experiment, GID (Generalized Inverted Dirichlet) mixture models) as emission probabilities. We make use

of an Expectation-Maximization (EM) algorithm [63] to compute maximum a posteriori (MAP) [108] estimates for model parameters. The complete inference and parameter estimation of our GID-FSHMM (GID Feature Selection-based HMM) are detailed in this work. Automatic recognition applications such as facial expression recognition and scenes categorization demonstrate comparable to higher performance compared to the extensively used Gaussian mixture-based HMM (GHMM), the Dirichlet-based (DHMM) and the inverted Dirichlet-based HMM (IDHMM) without feature selection and also when the latter is embedded in all of the aforementioned models.

3.1 Introduction

The successful application of HMMs to a great number of areas ranging from speech recognition to image categorization broke new grounds by bringing many extensions and novelties not only in terms of the methods used along with HMMs to better their performance but also in the volume and diversity of data collected for analysis using these methods. There is no doubt that this expansion of data, types of information, and features contributed enormously to refining and improving machine learning tasks and methods. Nevertheless, it has triggered a considerable amount of problems and challenges, namely the formidable curse of dimensionality often resulting from the manipulation of high-dimensional data. For example, in clustering tasks, it is a widely held view that the more information, data, and features we manipulate, the better an algorithm is expected to perform [145]. However, this is not the case in practice. Many features can be just "noise" and may cause the finest pattern recognition and machine learning techniques to struggle as a result of irrelevancy and thus degrade the modelling performance [72]. Thereby, feature selection is used to increase modelling performance since it allows eliminating noise in the data, speeding up the models' training and prediction, decreasing overfitting odds and most importantly reducing the computational cost after disregarding many features. We intend by feature

selection, the process of decreasing the number of gathered features to a relevant subset of features and is usually used to counter the curse of dimensionality [2]. Aside from feature extraction, which is a separate problem, feature selection determines relevant features from a given set of features, whereas feature extraction generates new features from a given set. Unlike feature extraction, feature selection does not come up with new features nor does it amend the primary features.

The primary inducement for adopting feature selection strategies is their important potential to improve modelling and generalization capabilities if performed reliably and properly [6]. Applying feature selection permits taking into consideration the significant contribution of feature screening to the classification process. In fact, each different feature contributes differently to the classification structure based on its degree of relevance [224, 83]. The latter is intended to be determined to improve our models' performance, in particular using simultaneous feature selection and classification in the course of an unsupervised process which is considered to be one of the most challenging problems in data mining and machine learning. In practice, the said case implies selecting features without a priori knowledge about data labels.

In most applications of HMM, features are pre-selected based on domain knowledge, and the feature selection procedure is completely omitted. Usually, to train HMMs, even in the case where feature selection is considered, features are selected traditionally. That is, features are selected in advance either based on already available data or relying on experts' knowledge. These practices are the result of the scarcity of literature in terms of unsupervised feature selection methods specific for HMMs [2] [103], not to mention the high computational cost of wrapping methods. Despite the extensive research and investigations that are made on feature selection in their general case, methods specific for HMMs are lacking. Feature selection methods for HMMs and mixture models are seldom treated as a joint topic. Most importantly, the use of generalized inverted Dirichlet mixtures to

model the emission probabilities within the HMM framework together with feature selection as an embedded process is unprecedented. In this work, we propose a fully customized feature selection methodology with a complete empirical and experimental study of Feature Saliency embedding into the GID-based HMM.

Feature selection plays a major role in speeding the learning process and refining the models' interpretation. It can drastically minimize the risk of overfitting and mitigates the effects of the curse of dimensionality [122]. Above-mentioned, the feature selection process is embedded in the training of the HMM, which represents the main takeaway from this research work.

Our vision of integrating feature selection in the HMM framework holds beyond the simple procedure of solely combining state of the art feature selection methods such as in the case of [172], where several ranking methods like Bhattacharyya distance [139], entropy and Wilcoxon [109], have been used to reduce the number of features fed to the HMM. As far as we are concerned, we are resolute to use the feature saliency as in [1], thoroughly embedded in the HMM framework making only one core method ready for use directly to treat any set of features.

The work presented in this manuscript can also be viewed intellectually at two different levels. First, it allows the integration of the non-conventional feature selection techniques into the framework of HMM, second, it allows the use of GID mixtures as a premiere to model data fed to HMM specifically emission distributions.

The remainder of this paper is organized as follows: In section 2 we summarize the previous works adopting HMMs. Then, we outline some of the applications using general feature selection methods along with HMMs as a predictive model. In section 3, we present our GID-FSHMM and explain all the corresponding integration steps. The subsequent section 4 showcases real-life problems experimentation and analyses the obtained results. Finally, the paper closes with a summary of the work and concluding remarks.

3.2 Related work

3.2.1 Hidden Markov Models

In this section, we recall a handful of background information on HMMs, while focusing on previous related work using HMMs conjointly with feature selection. Hidden Markov models are a ubiquitous tool commonly utilized to model time series data [110][116] with applications across numerous areas. Used for decades in speech recognition [204], text classification [140, 64], face recognition [178] and fMRI data analysis [93], HMMs represent a powerful statistical tool that have proven to be not only useful but also efficient in various machine learning-based applications.

An HMM consists mainly of two distinct sequences of states. The first is a sequence of hidden states modelled by a Markov chain [20], and the second is a sequence of observed events or features related to the hidden states. The typical purpose behind using HMMs is to represent probability distributions over sequences of observations, with the assumption that the observations are discrete. Therefore, the hidden states sequence can be estimated from the sequence of correlated observations. It is possible to specify an HMM by an initial probability, a matrix of transition probabilities between the states, and a set of parameters of the emission probability distribution which will be more focused on later in this paper. Most importantly, an HMM is outlined by two fundamental properties. Firstly, it assumes that an observation at time t is generated by some process whose state h_t is hidden from the observer. Second, it implied that the state of the said hidden process fulfills the Markov property [104]; that is, given the value of h_{t-1}, the current state h_t is independent of all the states before the time $t-1$. Thus, the observed features are modelled meeting the property of conditional independence given the state sequence. At the application level, the learning

of parameters is simply finding the best set of state transitions and emission probabilities amid the states of the model. Consequently, an output sequence or a set of sequences is specified. At each time a state sequence is handled, there is a corresponding vector of observations composed of features collected from various sources. However, not all features are likely to be useful to the model. That is why, to build a rigorous model, we ought to remove all features that do not contribute to its usefulness, without degrading its accuracy.

3.2.2 Feature selection and its application with HMMs

Feature selection is a wide research area and many methods to reduce a given set of features have been implemented in both supervised and unsupervised contexts [144].

Typically, feature selection techniques are present in the state of the art under three main categories, namely, filters, wrappers, metaheuristic methods and embedded [2]. While filter methods such as information gain [235, 149], Pearson's correlation coefficient [113] and variance threshold [240], treat the evaluation of all features and return a relevant subset out of them apart from the model building process, wrapper methods tend to optimize the classifier's performance for the most part. Wrappers, which commonly adopt either forward selection [142], backward elimination [77] or recursive feature elimination [167, 84, 259], identify the relevant features depending on the learning algorithm. That is, when using wrappers, the model itself is built depending on a certain subset of features and its performance is measured upon particular criteria. Methods relying on metaheuristic algorithms tackle feature selection as an optimization problem. Composed but not limited to evolution-based algorithms such as Genetic Algorithm [126], these methods obtain the optimal solution thanks to their simplicity, flexibility and their capability to avoid local optima [183]. They start their feature selection process by generating random solutions that do not require heavy derivatives calculations and carry on an exploration phase to thoroughly investigate search space and identify promising areas. Embedded methods namely L_1 regularization

and decision trees [115, 33] aspire to simultaneously select the features and build the model. Although filter methods exhibit a significant low time complexity, they are usually criticized for ignoring certain informative features [206]. On the other hand, metaheuristic and wrapper-based methods evaluate the usefulness of selected features using learner's performance and can thus be more complex but still not very time-consuming. However, it has been proved that other optimization algorithms, namely embedded methods, can be more efficient given the fact that they not only improve the performance of the model but also facilitate results analysis. Indeed, there is a significant complexity compromise when it comes to using embedded methods, but these methods succeeded in adapting to several types of data and can be used with the majority of machine learning models. Embedded methods are also very useful when investigating relationships between features, which is an arising challenge nowadays.

In particular, feature selection for HMMs is driven by a crucial need to determine which feature to use in the model. Despite being investigated in numerous general and mixture models-based studies [91][145], feature selection methods dedicated to HMMs are particularly limited. In fact, in the majority of applications, features are selected beforehand based on domain knowledge, and a consonant feature selection procedure is fully lacking [174][246]. Clearly, transformation methods such as Principal Component Analysis (PCA) and Independent Component Analysis (ICA) do reduce the number of features in the model and for this same reason, they have been integrated into HMMs in [17, 261]. However, the mentioned methods do not really act as feature selection techniques as they are not able to eliminate data streams, and hence they merely are considered feature extraction techniques.

Embedded or integrated feature selection approaches, which are the main focus in this manuscript, ought to consider the whole set of features at once. These features serve as an input to the maximizing learning algorithm that is deployed to optimize the models' performance. As an output, the reduced set of features, as well as the models' parameters,

are generated. Hence, an embedded feature selection method is identified as a simultaneous selection of features and model construction. This combines both the wrappers and filters advantages of respectively selecting feature subsets concerning a specific learning algorithm and the computational efficiency [1]. As previously indicated, one of the embedded methods for feature selection is the classification and regression trees [176]. The latter applies a recursive splitting of the feature space to generate a classification model. Features identified as the ones being involved in improving the model will exclusively be included in the learning algorithm. In contrast to the mixture models, context [90, 145, 42], literature about feature selection integration into the HMM framework is somewhat narrow. Nearly all HMM-based adaptations of feature selection were based on what is also known as the concept of feature saliency, which has been defined by [145] as a metric associated with a given feature, that is the probability that the said feature is relevant. Zhu et al. [263] are among the first to use a jointly embedded estimation and feature selection method, where they apply a variational Bayesian framework to the end of salient features inference. They use the implemented method to simultaneously infer the number of hidden states as well as the models' parameters. The adopted approach showed interesting results, however, the use of the variational Bayesian sometimes manifested a significant underestimation of the variance for the approximate distribution. Adams et al. [1] put forward a feature saliency model using hidden Markov models. The main idea is to use feature saliency variables to represent the probability that a given feature is relevant, by drawing a distinction between state-dependent and state-independent distributions. The said model operates in the case where the number of hidden states is known. For the matter, it provides a maximum a posteriori based estimation that selects the most relevant features using an Expectation-Maximization (EM) algorithm. This approach takes advantage of the already specified number of states to provide maximum a posteriori estimates and save the most relevant features by applying an Expectation-Maximization algorithm [63]. Moreover, Zheng et al.

[260] adopted a strategy that combines a hidden Markov model, a localized feature saliency measure, and two t-Student distributions for the purpose of distinguishing between relevant and non-relevant features. This strategy made it possible to accurately model emission parameters for each hidden state. Similarly to [263], the parameter estimation was operated using a variational Bayes framework. More recently, Fons et al. [103] incorporated Adams' feature saliency HMM (FSHMM) [3] into a dynamic asset allocation system. The authors applied their HMM-based feature selection method to train their systematic trading system by testing its performance on real-life data. It showed that even without a financial expert's involvement, the results reached a decent accuracy allowing the model to objectively contribute to portfolio construction and to prevent biases in the feature selection process. From their side, authors in [47] proposed a feature selection algorithm embedded in an HMM applied to gene expression time-course data, and they succeeded in reducing the feature domain by up to 90% leaving only a few but relevant features. The notable drawback of the mentioned work is that features deemed as irrelevant are eliminated and hence can drastically affect the models' accuracy in the case the aforementioned features seem to be relevant after treatment.

There is indubitably a significant challenge when analyzing dense data, that is dealing with the saliency parameters besides those imperative for the model itself. As a consequence, the parameter estimation can sometimes be a sensitive task, not to mention the huge impact that the number of needed hidden states has on the said estimation. For this particular reason, we need to adapt the model in a way that it can handle the modelling of the data using a lower number of parameters to come up with the most relevant features from the candidate sets.

3.3 The proposed GID-FSHMM model

3.3.1 Feature selection integration in Hidden Markov Model

In this section, we start by presenting the Hidden Markov Model and we recall the feature saliency concept that we will embed in the HMM. Then, we

3.3.1.1 The Hidden Markov Model

We consider a HMM with continuous emissions and K states. We put $y = \{y_0, y_1, ..., y_T\}$ the sequence of observed data with $y_t \in \mathbb{R}^L$, where T designates the time factor and L is the number of features. The observation for the l-th feature at time t, which is represented by the the l-th component of y_t, is denoted by y_{lt}.

Let $x = \{x_0, x_1, ..., x_T\}$ be the sequence of hidden data. The transition matrix of the Markov chain associated to this sequence is denoted as $B = \{b_{ij} = P(x_t = j|x_{t-1} = i)\}$ and π is the initial state probability. Thus the complete data likelihood can be expressed as:

$$p(x, y|\Lambda) = \pi_{x_0} c_{x_0}(y_0) \prod_{t=1}^{T} b_{x_{t-1}, x_t} c_{x_t}(y_t) \tag{31}$$

where Λ is the set of model parameters, and $c_{x_t}(y_t)$ is the emission probability given state x_t. In our feature selection hidden Markov model (FSHMM) we apply a feature saliency approach over the emission probability distribution in order to select the relevant features and to estimate our parameters[145]

The graphical model of the GID Hidden Markov Model can be seen in figure 14.

3.3.1.2 Feature saliency-based Hidden Markov Model

The feature-saliency based HMM measures the relevancy of a certain feature as follows; if the latter's distribution is dependent on the underlying state, the feature is believed to be relevant. In the case where its distribution is independent of the state, the feature is

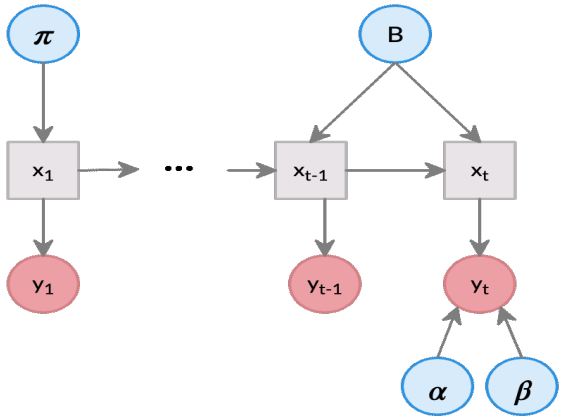


Figure 14: The Hidden Markov Model: Grey squares represent latent variable, pink circles are observations, and blue circles represent model parameters where α and β are GID parameters.

considered irrelevant [1].

Thus, we put a set of binary variables $z = \{z_1, ..., z_L\}$ indicating the relevancy of features, that is $z_1 = 1$ if the l-th feature is relevant and $z_1 = 0$ if it's irrelevant. The feature saliency ρ_l is the probability that the l-th feature is relevant.

In this work we assume that all features are conditionally independent given the state. Hence, the conditional distribution of y_t given z and x can be written as follows:

$$p(y|z, x_t = i, \Lambda) = \prod_{l=1}^{L} r(y_{lt}|\theta_{il})^{z_l} q(y_{lt}|\lambda_l)^{1-z_l} \tag{32}$$

where $r(y_{lt}|\theta_{il})$ is the conditional feature distribution for the l-th feature with state-dependent parameters θ_{il} which later will be detailed with depending on the adopted type of mixture, and $q(y_{lt}|\lambda_l)$ is the state independent feature distribution with parameters λ_l. $\Lambda = \{\theta, \rho, \lambda\}$ is the set of all our FSHMM model parameters. The marginal probability of z is:

$$p(z|\Lambda) = \prod_{l=1}^{L} \rho_l^{z_l} (1 - \rho_l)^{1-z_l} \tag{33}$$

The joint distribution of y_t and z given x can be expressed as:

$$p(y_t, z|x_t = i, \Lambda) = \prod_{l=1}^{L} [\rho_l r(y_{lt}|\theta_{il})]^{z_l} [(1-\rho_l) q(y_{lt}|\lambda_l)]^{1-z_l} \tag{34}$$

The marginal distribution for y_t given x over all values of z is:

$$\begin{aligned} c_{x_t}(y_t) &= p(y_t|x_t = i, \Lambda) \\ &= \prod_{l=1}^{L} \left(\rho_l r(y_{lt}|\theta_{il}) + (1-\rho_l) q(y_{lt}|\lambda_l)\right) \end{aligned} \tag{35}$$

The complete data likelihood of the FSHMM can thus be written as:

$$p(x, y, z|\Lambda) = \pi_{x_0} p(y_0, z|x_0, \Lambda) \prod_{t=1}^{T} b_{x_{t-1}, x_t} p(y_t, z|x_t, \Lambda) \tag{36}$$

The form of $q(.|.)$ indicates our prior knowledge about the distribution of the non-salient features. We put $q(.|.)$ and $r(.|.)$ to follow an inverted generalized Dirichlet distribution, as this can lead to better results for the reasons explained earlier in this paper.

In this work, the state-dependent and the state-independent distributions are assumed to be GID mixtures. Accordingly, the set of model parameters for the GID-FSHMM is $\Lambda = \{\pi, B, \alpha, \beta, \rho, \lambda\}$. Figure 15 shows the feature saliency GID-based HMM.

3.3.2 GID mixtures and integration into the FSHMM framework

3.3.2.1 Generalized Inverted Dirichlet

The choice of GID is backed by the several interesting mathematical properties that this distribution has. These properties allow for a representation of GID samples in a transformed space where features are independent and follow inverted Beta distributions. Adopting this

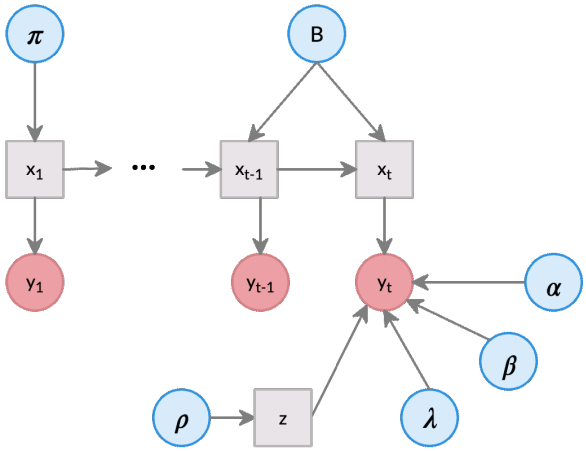


Figure 15: The feature saliency GID-based Hidden Markov Model: Grey squares represent latent variable, pink circles are observations, and blue circles represent model parameters.

distribution lets us take advantage of conditional independence among features. This interesting strength is used in this paper to develop a statistical model that handles not only positive data but also feature selection.

Let $\overrightarrow{X}$ a D-dimensional positive vector following a GID distribution. The joint density function is given by Lingappaiah [158] as:

$$p(\overrightarrow{X}|\overrightarrow{\alpha},\overrightarrow{\beta}) = \prod_{d=1}^{D} \frac{\Gamma(\alpha_d+\beta_d)}{\Gamma(\alpha_d)\Gamma(\beta_d)} \frac{X_d^{\alpha_d-1}}{\left(1+\sum_{l=1}^{d} X_d\right)^{\eta_d}} \tag{37}$$

where $\overrightarrow{\alpha} = [\alpha_1, ..., \alpha_D]$, $\overrightarrow{\beta} = [\beta_1, ..., \beta_D]$. η is defined such that $\eta_d = \alpha_d + \beta_d - \beta_{d+1}$ for $d = 0, ..., D$ with $\beta_{D+1} = 0$.

The GID estimation is made simple thanks to an essential propriety, that is if there exists a vector $\overrightarrow{X}$ that follows a GID distribution, then we can come up with another vector $\overrightarrow{W}_n = [\overrightarrow{W}_{n1}, ..., \overrightarrow{W}_{nD}]$ where each element follows an inverted Beta (IB) distribution following the

transformation:

$$W_{nd} = f(X_{nd}) = \begin{cases} X_{nd}, & \text{d=1} \\ \frac{X_{nd}}{1+X_{n1}+,...,+X_{nd-1}}, & \text{d=2, ..., D} \end{cases} \tag{38}$$

Then, the multivariate extension of the 2-parameters inverted Beta distribution is given by:

$$p_{IBeta}(W_{nd}|\alpha_{jd}, \beta_{jd}) = \frac{\Gamma(\alpha_{jd} + \beta_{jd})}{\Gamma(\alpha_{jd})\Gamma(\beta_{jd})} \frac{W_{nd}^{\alpha_{jd}-1}}{(1 + W_{jd})^{(\alpha_{jd}+\beta_{jd})}} \tag{39}$$

The mean of IB is given by:

$$E(W_d) = \frac{\alpha_d}{\beta_d - 1} \tag{40}$$

The variance of IB is given by:

$$Var(W_d) = \frac{\alpha_d(\alpha_d + \beta_d - 1)}{(\beta_d - 2)(\beta_d - 1)^2} \tag{41}$$

3.3.2.2 GID mixture model

Let us consider a data set $\mathcal{X}$ of N D-dimensional positive vectors, $\mathcal{X} = (\overrightarrow{X_1}, \overrightarrow{X_2}, ..., \overrightarrow{X_N})$. We assume that $\mathcal{X}$ is governed by a weighted sum of M GID component densities with parameters $\Theta = (\overrightarrow{\theta_1}, \overrightarrow{\theta_1}, ..., \overrightarrow{\theta_M}, p_1, p_2, ..., p_M)$ with $\overrightarrow{\theta_j}$ is the vector of parameters of the j-th component and p_j are the mixing weights which are positive and sum to one [6]:

$$p(\overrightarrow{X_i}|\Theta) = \sum_{j=1}^{M} p_j p(\overrightarrow{X_i}|\overrightarrow{\Theta_j}) \tag{42}$$

where $p(\overrightarrow{X_i}|\overrightarrow{\Theta_j})$ is the GID distribution with $\Theta_j = (\alpha_{j1}, \beta_{j1}, \alpha_{j2}, \beta_{j2}, ..., \alpha_{jD}, \beta_{jD})$ is

the set of parameters defining the j-th component. Furthermore, in mixture-based clustering, each data point $\overrightarrow{X_i}$ can be assigned to all classes with different posterior probabilities $p(j|\overrightarrow{X_i})$. Therefore, a factorization of the posterior probability can simply be expressed as:

$$p(j|\overrightarrow{Y_i}) \propto p_j \prod_{l=1}^{D} p_{IBeta}(X_{il}|\theta_{jl}) \tag{43}$$

where $X_{i1} = Y_{i1}$ et $X_{il} = \frac{Y_{il}}{1+\sum_{l=1}^{D} Y_{il}}$ for $l > 1$, $p_{IBeta}(X_{il}|\theta_{jl})$ is an inverted Beta distribution with $\theta_{jl} = (\alpha_{jl}, \beta_{jl})$, $l = 1, ..., D$

In this fashion, the clustering structure underlying $\mathcal{X}$ is the same as that underlying $\mathcal{Y} = (\overrightarrow{Y_1}, ..., \overrightarrow{Y_N})$, and it can be described by the following mixture model with conditionally independent features:

$$p(\overrightarrow{X}_i|\Theta) = \sum_{j=1}^{M} p_j \prod_{l=1}^{D} p_{IBeta}(X_{il}|\theta_{jl}) \tag{44}$$

3.3.2.3 GID mixture-based FSHMM

As a first attempt in the context of feature saliency-driven HMMs, to the extent of our knowledge, we are using a mixture of GID as emission probabilities of our FSHMM. Gaussian mixtures, in particular, have seldom been tested previously and applied successfully [145] [263]. Assuming the relevant feature distribution is represented by a mixture of M GID distributions, we let $\Phi = \{\phi_{1t}, ..., \phi_{Mt}\}$ be the set of variables indicating the mixture component, where $\phi_m = 1$ if observation t comes from the m^th mixture and $\phi_{mt} = 0$ otherwise. To indicate the probability that the observation comes from the m^th mixture, given the state, we put ω_{im}. In this regard, the set of model parameters Λ becomes $\{\pi, B, \alpha, \beta, \rho, \lambda, \omega\}$. The idea behind the GID-based FSHMM is to suppose that a given feature y_{lt} is generated from a mixture of two univariate distributions. The first one is supposed to generate relevant features and is distinct for each cluster. The second is common to all clusters in a way that it is independent of class labels, and generates irrelevant features.

This purpose can be formulated as follows.

The marginal probability of ϕ_t can be expressed as

$$p(\Phi|\Lambda) = \prod_{m=1}^{M} \omega_{im}^{\phi_{mt}} \tag{45}$$

In the same manner as in (33), we assume the features are conditionally independent given the state. Thus, the conditional distribution of y_t given x, y and Φ can be formulated as

$$p(y_t|\Phi, z, x_t = i, \Lambda) = \prod_{l=1}^{L} \left[r(y_{lt}|\alpha_{ilm}, \beta_{ilm})^{z_l} q(y_{lt}|\alpha_{\lambda,ilm}, \beta_{\lambda,ilm})^{1-z_l} \right]^{\phi_{mt}} \tag{46}$$

The joint distribution of y_t, Φ, and z given x is

$$\begin{aligned} p(y_t, \Phi, z|x_t = i, \Lambda) &= p(y_t|\Phi, z, x_t = i, \Lambda) p(\Phi|\Lambda) p(z|\Lambda) \\ &= \prod_{m=1}^{M} \left[\omega_{im} \prod_{l=1}^{L} \left[\rho_l r(y_{lt}|\alpha_{ilm}, \beta_{ilm})^{z_l} \right] \left[(1-\rho_l) q(y_{lt}|\alpha_{\lambda,ilm}, \beta_{\lambda,ilm})^{1-z_l} \right]^{\phi_{mt}} \right] \end{aligned} \tag{47}$$

The marginal distribution for y_t given x is obtained by summing (47) over z and Φ such as

$$\begin{aligned} c_{x_t}(y_t) &= p(y_t|x_t = i, \Lambda) \\ &= \sum_{m=1}^{M} \omega_{im} \prod_{l=1}^{L} (\rho_l r(y_{lt}|\alpha_{ilm}, \beta_{ilm}) + (1-\rho_l) q(y_{lt}|\alpha_{\lambda,ilm}, \beta_{\lambda,ilm}) \end{aligned} \tag{48}$$

The complete data likelihood for the FSHMM with GID emissions is

$$p(x, y, z, \Phi|\Lambda) = \pi_{x_1} p_{IBeta}(y_1, \Phi, z|x_1, \Lambda) \prod_{t=2}^{T} b_{x_{t-1}, x_t} p_{IBeta}(y_t, \Phi, z|x_t, \Lambda) \tag{49}$$

3.3.3 Parameter estimation of the GID-FSHMM

3.3.3.1 Update equations for FSHMM parameters

In order to perform the estimation of parameters, we opt for using the EM algorithm, referred to as Baum-Welch when applied in the context of HMMs [30, 204]. We use this algorithm to calculate the maximum-likelihood (ML) estimates for the model parameters. For the part where we evaluate the features, we are bound to place priors on the parameters to compute the maximum a posteriori (MAP) estimates [108]. We need to go over the two steps of the Baum-Welch algorithm. First, in the E-step we need to find the expected value of the complete log-likelihood taking into consideration the data and the underlying model parameters. Second, in the M-step we proceed to maximize the expectation computed in the previous step in order to figure the next set of model parameters out. The Baum-Welch is iterated until an experimentally determined stopping threshold is met. The $\mathcal{Q}$ function designates the expectation of the complete log-likelihood and is given by:

$$\begin{aligned} \mathcal{Q}(\Lambda, \Lambda') &= \mathbb{E}\big[logp(x, y, z, \Phi|\Lambda)|y, \Lambda'\big] \\ &= \sum_{x,z,\Phi} log(p(x, y, z, \Phi|\Lambda)|\Lambda')p(x, z, \Phi|y, \Lambda') \end{aligned} \tag{50}$$

where Λ and Λ' represent the set of model parameters for the current iteration and the set of parameters from the previous iteration respectively. We place priors on the parameters and calculate the MAP estimates with an eye toward the automatic feature assessment and selection. Hence the $\mathcal{Q}$ is changed by adding $G(\Lambda)$ the prior on the model parameters such as:

$$\mathcal{Q}(\Lambda, \Lambda') + logG(\Lambda) \tag{51}$$

By analogy to the previously explained EM procedure, the complete log-likelihood $\mathcal{Q}$ is calculated in the E-step (50), then the $logG(\Lambda)$ is added up and equation (51) is maximized in the M-step. For this matter, several probabilities are needed for the FSHMM, the E-step takes in charge the computation of the following quantities:

$$\zeta_t(i) = \mathbb{P}(x_t = i|y, \Lambda), \tag{52}$$

$$\xi_t(i, j) = \mathbb{P}(x_t = i, x_{t+1} = j|y, \Lambda) \tag{53}$$

where $\zeta_t(i)$ et $\xi_t(i, j)$ are respectively the conditional state probabilities and the conditional transition probabilities. These quantities are calculated using the forward-backward algorithm. As a result, the following quantities are what the E-step probabilities turn out to be after iterating the forward-backward algorithm

$$\begin{aligned} \delta_{ilmt} &= p(y_{lt}, z_l = 1|\phi_{mt} = 1, x_t = i, \Lambda') \\ &= \rho_l p(y_{lt}|\alpha_{ilm}, \beta_{ilm}), \end{aligned} \tag{54}$$

$$\begin{aligned} \epsilon_{ilmt} &= p(y_{lt}, z_l = 0|\phi_{mt} = 1, x_t = i, \Lambda') \\ &= (1 - \rho_l) q(y_{lt}|\alpha_{\lambda,ilm}, \beta_{\lambda,ilm}), \end{aligned} \tag{55}$$

$$\begin{aligned} \tau_{ilmt} &= p(y_{lt}|\phi_{mt} = 1, x_t = i, \Lambda') \\ &= \delta_{ilmt} + \epsilon_{ilmt}, \end{aligned} \tag{56}$$

$$
\begin{aligned}
u_{ilmt} &= p(z_l = 1, x_t = i, \phi_{mt} = 1 | y, \Lambda') \\
&= \zeta_t(i) \left(\frac{\delta_{ilmt}}{\tau_{ilmt}} \right) \left(\frac{\omega_{im} \prod_{l=1}^{L} \tau_{ilmt}}{\sum_{m}^{M} \omega_{im} \prod_{l=1}^{L} \tau_{ilmt}} \right),
\end{aligned} \tag{57}
$$

and

$$
\begin{aligned}
v_{ilmt} &= \mathbb{P}(z_l = 0, x_t = i, \phi_{mt} = 1 | y, \Lambda') \\
&= \zeta_t(i) \left(\frac{\epsilon_{ilmt}}{\tau_{ilmt}} \right) \left(\frac{\omega_{im} \prod_{l=1}^{L} \tau_{ilmt}}{\sum_{m}^{M} \omega_{im} \prod_{l=1}^{L} \tau_{ilmt}} \right),
\end{aligned} \tag{58}
$$

The $\mathcal{Q}$ function is expanded into a sum of terms where each term can be maximized independently. These terms are the $\mathcal{Q}$ function applied to the initial state π, the state-transition b, and the parameters for the emission distribution $\theta = \{\alpha, \beta, \lambda, \rho\}$. Consequently, for all parameters, except for the GID distribution ones, the maximization step gives, as a result, the following parameters and their updates

$$
\hat{\pi_i} = \zeta_t(i), \tag{59}
$$

$$
\hat{b_{ij}} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \zeta_t(i)}, \tag{60}
$$

$$
\hat{\omega_{im}} = \frac{\sum_{t=1}^{T-1} \sum_{l=1}^{L} u_{ilmt}}{\sum_{t=1}^{T-1} \sum_{l=1}^{L} \sum_{m=1}^{M} u_{ilmt}} \tag{61}
$$

$$\begin{aligned}\hat{\rho}_l &= \frac{\sum_{t=1}^{T-1}\sum_{l=1}^{L}\sum_{m=1}^{M} u_{ilmt}}{\sum_{t=1}^{T-1}\sum_{l=1}^{L}\sum_{m=1}^{M} u_{ilmt} + \sum_{t=1}^{T-1}\sum_{l=1}^{L}\sum_{m=1}^{M} v_{ilmt}} \\ &= \frac{\sum_{t=1}^{T-1}\sum_{l=1}^{L}\sum_{m=1}^{M} u_{ilmt}}{T}\end{aligned} \tag{62}$$

3.3.3.2 Estimation of GID parameters

Here, the estimation of the GID parameters is equivalent to maximization of the following log-likelihood function

$$\begin{aligned}log(p(\overrightarrow{X}_i|\Theta) &= \sum_{i=1}^{N}\sum_{j=1}^{M} p_j \prod_{l=1}^{D} p_{IBeta}(X_{il}|\theta_{jl})) \\ &= \sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{l=1}^{D}(log(p_j) + logp_{IBeta}(X_{il}|\theta_{jl}))\end{aligned} \tag{63}$$

In the E-step, we compute the conditional expectation of log-likelihood, which is reduced to the computation of the posterior probabilities meaning the probability that a vector $\overrightarrow{X}_i$ is assigned to a cluster j, such as following

$$p(j|\overrightarrow{X}_i, \Theta, \overrightarrow{p}_j) = \frac{p_j p(\overrightarrow{X}_i|\theta_j)}{\sum_{j=1}^{M} p_j p(\overrightarrow{X}_i|\theta_j)} = \frac{p_j \prod_{l=1}^{D} p_{IBeta}(X_{il}|\theta_{jl})}{\sum_{j=1}^{M} p_j \prod_{l=1}^{D} p_{IBeta}(X_{il}|\theta_{jl})} \tag{64}$$

where $\overrightarrow{p}_j = (p_1, ..., p_M), p_j > 0$ and $\sum_{j=1}^{M} p_j = 1$

Hence, we have

$$logp(\mathcal{X}|\Theta, \overrightarrow{p}_j) = \sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{l=1}^{D} p(j|\overrightarrow{X_i}, \Theta, \overrightarrow{p_j})(log(p_j) + logp_{IBeta}(\overrightarrow{X_i}|\theta_{jl})) \tag{65}$$

Whence, the conditional expectation of the complete-data log likelihood

$$\mathbb{Q}(\mathcal{X}, \Theta, \overrightarrow{p}_j, \Upsilon) = logp(\mathcal{X}|\Theta, \overrightarrow{p}_j) + \Upsilon(1 - \sum_{j=1}^{M} p_j) \tag{66}$$

where Υ is the Lagrange multiplier.

We move forward in maximizing the log-likelihood function by finding the roots to its derivations with respect to the set of parameters. The mixture weights can be easily estimated as follows

$$p_j = \frac{\sum_{i=1}^{N} p(j|\overrightarrow{X}_i, \Theta, \overrightarrow{p}_j)}{N} \tag{67}$$

Regarding the derivatives with respect to α_{jl} and β_{jl}, we have

$$\begin{aligned} \frac{\partial \mathbb{Q}}{\partial \alpha_{jl}} &= \sum_{i=1}^{N} p(j|\overrightarrow{X}_i, \Theta, \overrightarrow{p}_j) \frac{\partial logp_{IBeta}(\overrightarrow{X}_i|\theta_{jl})}{\partial \alpha_{jl}} \\ &= \sum_{i=1}^{N} p(j|\overrightarrow{X}_i, \Theta, \overrightarrow{p}_j)(\Psi(\alpha_{jl} + \beta_{jl}) - \Psi(\alpha_{jl}) + log(\frac{X_{il}}{1 + X_{il}})) \end{aligned} \tag{68}$$

$$\begin{aligned} \frac{\partial \mathbb{Q}}{\partial \beta_{jl}} &= \sum_{i=1}^{N} p(j|\overrightarrow{X}_i, \Theta, \overrightarrow{p}_j) \frac{\partial logp_{IBeta}(\overrightarrow{X}_i|\theta_{jl})}{\partial \beta_{jl}} \\ &= \sum_{i=1}^{N} p(j|\overrightarrow{X}_i, \Theta, \overrightarrow{p}_j)(\Psi(\alpha_{jl} + \beta_{jl}) - \Psi(\beta_{jl}) + log(\frac{1}{1 + X_{il}})) \end{aligned} \tag{69}$$

where $\Psi(.)$ is the digamma function. We can clearly see that a closed form solution to estimate θ_{jl} does not exist. Therefore, we ought to use the Newton-Raphson method [157]

such as

$$\theta_{jl}^{old} = \theta_{jl}^{new} - H_{jl}^{-1} \tag{70}$$

where H_{jl} is the Hessian matrix associated with $\mathbb{Q}(\mathcal{X}, \Theta, \overrightarrow{p}_j, \Upsilon)$ with first derivatives vector $G_{jl} = \left(\frac{\partial \mathbb{Q}(\mathcal{X}, \Theta, \overrightarrow{p}_j, \Upsilon)}{\partial \alpha_{jl}}, \frac{\partial \mathbb{Q}(\mathcal{X}, \Theta, \overrightarrow{p}_j, \Upsilon)}{\partial \beta_{jl}} \right)$

$$H_{jl} = \begin{pmatrix} \frac{\partial^2 \mathbb{Q}(\mathcal{X}, \Theta, \overrightarrow{p}_j, \Upsilon)}{\partial \alpha_{jl}^2} & \frac{\partial^2 \mathbb{Q}(\mathcal{X}, \Theta, \overrightarrow{p}_j, \Upsilon)}{\partial \alpha_{jl} \partial \beta_{jl}} \\ \frac{\partial^2 \mathbb{Q}(\mathcal{X}, \Theta, \overrightarrow{p}_j, \Upsilon)}{\partial \alpha_{jl} \partial \beta_{jl}} & \frac{\partial^2 \mathbb{Q}(\mathcal{X}, \Theta, \overrightarrow{p}_j, \Upsilon)}{\partial \beta_{jl}^2} \end{pmatrix} \tag{71}$$

with the following second and mixed derivatives

$$\frac{\partial^2 \mathbb{Q}(\mathcal{X}, \Theta, \overrightarrow{p}_j, \Upsilon)}{\partial \alpha_{jl}^2} = \sum_{i=1}^{N} p(j|\overrightarrow{X}_i, \Theta, \overrightarrow{p}_j) \Big(\Psi'(\alpha_{jl} + \beta_{jl}) - \Psi'(\alpha_{jl}) \Big) \tag{72}$$

$$\frac{\partial^2 \mathbb{Q}(\mathcal{X}, \Theta, \overrightarrow{p}_j, \Upsilon)}{\partial \beta_{jl}^2} = \sum_{i=1}^{N} p(j|\overrightarrow{X}_i, \Theta, \overrightarrow{p}_j) \Big(\Psi'(\alpha_{jl} + \beta_{jl}) - \Psi'(\beta_{jl}) \Big) \tag{73}$$

$$\frac{\partial^2 \mathbb{Q}(\mathcal{X}, \Theta, \overrightarrow{p}_j, \Upsilon)}{\partial \alpha_{jl} \partial \beta_{jl}} = \sum_{i=1}^{N} p(j|\overrightarrow{X}_i, \Theta, \overrightarrow{p}_j) \Big(\Psi'(\alpha_{jl} + \beta_{jl}) \Big) \tag{74}$$

3.3.3.3 MAP estimation

A standard choice for the mixing parameters vector $\overrightarrow{p}_j$ is the Dirichlet distribution (Dir), given its definition on the simplex $\{(p_1, ..., p_M) : \sum_{j=1}^{M-1} p_j < 1\}$[209]. We pick the same distribution for both initial and transition probabilities π and B as well as for the observation weights such that

$$\pi \sim Dir(\pi|p)$$

$$B_i \sim Dir(B_i|b_i)$$

$$\omega_i \sim Dir(\omega_i|\Omega_i)$$

$$\rho_l \sim \frac{1}{\mathbf{Z}} e^{-k_l \rho_l}$$

where $\mathbf{Z}$ is the normalizing constant and Ω_i is the hyperparameter vector of the observation weights.

The prior for the mixing parameters vector $\overrightarrow{p}_j$ can be written as follows

$$p(\overrightarrow{p}_j|M, \overrightarrow{\Delta}) = \frac{\Gamma(\sum_{j=1}^{M} \Delta_j)}{\prod_{j=1}^{M} \Gamma\Delta_j} \prod_{j=1}^{M} p_j^{\Delta_j - 1} \tag{75}$$

where $\overrightarrow{\Delta} = \{\Delta_1, ..., \Delta_M\}$ is the Dirichlet parameters vector.

For the GID parameters, the Gamma (G) function is chosen as a prior given its exponential nature under the assumption of parameters independence. Thus we have the priors

$$p(\alpha_{jl}) = G(\alpha_{jl}|\nu_{jl}, \vartheta_{jl}) = \frac{\vartheta_{jl}^{\nu_{jl}}}{\Gamma(\nu_{jl})} \alpha_{jl}^{\nu_{jl}-1} e^{-\nu_{jl}\alpha_{jl}} \tag{76}$$

$$p(\beta_{jl}) = G(\beta_{jl}|\kappa_{jl}, \varsigma_{jl}) = \frac{\varsigma_{jl}^{\kappa_{jl}}}{\Gamma(\kappa_{jl})} \beta_{jl}^{\kappa_{jl}-1} e^{-\kappa_{jl}\beta_{jl}} \tag{77}$$

where ν, ϑ, κ and ς are positive hyperparameters.

3.4 Experiments and results

In this section, extensive experiments are conducted and we have implemented several real-world topical yet challenging applications using the FSHMM with GID emission probabilities. We are mainly comparing our new approach to its classical FSHMM competitors and other new adaptations that we executed for the sake of comparison and testing, e.g., inverted Dirichlet-based FSHMM (ID-FSHMM) and Dirichlet-based FSHMM (Dir-FSHMM), not

to mention the widely used GMM-FSHMM. It is noteworthy that the learning of the mentioned adaptations has been based on the same methodology described in the previous section to learn the GID mixture-based FSHMM. Two real-world applications, facial expressions recognition, and scene categorization are here tested and explained. Experimentation and results presented in this section have been yielded on a macOS environment over a 2.3 GHz Dual-Core Intel Core i5 MacBook Pro, using Python.

3.4.1 Facial expressions recognition

Facial expression recognition is a powerful process that usually commends the way we interact with other people. It is one of the non-verbal communication media that humans naturally use in everyday interactions. Besides its role in supporting humans' understanding of people's intentions and feelings, facial expression recognition plays a major role in making decisions about relationships or situations. For all these reasons, substantial efforts have been devoted to automating this recognition [75, 86] and using it as a fundamental step within multiple decision-making systems. A human being is naturally empowered to interpret these expressions and make his decisions in a real-time matter. Nonetheless, this task is still approached as a complex and challenging process in the field of machine learning [118, 97]. Facial Expression Recognition is applied in a wide range of contexts and is used in numerous applications such as Human-Computer Interaction [214], student automatic E-learning [169], Behavioural Science [143], psychological studies [151], image understanding, and synthetic face animation. The principal purpose of researchers working on these applications is to produce automated systems capable of automatically recognizing the emotional state of a person and further draw an analysis or make a decision based on a specific context [65].

3.4.1.1 HMM-based facial expression recognition

Classification is the most significant part of a facial expression recognition system [230]. Methods applied to classify this type of images are generally sorted into static or dynamic [256]. Static methods are based on the information acquired from the input image, they take benefit from the use of support vector machines, neural network, Bayesian network to perform the assigned task. HMMs are dynamic classifiers that exploit temporal records to analyze facial expressions. Hence, they are highly recommended by psychological experiments carried out as indicated in [13].

As early as 1990, [215] used HMMs to come up with a solution for the challenging task of automating facial expressions recognition. Authors in [215], used an HMM along with the integration of a priori structural knowledge with statistical information. HMMs offer a perfect analogous representation to the experience of observing a particular feeling through the way they statistically handle the behaviour of an observable symbol sequence. These models provide a specification of the probability distribution over all hidden events that are behind a certain symbol sequence. The performances of HMMs when dealing with such challenges are promising, especially in the case when they learn through an entire sequence of images describing a group of actions taken by a person when undergoing a certain feeling. The learning process is conducted in a much smoother way thanks to HMMs capability of handling the Spatio-temporal nature of the debated application. In fact, there is a metaphorical resemblance between human performance when naturally processing the recognition task, and the stochastic nature of the HMM process inasmuch as it analyses the measurable (observable) actions in order to infer the immeasurable (hidden) feelings of the person.

In this particular context, we choose to apply our model on the challenging Dollar facial expression database [67], see figure16. This application is unprecedented as it uses for the first time an embedded model-based feature selection into the HMM structure.

Figure 16: Samples of facial frames from the Dollar facial expressions dataset

3.4.1.2 Experimental trials and results

The Dollar database is composed of 192 sequences performed by 2 individuals, each expressing 6 different basic emotions 8 times under 2 lighting setups. Each subject starts with a neutral expression, then expresses emotion, and returns to a neutral expression. For our simulations, we follow the experimental setting considered in [178], which consists of using three peak frames of each sequence for 6-class expression recognition (576 images: anger, disgust, fear, joy, sadness, and surprise). The pre-processing steps are also the same and consist of extracting features from the whole face region by cropping original face images into 110×150 pixels, keeping only the central part of facial extraction. A Local Binary Pattern (LBP) descriptor [190], is used for feature extraction. More specifically, each cropped face image is first divided into small regions from which LBP histograms are extracted and then concatenated altogether into a single feature histogram representing the face image. We use a 59-bin LBP operator in the (8, 2) neighbourhood. This means 8 sampling points on a circle of radius 2, then we divide each image (110 × 150) into 18 × 21 pixels regions. Therefore, each face image is divided into 42 (6 × 7) regions and is then represented by LBP histograms with a length of 2478 (59 × 42). After that, these histograms are normalized. The procedure is applied as the one originally used in [223]. We figured that if we reduce the feature vector the algorithm tends to diverge early, however since features will later be reduced by the model itself, and in order to give the algorithm the time to learn we will not reduce the feature vector ourselves and will leave it as it

is. The obtained feature vector is actually handled with our GIDHMM where the feature saliency is considered. Hence, recognition is carried out via a single HMM recognizer. A collection of HMMs each representing a different subject is matched against the test image and the highest match is selected as explained in figure 17. For the sake of comparison, we conduct several experiments with different used models, with and without taking into consideration feature relevancy, then we report the results including features relevancies and the confusion matrix for each experiment.

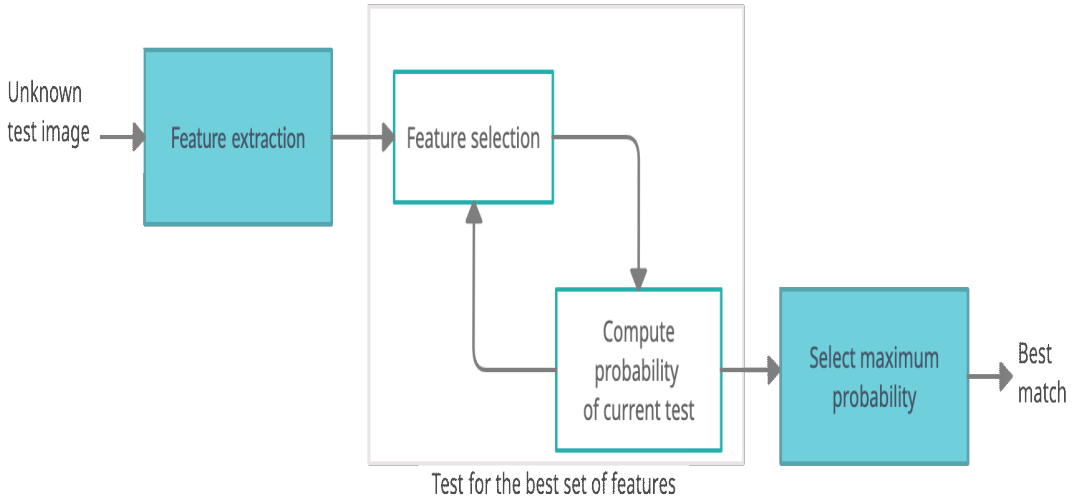
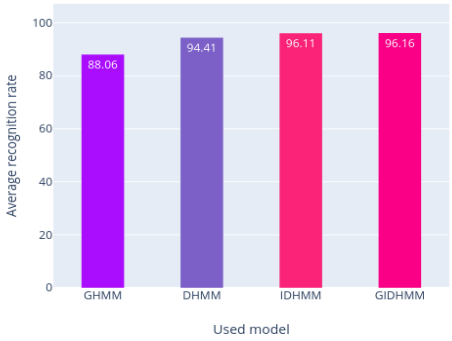


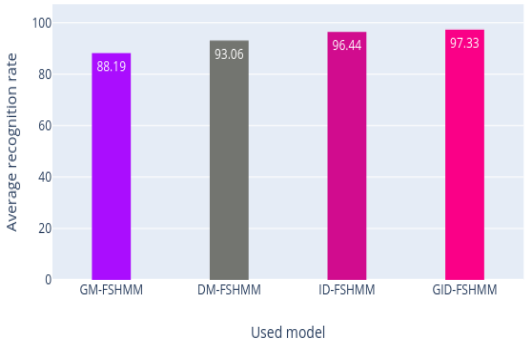
Figure 17: Block diagram for FSHMM-based face recognizer

In order to evince the advantages of the proposed approach and most importantly underline the crucial role of feature selection integration in improving results, we compare the latter with other emotion recognition approaches that are mainly based on mixture models. These approaches, have been personally implemented, and include inverted Dirichlet-based HMM without feature selection (IDHMM) [178], inverted Dirichlet-based HMM with feature selection (ID-FSHMM), generalized inverted Dirichlet-based HMM without feature

selection (GIDHMM), and generalized inverted Dirichlet-based HMM with feature selection (GID-FSHMM). On top of that, we put a special emphasis on the improvements noticed from the use of GID mixtures measured against the Gaussian mixtures-based HMM with feature selection (GM-FSHMM). Results obtained are displayed in Figure 18, where we present the average recognition rates for the different used methods.



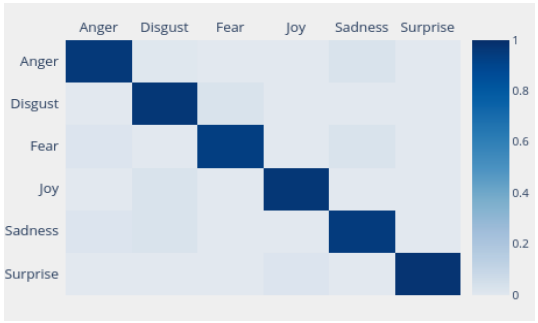
(a) Average recognition rate of each used model for facial recognition application without applying feature selection



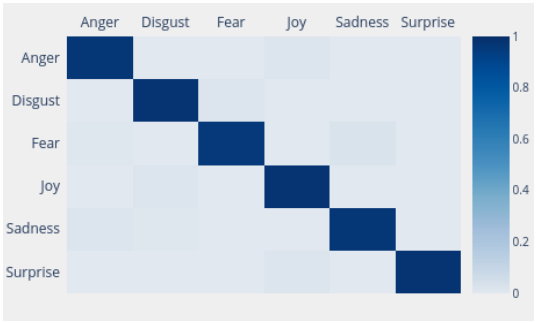
(b) Average recognition rate of each used model for facial recognition application with applying feature selection

Figure 18: Average recognition rates for facial expressions recognition with and without applying feature selection

There is an interesting observation to make after nailing these results, that is the amelioration in average recognition results after using the GIDHMM as a model. Initially, using only the latter allowed for a slight but worth mentioning amelioration in the average recognition rate from 96.11% to 96.16%, this itself shows that the GIDHMM is better in modelling our data than the IDHMM.



(a) GIDHMM without feature selection



(b) GIDHMM with feature selection

Figure 19: Confusion matrices for facial expressions recognition with and without applying feature selection for GID-FSHMM

Further, we noticed when incorporating feature selection into our previously established IDHMM model, the average recognition rate improved from 96.11% to 96.44%. On top of that, we plainly succeeded to bear out our theoretically anticipated projections regarding recognition rates. In fact, the feature selection-based GIDHMM executed the task of recognizing each facial expression considerably better than GIDHMM without FS. This conclusion comes after several trials on each emotion type separately. The individual recognition rates per category and confusion matrices for the mentioned trials are displayed in table 6 and figure 19.

	Anger	Disgust	Fear	Joy	Sadness	Surprise
Anger	0.96	0.01	0	0	0.03	0
Disgust	0	0.97	0.03	0	0	0
Fear	0.02	0.01	0.94	0	0.03	0
Joy	0	0.03	0	0.97	0	0
Sadness	0.02	0.03	0	0	0.95	0
Surprise	0	0	0	0.02	0	0.98

(a) Recognition rates without applying feature selection

	Anger	Disgust	Fear	Joy	Sadness	Surprise
Anger	0.97	0	0	0	0.03	0
Disgust	0	0.98	0.02	0	0	0
Fear	0.01	0	0.96	0	0.03	0
Joy	0	0.02	0	0.98	0	0
Sadness	0.02	0.01	0	0	0.97	0
Surprise	0	0	0	0.02	0	0.98

(b) Recognition rates when feature selection is applied

Table 6: Detailed recognition rates in the case of facial recognition application with and without applying feature selection for GIDHMM

As indicated in Figure 18, average recognition rates for GIDHMM with and without feature selection integration are respectively 97.33% and 96.16% with the corresponding average misclassified images of 22.11 and 15.32 per dataset. There is also a significant variation in the run time when using each of the cited methods, 36.4 min for GIDHMM, and 41.6 min for GIDHMM-FS.

Needless to say, the integration of feature selection in our models brought an obvious amelioration to the yielded results for all adopted distributions, this shows the important

role of taking into consideration the feature saliency when dealing with image classification tasks. Further investigations with respect to features relevancy are conducted in the following applications to emphasize this role.

3.4.2 Scene categorization

Recently, there has been an abundance of research works and experimental trials aiming to bridge the semantic gap between the perceptual ability of human vision and the capacity of automated systems when performing the same related tasks. This challenge is prompted by the impressive trait of the human visual system to rapidly, accurately, and comprehensively recognize and understand a complex scene [99, 98, 251]. Thus, it would be worthwhile if each image in a studied collection could be annotated with semantic descriptions allowing for a better automatic interpretation and hence an improved visual recognition ability. In this section, we work on a challenging problem related to the mentioned area of research, which is recognizing scene categories. Visual scenes classification has many applications in robot navigation and robot path planning [232], video analysis [244], content-based image retrieval [58].

Inasmuch as this application is complex due to the variety of scenes and variations of viewing angles and changing backgrounds, choosing efficient features plays a major role in the accuracy level of the recognition task.

In this section, we test the effectiveness of our proposed feature-selection-based GIDHMM, in categorizing images of real-world scenes from the notorious MIT benchmark [192] [1]. The indicated database contains about 2688 diverse outdoor scene images in colours from 8 categories: coast (360 images), mountain (374 images), forest (328 images), open country (410 images), inside city (308 images), street (292 images), tall building (356

[1] http://people.csail.mit.edu/torralba/code/spatialenvelope/

images) and highways (260 images). Images come in 256 $\times$ 256 pixels resolution. We choose to randomly select 200 images from each category for training and leave the rest for testing purposes. Figure 20, shows example images from the MIT outdoor data set.

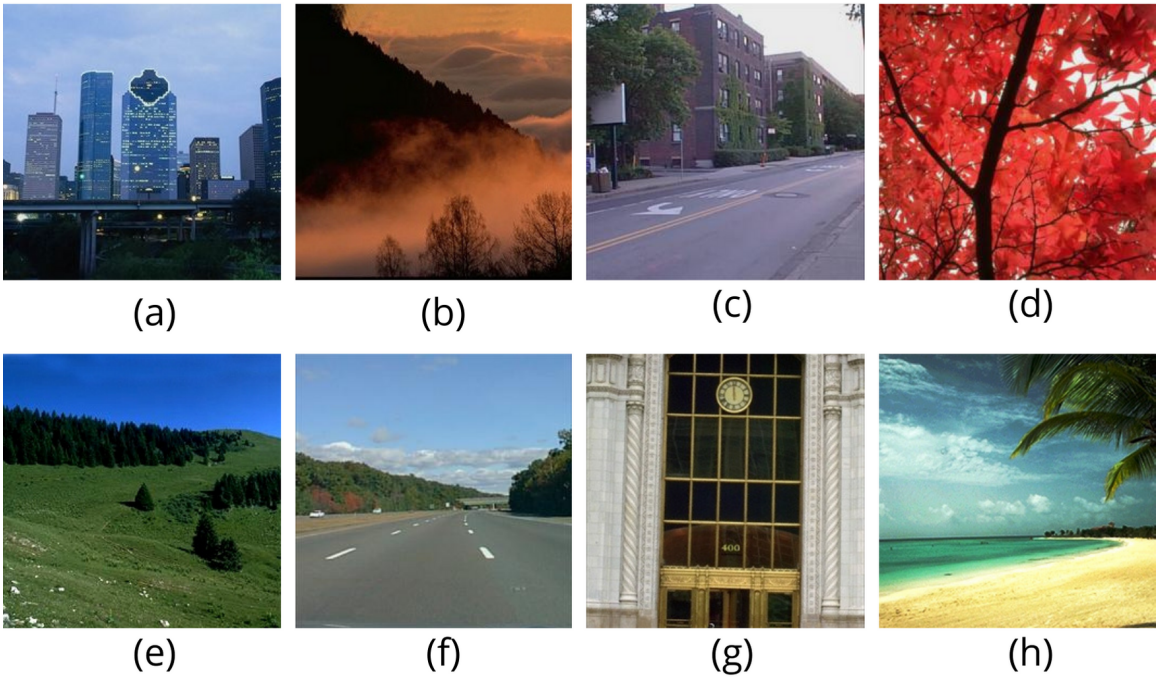


Figure 20: Sample images from the 8 categories MIT data set: (a) Tall buildings, (b) Mountain, (c) Street, (d) Forest, (e) Open country, (f) Highway, (g) Inside city, (h) Coast.

A crucial step for the scene categorization task is feature extraction. For this matter, we adopt a process where we normalize images which will afterward be represented each by a collection of local image patches. These patches are scanned and low-level feature vectors are thereafter extracted. We then use the bag of words approach (BOW) method, adapted likewise by [249] for scene classification, in which Yang et al. mapped the key points of an image into visual words. Hence, each image could be represented as a "*bag of visual words*" BoVW, and in this instance as a vector of counts of each visual word in that image.

This will allow for an overall representation for each image through a feature vector, upon which the task of image classification is built. Following [34], and after obtaining the intended histograms, we apply a probabilistic Latent Semantic Analysis (pLSA) [123, 125] in order to represent each image by a D-dimensional vector with D being the number of latent aspects (hidden aspects, features, or hidden states in our analogy). Ultimately, our objective is to identify the right category for each image by applying our previously developed model.

In this work, we use dense SIFT 16×16-pixel patches calculated over a grid of 8 pixels. Besides, we build a bag of words dictionary using a K-means algorithm [125] to cluster our descriptors in a V visual words vocabulary. For each SIFT point in a candidate image, the nearest neighbour within the vocabulary is computed, and thereby a feature vector with dimension V is built. Hence, each image can be represented as a frequency histogram over the V visual words. As previously explained, in this work we apply pLSA to allow for a description through a D-dimensional vector where D is the number of aspects. We employ our GIDHMM to model the set of images designated for training. We compute the class-relationship likelihood of each input image and classify it to the class that maximizes its likelihood. In our approach, each image class is characterized by its own behaviour, therefore each class is described by its own HMM. That being the case, for each scenery type, a distinct 8-state HMM is trained. Experiments are carried out 30 times with the average accuracy reported for both feature-saliency-based and non-feature-saliency-based methods.

Through these experiments, we aim to evaluate not only the effectiveness of GIDHMM measured against IDHMM and GHMM but also the effectiveness of embedding the process of feature selection in the core of each of the aforementioned models. Experiments are chosen to be conducted in the following order: first, we compare the performance of

GIDHMM, IDHMM, and GHMM without taking into consideration the relevancy of features. Then we reproduce the same experiments by taking into account feature relevancy. Table 7 presents the confusion matrix when GIDHMM is applied without feature selection. According to this table, we get an average accuracy of 91.37%. On the other hand, Table 8 shows the confusion matrix when GIDHMM is used along with feature selection: the average accuracy is 93.12%.

	Tall building	Mountain	Street	Forest	Open country	Highway	Inside city	Coast
Tall building	0.94	0.02	0	0	0	0	0.04	0
Mountain	0.01	0.92	0	0	0.07	0	0	0
Street	0	0	0.92	0	0	0.06	0.02	0
Forest	0	0.02	0	0.95	0.03	0	0	0
Open country	0	0.03	0	0.01	0.87	0	0	0.09
Highway	0	0.01	0	0	0.03	0.88	0	0.08
Inside city	0.01	0.01	0.05	0	0.03	0	0.90	0
Coast	0	0.01	0.01	0	0.05	0	0	0.93

Table 7: The confusion matrix in the case of MIT scene recognition problem when applying GIDHMM without feature selection

	Tall building	Mountain	Street	Forest	Open country	Highway	Inside city	Coast
Tall building	0.96	0.02	0	0	0	0	0.02	0
Mountain	0.01	0.95	0	0	0.04	0	0	0
Street	0	0	0.93	0	0	0.06	0.01	0
Forest	0	0.02	0	0.96	0.02	0	0	0
Open country	0	0.03	0	0.01	0.90	0	0	0.06
Highway	0	0	0	0	0.02	0.91	0	0.07
Inside city	0.02	0.01	0.05	0	0.03	0	0.89	0
Coast	0	0.01	0.01	0	0.03	0	0	0.95

Table 8: The confusion matrix in the case of MIT scene recognition problem when applying GIDHMM with feature selection

Results of other experimentation on the different used models are presented in Table 9 and confirm our previous assumptions about the role of feature selection in improving recognition rates. Our algorithm analyzed all extracted features and succeeded to determine their saliency, hence the use of the better features yielded better results. Figure 21 shows the feature saliencies obtained by our GID-FSHMM.

Method	Average recognition Rate (%)	Integrating Feature Selection
GHMM	87.60	88.14
DHMM	88.79	89.05
IDHMM	90.01	90.66
GIDHMM	**91.37**	**93.12**

Table 9: Average recognition rates for different used HMMs in the context of natural scenes recognition, with and without feature selection.

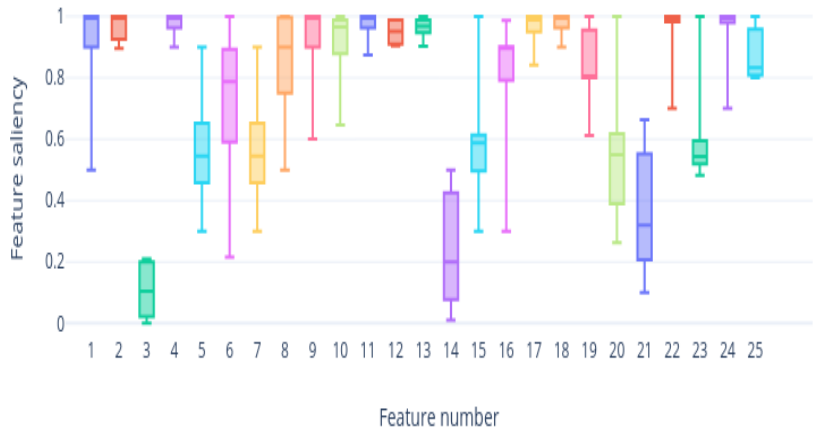


Figure 21: Feature saliencies obtained in the case of natural scenes recognition problem when performing feature selection-based GIDHMM

3.5 Conclusion

While there are multiple general techniques of applying feature selection, and despite the buildup of standardized procedures for features and dimensionality reduction, the literature reveals time and again that custom methods keep outperforming the general methods. Besides there is an overwhelming need for some sort of supervised data and knowledge when applying general feature selection models. In our context, this supervised data can take the form of information about the class, a label of each observation, or even a piece of knowledge about the latent variable. This additional information is often not readily accessible especially in areas where mixture models and HMMs are applied, considering that those models account for the fact that supervised data is unavailable. Therefore, unsupervised feature selection methods are essentially needed when using HMMs, allowing for significantly better performing models compared to those based upon general feature selection methods. Further, the interest in adopting the GID for modelling our data arose from the limitations encountered when inverted Dirichlet was adopted, in particular its restraining

strictly positive covariance. In this paper, we proposed a framework in which all the aforementioned problems are addressed simultaneously in the case of automatic recognition. The developed approach applies feature selection to a GID-based HMM. Parameters are learned via a MAP method adding a huge advantage in raising both accuracies of parameter estimates and feature saliencies. Experimental results involving challenging real-life applications such as facial expressions recognition and natural outdoor scene recognition showed that the proposed approach is highly promising. Future works are intended to be done in the near future, extending this work to different flexible distributions and considering online learning for more precise results.

Chapter 4

Online learning of Inverted Beta-Liouville HMMs for Anomaly Detection in Crowd Scenes

Abnormal behavior detection in crowd scenes has become a topic of great interest in the computer vision field. In this paper, an online inverted Beta-Liouville-based hidden Markov model (HMM) is presented and applications related to crowd scenes analysis illustrating its performance are demonstrated. Often referred to as "adaptive" or "recursive", the online approach is a topic of great interest in time series modeling. Here, we put forward an online parameter estimation algorithm based on an Expectation-Maximization (EM) methodology to tackle the problem of activity recognition. In some cases, when processing large sets of data or big data streams, EM becomes intractable due to the acute demand in terms of data but mostly to the need of providing the whole data set at each iteration of the algorithm. In this manuscript, we present a new online EM algorithm for model parameter inference. This algorithm allows the update of parameter estimates after several increments of observation are processed (online). We also give positive vectors a special focus by using

Inverted Beta-Liouville (IBL) mixtures as emission probabilities for our HMM. Using this distribution is expected to improve the modeling accuracy considering that it contains inverted Dirichlet distribution as a special case, hence the additional flexibility. Experimental results on several publicly available crowd video data sets verify the effectiveness of the proposed model.

4.1 Introduction

Data categorization is rapidly becoming one of the most important parts of data analysis, particularly with the exponential growth of data under all sorts of formats. Thereby, it is crucial to study and discover hidden patterns in order to extract valuable information promoting accurate and solid decision making.

When modeling data, it is a notable fact that Gaussian mixture models (GMMs) are not always the perfect solution for all data types. Through HMM deployment, most existing related works have not considered the characteristics of data sets. In fact, most of the work present in the literature relies on the use of Gaussian distributions. Although HMMs were mainly developed for discrete and Gaussian data [203], diversity of applications in contexts and domains such as activity recognition, image categorization, and dynamic forecasting, increased the necessity of modifying the underlying HMM model so that it efficiently suits those new data types [178, 221].

Thanks to the proliferation of carried research on these distributions and their mathematical simplicity, most finite mixture models mainly consider Gaussian as their basic distributions. Nevertheless, it is undeniable that the least appropriate way of modeling non-Gaussian data is to use Gaussian distributions. For example, inverted Dirichlet or generalized inverted Dirichlet [24, 80, 78] can often outperform the Gaussian mixture model for modeling positive vectors in many applications such as image categorization,

human action video recognition, etc. Recently, numerous works have been achieved in order to model positive vectors based on inverted Dirichlet mixture models [24, 178]. However, the inverted Dirichlet distribution has a very restraining covariance structure that significantly limits its flexibility. In our work, we propose to model positive vectors based on a finite mixture model with inverted Beta-Liouville (IBL) distributions embedded into the framework of HMMs as emission probabilities.

Inverted Beta-Liouville mixture models have recently arisen as an efficient way to model positive vectors [41]. Thanks to its general covariance structure and its smaller number of parameters compared to the inverted Dirichlet and generalized inverted Dirichlet, IBL has proven its effectiveness when dealing with positive vectors modeling [127]. Originally derived from the Liouville distributions family [114]. As earlier mentioned, one of the main advantages of the IBL is its general covariance structure that can either be positive or negative. It's noteworthy to mention that the discussed distribution has not been extensively investigated and that only a handful of works have adopted it, giving more room to further exploitation of this surprisingly underrated distribution. Even more effectively, this choice is mainly motivated by the fact that the IBL distribution contains inverted Dirichlet distribution as a special case and therefore can provide more flexibility compared to previously investigated distributions [178]. Also, compared with Gaussian which can only approximate symmetric distributions, IBL allows both symmetric and asymmetric distributions.

The work presented in this manuscript can be viewed intellectually at two different levels. First, it allows the application, for the first time to the best of our knowledge, of IBL-based HMMs to effectively handle positive vectors, second, it proposes to undertake online-based learning of parameters by applying an online EM procedure for HMMs.

The remainder of this chapter is organized as follows: in section 2, we present some of the work related to online and incremental learning, and we discuss the choice of application in this paper. Section 3 presents HMMs, their formulation, and the online EM derivations. Section 4 explains the choice of the IBL mixture models and details derivations and parameters estimation. Then, in section 5 we present our applied model as well as results and interpretations. Finally, we conclude with some insights and future work perspectives.

4.2 Related work

The performance of hidden Markov models (HMMs) is often acclaimed through their massive use in several complex real-world applications namely image categorization [81], action recognition [92], occupancy estimation in smart buildings [178] and unusual events detection [80]. HMMs are highly capable of representing probability distributions corresponding to these complex real-world phenomena when they are fed an adequate number of states as well as a sufficiently rich set of data. Nevertheless, the mentioned applications tend to often drain HMMs' performance particularly when results need to be inferred from very long sets of data such as videos in an action recognition context. In fact, in the context of HMMs, analyzing large sets of training data is costly, laborious, and long sustained. Thus, there is often not enough analyzed data to be representative of the underlying distribution, causing the HMM to incorporate some uncertainty.

In such cases, it is suitable to update the model's parameters online [229]. Online learning of new data sequences permits the adaptation of the HMM parameters whilst new data becomes available. This way of feeding data to the model is also called an incremental method. It is actually common for a model to be fed additional data after its training. This allows for a more adaptation of HMMs as a result of newly acquired data. Therefore, incremental learning is an undeniable asset to refine HMMs' behavior toward any novelties encountered in the environment and thus reducing their level of uncertainty by maintaining

a high level of performance.

When applying incremental learning for HMM parameters estimation, there are commonly standard techniques used that mostly involve batch learning. Those techniques can either rely on specialized EM techniques [63] such as Baum-Welch (BW) algorithm [21] or on numerical optimization techniques such as the Gradient Descent algorithm [153], where regardless of the used technique, parameters are estimated after numerous training repetitions prior to maximizing an objective function over certain independent validation data. In most cases, when applying a batch learning technique, a fixed-length sequence $O = o_1, o_2, .., o_T$ of T training observations, o_i is hypothetically available during the whole learning process. If we suppose that O is assembled into a block D of training data, each training iteration involves observing all sub-sequences in D prior to updating HMM parameters. When a new block of data comes through, the previously trained HMM cannot accommodate the second batch without accumulating and storing all the training data in memory. It will eventually train again for the beginning making use of all the cumulative data involving both batches. This procedure is deemed to be necessary in order to prevent any sort of corruption of the previously acquired knowledge, and that could compromise the HMM performance. Notwithstanding, there are clearly some significant costs relating to processing time and storage requirements when using batch learning methods. Time and memory complexity would grow linearly with the length and number of training observation sequences and quadratically with the number of HMM states.

As a viable alternative, numerous online learning techniques have been proposed in the literature, this includes techniques based on EM [46, 173] where numerical optimization and recursive estimations are performed, and EM variants such as BOEM (Block Online EM) [148]. These methods assume the observation is a stream of data and are particularly used in situations where training symbols are organized into a block of one or more sub-sequences. Their parameters are re-estimated upon observing each

new sub-sequence of symbols. Some of the aforementioned techniques are tailored to update HMM parameters at a symbol level, also perceived as recursive or sequential estimation techniques. Symbol-wise updated techniques are designed for situations in which training symbols are received one at a time where parameters are then re-estimated upon observing each new symbol. Across the full range of contexts, HMMs parameters are updated from new training data, beyond any requirement for access to the formerly learned training data and most plausibly preventing corruption of any previously acquired knowledge [48]. In this manner, the main takeaway of the stated techniques is essential to allow sustaining a high level of performance while preserving the memory requirements, given the fact that storing data from previous training phases is completely unnecessary. Besides, bearing in mind that training is only performed on the new training sequences, and not all accumulated data, online learning also provides lower time complexity when learning new data. In this work, we aim to study the effectiveness of online-based HMMs compared to standard-learning-based HMMs when used along with a remarkably interesting distribution, that is the Inverted Beta-Liouville, as emission probabilities.

Further to the raised interest in online learning as a technical concern, the studies carried out in this work revolve around analyzing human-related visual data. We choose to bring a special focus on disclosing information from looking at videos with humans doing certain activities and analyzing, in particular, security surveillance to predict certain anomalies. Indeed, it would be of great help to assist in detecting either normal or abnormal events or behaviors and use this as a starting point to make decisions such as in the contexts of smart cities where there is a growing need to improve security. In fact, this can be achieved by quickly and accurately identifying criminal activities in a real-time fashion [50]. Similarly, in an entertainment environment, activity recognition can notably improve users' experience by automatically recognizing different player's actions during

a game of tennis or a soccer game for example [141, 219], with the goal of understanding the action of each player and how they interact with each other.

What is challenging in performing this type of analysis, is that crowded scenes and dynamic environments are bound to a degraded performance as soon as the crowd becomes too dense [79]. In fact, the number of independent objects moving at the same time and the occlusions it involves degrades the performance of detection. Additionally, the dynamic background is an important restriction when it comes to tracking movements.

As far as HMMs are concerned, modeling normal scenes and determining whether an unseen video sequence deviates from normality is an achievable task, which serves perfectly the anomaly detection aim. In the work of Bettini et al. in [28], the features used are histograms that can be seen as positive vectors once extracted. The likelihood criterion for anomaly detection is somehow efficient despite the simple adaptive threshold adopted by the classifier. The work is obviously relying on different processes leading together to detection results, which constitutes a clear limitation to the improvement of the global approach and the use of a standardized, unique model, capable of providing a more compact representation of the data and thus a more accurate anomaly detection. In a related context, the author in [245] exploits the notion of profiling and online anomaly sampling to model dynamic scenes in a way that optimizes the intrusion detection rate by refraining from using any manual labeling of the training data set. The method relies on a Dynamic Bayesian Network (DBN) to model each behavior pattern. Further, an online Likelihood Ratio Test (LRT) method is used to detect abnormal behavior, while normal behavior is recognized when sufficient visual evidence is available. The mentioned procedure lacks accuracy since some events can sometimes be undetected by the model due to missing visual evidence or ambiguities between event classes. This can be avoided by taking the temporal information

into consideration by developing a Baum-Welch EM algorithm to the mixture of DBNs to learn the behavior model directly rather than taking a phased approach such as the one adopted. Andrea et al. [14, 15, 16] used HMMs with Gaussian mixtures to characterize the normal behavior of a crowd by learning normal motion patterns from the optical flow of image blocks. The method relied mainly on Principal Component Analysis (PCA) to build feature prototypes, along with spectral clustering to find the optimal number of models to group video segments containing similar motion patterns. An HMM was trained for each model and used for event recognition and anomaly detection.

4.3 Hidden Markov Models

Hidden Markov Models are described according to Ghahramani [110], as an ubiquitous tool to model time series data. They have been used for decades in speech recognition systems as well as artificial intelligence and pattern recognition applications. These models are a generalization of mixture models [112]. In fact, the probability density functions overall observable states defined by an HMM, are considered as a mixture of densities defined by each state.

HMMs allow us to represent probability distributions over sequences of observations, with the assumption that observations are discrete. An observation at time t is denoted by the variable O_T.

Hidden Markov Models are governed by two main properties. First, it assumes that the observation at time t is generated by some process whose state h_t is hidden from the observer. Second, it assumes that the state of this hidden process satisfies the Markov property; that is, given the value of h_{t-1}; the current state h_t is independent of all the states prior to the time $t-1$.

A hidden Markov model is characterized by a set of parameters that will be specified later in this paper. The task of the learning algorithm is to find the best set of state transitions

and emission probabilities between the states of the model. Therefore, an output sequence or a set of these sequences is given. To illustrate our model, we are first listing various HMM notations and enumerating the upcoming used work script.

4.3.1 Notations and offline EM for HMMs

We consider a HMM with continuous emissions and K states. We put $y = \{y_0, y_1, ..., y_T\}$ the sequence of observed data with $y_t \in \mathbb{R}^L$. The observation for the l-th feature at time t, which is represented by the the l-th component of y_t, is denoted by y_{lt}.

Let $x = \{x_0, x_1, ..., x_T\}$ be the sequence of hidden data. The transition matrix of the Markov chain associated to this sequence is denoted as $B = \{b_{ij} = P(x_t = j | x_{t-1} = i)\}$ and π is the initial state probability. Thus the complete data likelihood can be expressed as:

$$p(x, y|\Lambda) = \pi_{x_0} c_{x_0}(y_0) \prod_{t=1}^{T} b_{x_{t-1}, x_t} c_{x_t}(y_t) \tag{78}$$

where Λ is the set of model parameters, π_{x_0} is the initial state (x_0) probability, and $c_{x_t}(y_t)$ is the emission probability given state x_t.

The M-step aims to maximize the data log-likelihood. By denoting Z as hidden variables and X as the data, we can express the data likelihood $\mathcal{L}(\theta|X) = p(X|\theta)$ by:

$$\begin{aligned} E(X, \theta) - R(Z) &= \sum_Z p(Z|X) \log(p(X, Z)) - \sum_Z p(Z|X) \log(p(Z|X)) \\ &= \sum_Z p(Z|X) \log(p(X|\theta)) \\ &= \log(p(X|\theta)) \sum_Z p(Z|X) \log(p(X|\theta)) \\ &= \log(p(X|\theta)) = \mathcal{L}(\theta|X) \end{aligned} \tag{79}$$

with θ representing all the HMM parameters, $E(X, \theta)$ is the value of the complete-data

log-likelihood with the maximized parameters θ, and $R(Z)$ is the log-likelihood of the hidden data given the observations.

The expected complete-data log-likelihood is:

$$E(X, \theta, \theta^{old}) = \sum_{Z} p(Z|X, \theta^{old}) \log(p(X, Z|\theta)) \tag{80}$$

In the following, we take the case of a unique observation sequence, X, then the complete-data likelihood is expanded as

$$p(X, Z|\theta) = p(h_0) \prod_{t=0}^{T-1} p(h_{t+1}|h_t) \prod_{t=0}^{T} p(m_t|h_t) p(x_t|h_t, m_t) \tag{81}$$

When considering an HMM, as defined earlier in this section, where the final time T may be unbounded in the online case, offline learning consists of adjusting the model parameters to maximize the likelihood of a given training sequence $y_{0 \longrightarrow T}$. This procedure results in the following update equations that can be reviewed in detail in a previous work [178].

$$\hat{b}_{ij}^{(n+1)} = \frac{\sum_{t=1}^{T} P(x_{t-1} = i, x_t = j|y_{0 \longrightarrow T}, \hat{\theta}_n)}{\sum_{t=1}^{T} P(x_{t-1} = i|y_{0 \longrightarrow T}, \hat{\theta}_n)} \tag{82}$$

$$\hat{c}_{jk}^{(n+1)} = \frac{\sum_{t=1}^{T} P(x_t = j, y_t = k|y_{0 \longrightarrow T}, \hat{\theta}_n)}{\sum_{t=1}^{T} P(x_t = j|y_{0 \longrightarrow T}, \hat{\theta}_n)} \tag{83}$$

where $k = 1, \ldots, K$ and the probabilities on the right-hand side are conditioned on the training sequence $y_{0 \longrightarrow T}$ and on the current parameters' estimate $\hat{\theta}_n \equiv (\{\hat{b}_{ij}^{(n)}\}, \{\hat{c}_{jk}^{(n)}\})$.

Computation of these quantities can be done efficiently using the forward-backward procedure, although this will imply storing the whole training sequence.

4.3.2 Online EM for HMMs

Online learning has proven to be an effective way to improve learning, mainly in large-scale settings [35, 173]. In this work, we build upon the work presented by Mongillo et al. in [173] and Cappé in [46], to put forward an online and incremental EM algorithm for HMMs. For the matter, a recall of Cappés' online EM is desired. The latter uses a stochastic approximation approach in the scope of sufficient statistics in order to achieve a limiting EM recursion. This EM recursion is nothing but a batch-based EM algorithm with infinite data. All the parameter updates are handled in a recursive manner. This procedure is built around a forward-only smoothing recursion, in which the expected sufficient statistics needed for parameter updates are computed recursively. This can be achievable thanks to an expectation-maximization algorithm that updates and improves lower bounds on the likelihood after each observation.
In this phase, we focus on calculating the likelihood of an observation sequence of a given length to classify it. After determining the sequence category, we use the corresponding data to train a specific HMM and use its parameters to update the previously trained HMM corresponding to the said category.

The adopted method consists of applying the online EM developed in [173], which we expand to handle positive vector modeling thanks to the adoption of IBL mixtures as emission probabilities.

We here derive a version of the EM procedure that does not require the storage of the inputs by reproducing the EM update (equations 82 and 83) in terms of sufficient statistics updated recursively

4.3.2.1 Sufficient statistics for parameter estimation

The required sufficient statistics are

$$\phi_{ijk}(T;\theta) = \frac{1}{T}\sum_{t=1}^{T}\delta(y_t - k)\cdot P(x_{t-1} = i, x_t = j|y_{0\longrightarrow T},\theta) \tag{84}$$

with $1 \leq i,j \leq K$ and $1 \leq k \leq M$, where $\delta(\cdot)$ is the Kronecker delta: 1 when its argument is 0 and 0 otherwise. The prefactor $\frac{1}{T}$ ensures that $\phi_{ijk}(T;\theta)$ do not diverge for an infinitely long training sequence $T \longrightarrow \infty$.

The update equations can thus be written as follows

$$\hat{b}_{ij}^{(n+1)} = \frac{\sum_k \phi_{ijk}(T;\hat{\theta}_n)}{\sum_{jk}\phi_{ijk}(T;\hat{\theta}_n)} \tag{85}$$

$$\hat{c}_{jk}^{(n+1)} = \frac{\sum_i \phi_{ijk}(T;\hat{\theta}_n)}{\sum_{ik}\phi_{ijk}(T;\hat{\theta}_n)} \tag{86}$$

4.3.2.2 Recurrence relations

$$\phi_{ijk}^{\gamma}(T) = \frac{1}{T}\sum_{t=1}^{T}\delta(y_t - k)\cdot P(x_{t-1} = i, x_t = j, x_T = \gamma|y_{0\longrightarrow T}) \tag{87}$$

where we drop the explicit independence on the model parameters θ assumed to be constant and hence $\sum_\gamma \phi_{ijk}(T) = \phi_{ijk}(T)$. We can then write

$$\begin{aligned} P(x_{t-1} = i, x_t = j, x_{T-1} = \zeta, x_T = \gamma, y_{0\longrightarrow T}) &= P(y_T|x_T = \gamma) \\ \times P(x_T = \gamma|x_{T-1} = \zeta)P(x_{t-1} = i, x_t = j, x_T = \zeta, y_{0\longrightarrow T-1}) & \end{aligned} \tag{88}$$

where we used the product rule and the dependency conditions. Dividing both sides by

$P(y_{0\longrightarrow T-1})$ and summing over ζ we get

$$\begin{aligned} &P(x_{t-1}=i, x_t=j, x_T=\gamma|y_{0\longrightarrow T}) \\ &=\sum_{\zeta}\eta_{\zeta\gamma}(y_T)\cdot P(x_{t-1}=i, x_t=j, x_{T-1}=\zeta|y_{0\longrightarrow T-1}) \end{aligned} \tag{89}$$

with

$$\eta_{\zeta\gamma}(y_T)\equiv\frac{P(y_T|x_T=\gamma)P(x_T=\gamma|x_{T-1}=\zeta)}{P(y_T|y_{0\longrightarrow T-1})} \tag{90}$$

Equation 90 inserted into equation 87 provides the following recurrence relation for the ϕ^{γ}_{ijk}

$$\begin{aligned} \phi^{\gamma}_{ijk}(T)=\frac{1}{T}\cdot\delta(y_T-k)\cdot\eta_{ij}(y_T)\cdot P(x_{T-1}=j|y_{0\longrightarrow T-1})+\frac{1}{T}\sum_{t=1}^{T-1}\delta(y_t-k) \\ \cdot\sum_{\zeta}\eta_{\zeta\gamma}(y_T)\cdot P(x_{t-1}=i, x_t=j, x_{T-1}=\zeta|y_{0\longrightarrow T-1}) \end{aligned} \tag{91}$$

by changing the order of summation we can write the second term on the right-hand side of the equation as

$$\begin{aligned} &\frac{1}{T}\sum_{t=1}^{T-1}\delta(y_t-k)\cdot\sum_{\zeta=1}\eta_{ij}(y_T)\cdot P(x_{t-1}=i, x_{T-1}=\zeta|y_{0\longrightarrow T-1}) \\ &=\left(1-\frac{1}{T}\right)\sum_{\zeta}\eta_{\zeta\gamma}(y_T)\cdot\phi^{\zeta}_{ijk}(T-1) \end{aligned} \tag{92}$$

Finally by inserting equation 92 into equation 91 and changing terms order we obtain

$$\begin{aligned} \phi^{\gamma}_{ijk}(T)=&\sum_{\zeta}\eta_{\zeta\gamma}(y_T) \\ &\times\phi^{\zeta}_{ijk}(T-1)+\frac{1}{T}[\delta(y_T-k)\cdot g_{ij}(\zeta,\gamma)\cdot\omega_{\zeta}(T-1)-\phi^{\zeta}_{ijk}(T-1)] \end{aligned} \tag{93}$$

with $g_{ij}(\zeta,\gamma) \equiv \delta(i-\zeta)\cdot\delta(j-\gamma)$, and $\omega_\zeta(T-1) \equiv P(x_{T-1} = \zeta|y_{0\longrightarrow T-1})$ which can be computed recursively, and $\eta_{\zeta\gamma}(y_T)$ is expressed in terms of the model's parameters as

$$\eta_{\zeta\gamma}(y_T) = \frac{b_{\zeta\gamma}c_{\gamma,y_T}}{\sum_{m,k} b_{m,k}c_{k,y_T}\omega_m(T-1)} \tag{94}$$

with c_{γ,y_T} is the probability of emitting an output y_T in state γ , that is $c_{\gamma,y_T} \equiv \sum_k c_{\gamma k}\cdot\delta(y_T-k)$

4.4 Inverted Beta-Liouville Mixture Model

We suppose a D-dimension vector $\vec{X} = (X_1,\ldots,X_D)$ is drawn from an inverted Beta-Liouville distribution [94], then we have

$$\begin{aligned} p(\vec{X}|\alpha_d,\ldots,\alpha_d,\alpha,\beta,\lambda) = {} & \frac{\Gamma\left(\sum_{d=1}^D \alpha_d\right)\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\prod_{d=1}^D \frac{X_d^{\alpha_l-1}}{\Gamma(\alpha_d)} \\ & \times \lambda^\beta\left(\sum_{d=1}^D X_d\right)^{\alpha-\sum_{d=1}^D\alpha_d}\left(\lambda+\sum_{d=1}^D X_d\right)^{-(\alpha+\beta)} \end{aligned} \tag{95}$$

where $X_d > 0$ for $d = 1,\ldots,D$, $\alpha > 0, \beta > 0$ and $\lambda > 0$. In fact, the IBL distribution can be viewed as a generalized form of the inverted Dirichlet distribution that involves multiple symmetric and asymmetric modes. The mean, variance and covariance of the IBL distribution are given by:

$$E(X_d) = \frac{\lambda\alpha}{\beta-1}\frac{\alpha_d}{\sum_{d=1}^D\alpha_d} \tag{96}$$

$$Var(X_d) = \frac{\lambda^2\alpha(\alpha+1)}{(\beta-1)(\beta-2)}\frac{\alpha_d(\alpha+1)}{\sum_{d=1}^{D}\alpha_d(\sum_{d=1}^{D}\alpha_d+1)} - \frac{\lambda^2\alpha^2}{(\beta-1)^2}\frac{{\alpha_d}^4}{(\sum_{d=1}^{D}\alpha_d)^4} \tag{97}$$

$$Cov(X_m, X_n) = \frac{\alpha_m\alpha_n}{\sum_{d=1}^{D}}\left[\frac{\lambda^2\alpha(\alpha+1)}{(\beta-1)(\beta-2)(\sum_{d=1}^{D}\alpha_d+1)} - \frac{\lambda^2\alpha^2}{(\beta-1)^2(\sum_{d=1}^{D}\alpha_d)}\right] \tag{98}$$

If a set of data contains N vectors: $\mathcal{X} = \{\vec{X_1}, \ldots, \vec{X_N}\}$, where each $\vec{X_i} = (X_{i1}, \ldots, X_{iD})$ is drawn from the IBL mixture model with M components and is defined as follows

$$p(\vec{X_i}|\vec{\pi}, \Theta) = \sum_{j=1}^{M}\pi_j p(\vec{X_i}|\theta_j) \tag{99}$$

where $\Theta = (\theta_1, \ldots, \theta_M)$, $p(\vec{X_i}|\theta)$ denotes the IBL distribution in Eq.(95) associated with the jth component with parameters $\theta_j = (\alpha_{j1}, \ldots, \alpha_{jD}, \alpha_j, \beta_j, \lambda_j)$, and $\vec{\pi} = (\pi_1, \ldots, \pi_M)$ are the mixing coefficients where $0 \leq \pi_j \leq 1$ and $\sum_{j=1}^{M} = 1$.

Maximum likelihood estimation

In order to learn the models' parameters, we choose a learning approach based on Maximum Likelihood (ML). The values of different parameters are obtained by maximizing the log-likelihood function such as:

$$\widetilde{\Theta} = \underset{\Theta}{\operatorname{argmax}} \log p(\mathcal{X}|\vec{\pi}, \Theta) \tag{100}$$

where the log-likelihood function is given by

$$\begin{aligned}\mathcal{L}(\mathcal{X}|\vec{\pi},\Theta) &= \log p(\mathcal{X}|\vec{\pi},\Theta) = \log \prod_{i=1}^{N} p(\vec{X}_i|\vec{\pi},\Theta) \\ &= \sum_{i=1}^{N} \log \left(\sum_{j=1}^{M} \pi_j p(\vec{X}_i|\theta_j) \right)\end{aligned} \tag{101}$$

We define latent variables as indicators for a set of observed data. Let $\mathcal{Z} = \{\vec{Z}_1, \ldots, \vec{Z}_N\}$, each $\vec{Z}_i = (Z_{i1}, \ldots, Z_{iM})$ corresponds to an observed data vector $\vec{X}_i$, where $Z_{ij} \in \{0, 1\}$ and $\sum_{j=1}^{M} Z_{ij} = 1$, and $Z_{ij} = 1$ if $\vec{X}_i$ belongs to component j, and 0 otherwise. The log-likelihood of the complete data can thus be expressed as follows:

$$\Phi(\mathcal{X}, \mathcal{Z}|\vec{\pi},\Theta) = \sum_{i=1}^{N} \sum_{j=1}^{M} Z_{ij} \big\{ \log \pi_j + \log p(\vec{X}_i|\theta_j) \big\} \tag{102}$$

Next, the conditional expectation of the complete data log-likelihood is maximized in the M-step of the EM algorithm which is given by

$$\Omega(\mathcal{X}|\Theta) = \sum_{i=1}^{N} \sum_{j=1}^{M} \langle Z_{ij} \rangle \left\{ \log \pi_j + \log p(\vec{X}_i|\theta_j) \right\} \tag{103}$$

with the posterior probability $\langle Z_{ij} \rangle$ being the expected value of the indicator variable and is given by

$$\langle Z_{ij} \rangle = \frac{\pi_j p(\vec{X}_i|\theta_j)}{\sum_{k=1}^{M} \pi_k p(\vec{X}_i|\theta_k)} \tag{104}$$

We maximize the conditional expectation of the complete-data log-likelihood by computing the first derivatives with respect to all parameters

$$\begin{aligned}\frac{\partial\Omega(\mathcal{X}|\Theta)}{\partial\alpha_j} &= \sum_{i=1}^{N}\langle Z_{ij}\rangle\left[\log\sum_{d=1}^{D}X_{id} - \log(\lambda_j + \sum_{d=1}^{D}X_{id})\right] \\ &+ [\Psi(\alpha_j+\beta_j) - \Psi(\alpha_j)]\sum_{i=1}^{N}\langle Z_{ij}\rangle\end{aligned} \tag{105}$$

$$\begin{aligned}\frac{\partial\Omega(\mathcal{X}|\Theta)}{\partial\beta_j} &= \sum_{i=1}^{N}\langle Z_{ij}\rangle\left[\log\lambda_j - \log(\lambda_j + \sum_{d=1}^{D}X_{id})\right] \\ &+ [\Psi(\alpha_j+\beta_j) - \Psi(\beta_j)]\sum_{i=1}^{N}\langle Z_{ij}\rangle\end{aligned} \tag{106}$$

$$\begin{aligned}\frac{\partial\Omega(\mathcal{X}|\Theta)}{\partial\alpha_{jd}} &= \sum_{i=1}^{N}\langle Z_{ij}\rangle\left[\log X_{id} - \log\sum_{d=1}^{D}X_{id}\right] \\ &+ [\Psi(\sum_{d=1}^{D}\alpha_{jd} - \Psi(\alpha_{jd})]\sum_{i=1}^{N}\langle Z_{ij}\rangle\end{aligned} \tag{107}$$

$$\frac{\partial\Omega(\mathcal{X}|\Theta)}{\partial\lambda_j} = \sum_{i=1}^{N}\langle Z_{ij}\rangle\left[\frac{\beta_j}{\lambda_j} - \frac{\alpha_j+\beta_j}{\lambda_j + \sum_{d=1}^{D}X_{id}}\right] \tag{108}$$

with $\Psi(.)$ being the digamma function. It is obvious that a closed-form solution for θ_j does not exist.Thus, to estimate these parameters, we use the Newton-Raphson method [186] such as

$$\theta_j^{(t+1)} = \theta_j^{(t)} - H(\theta_j^{(t)})^{-1}\frac{\partial\Omega(\mathcal{X}|\vec{\pi}^{(t)},\Theta^{(t)})}{\partial\theta_j^{(t)}} \tag{109}$$

where $H(\theta_j^{(t)})^{-1}$ represents the inverse Hessian matrix for parameter θ_j and is described in detail in [127].

4.4.1 Online update for the sufficient statistics and model parameters

To set up an online EM, we start with an initial guess for the model parameters $\hat{\theta}(0)$, the initial state probabilities, $\omega_\zeta \equiv P(x_0 = \zeta)$, and the sufficient statistics, $\hat{\phi}^{\gamma}_{ijk}(0)$. After removing the contribution of the sufficient statistics such as performed in [173], state estimates, $\omega_\zeta(T)$ which represent the probability of being in the state ζ at time T are then expressed such as

$$\hat{\omega}_\zeta(T) = \sum_m \eta_{m\zeta}(y_T; \hat{\theta}(T-1)) \cdot \hat{\omega}_m(T-1) \tag{110}$$

Finally, the parameters are re-estimated according to the following equations

$$\hat{b}_{ij}(T) = \frac{\sum_k \sum_\zeta \phi^{\zeta}_{ijk}(T)}{\sum_{j,k} \sum_\zeta \phi^{\zeta}_{ijk}(T)} \tag{111}$$

$$\hat{c}_{jk}(T) = \frac{\sum_i \sum_\zeta \phi^{\zeta}_{ijk}(T)}{\sum_{i,k} \sum_\zeta \phi^{\zeta}_{ijk}(T)} \tag{112}$$

4.5 Experiments and results

In this section, extensive experiments are conducted and we have implemented several real-world topical yet challenging applications using the online HMM with IBL emission probabilities. We are mainly comparing our new approach to its classical online Gaussian-based HMMs competitors and other new adaptations that we executed for the sake of comparison and testing, e.g., inverted Dirichlet-based online HMM (Online ID-HMM) and Dirichlet-based online HMM (Online Dir-HMM). It is noteworthy that the learning of the mentioned adaptations has been based on the same methodology described in the previous section to learn the IBL mixture-based HMM. Real-world applications on two video data sets, an anomaly in a crowd context and direction-related anomaly detection in an airport, are tested

to validate the performance of our model.

Recognition of human action in videos gained a great deal of attention thanks to the multitude of applications in many domains such as human-computer interfaces, video surveillance [254, 208] and activity biometry [70]. Applications involve but are not limited to, detecting violence, hostile behavior, and sexual harassment [222], not to mention life-threatening events such as pedestrians accidents, criminality [226].

It is worthwhile to mention that dealing with crowded scenes analysis often involves a sizable amount of individuals acquiring irregular directions in an exceedingly vast region hence the complexity of the task. Anomalies or abnormal events can be intuitively defined as any occurrence of a deviation from the conventional crowd behavior in an exceedingly vast video. Moreover, an anomaly could eventually be a pattern that doesn't follow expected traditional behavior in a given context.

Hidden Markov Models are indeed an appropriate tool to tackle this problem since they are particularly suitable when working with dynamic data such as videos, and attempting to unveil unknown natures of anomalies.

4.5.1 Anomaly detection in a crowd of pedestrians

The main goal of this experiment is to detect any anomalies in the surveillance video of the publicly available UCSD Ped1 and Ped2 data set [133]. Both data sets are formed from video sequences of pedestrians on a walkway and divided into a training set, with normal frames only, and a testing set composed of both normal and abnormal frames. These two data sets only differ in the camera viewpoint from which footage has been captured. We still are able to benefit from ground truth, provided for all test sequences. Sample frames from the training set with different crowd densities and anomalies are presented in figures 22 and 23. In the following we proceed to the feature extraction in a procedure we describe briefly (see [79] for further details).

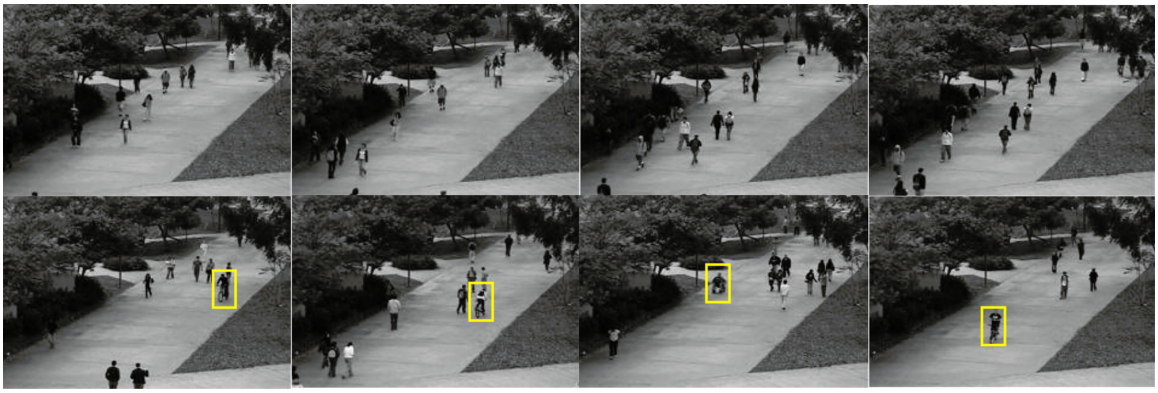


Figure 22: Frames from the Ped1 normal (upper row) and abnormal activities (bottom row) with anomalies highlighted

Figure 23: Frames from the Ped2 normal (upper row) and abnormal activities (bottom row) with anomalies highlighted

The pre-processing involves a gray-level re-sampling of the frames to a size of 160×240 pixels, with a filter-based Gaussian noise reduction where the filter size is $[3, 3]$ and $\sigma = 1.1$. Next, we perform some dimensional sampling steps in order to meet HMMs use requirements such as lowering the length of our histograms (12 in this application), small volumes extracted from sequences, called *cuboids*, are repeated several times in a sequence in order to avoid model overfitting. The dimensional sampling adopted here consists in dividing sequences into cuboids each of them subdivided into 8 subregions, 2

along each direction. Pixels' contribution within a subregion is weighted by its magnitude and is computed in the same fashion as in [79]. We model each cuboid by a series of 8 normalized histograms through which a dynamic mechanism embedded in each cuboid is illustrated. An HMM is trained for each cuboid location taking into account all the available observations.

We set a threshold to compare each computed likelihood from the testing videos in order to fulfill the classification task. This threshold is tied to the location of cuboids and is set using the minimum likelihood value of training samples at each location multiplied by a factor k chosen depending on the frequency of anomalous sequences and can either be $k = 1, k < 1$ or $k > 1$ [79].

Eventually, when dealing with applications such as anomaly detection, we wish, as far as practicable, to achieve the optimal Equal Error Rate (EER). However, the latter is not the only point of performance on which we should rely when assessing our results. The overall performance can thus be studied by computing the Area Under the Curve (AUC).

We choose to set our model to a number of states $K = 2$ and a number of mixtures per state $M = 3$. It is better to keep those two values low as they drastically contribute to the simplicity of computing. We also carry offline and online trials for the sake of comparison. Results will be detailed later in this section. The number of states K and mixture components M is set using K-means [120] clustering of the training data, with the number of clusters varying from 2 to 20.

We train each HMM with a set of training features for each of the classes 10 times. Then we keep track of the scored results as an average across the training times. Results and comparison with different used models in the same experimental context can be observed in table 10.

The results show an apparent improvement each time we chose to integrate the online EM into the HMM framework. This is related to the gradual adjustment of the parameters

that allow for better fitting of the data by the proposed model. Nonetheless, it is noteworthy to mention that Online IBL-HMM performed significantly better than its offline peer, plus even better than the inverted Dirichlet-based HMM and the generalized inverted Dirichlet-based HMM as well. The online setup combined with an appropriate choice of distribution contributed to this decent amelioration.

Method	**Ped1**	**Ped2**
GMM-HMM	72.03	73.19
ID-HMM	75.28	77.51
GID-HMM	89.99	87.27
IBL-HMM	**90.09**	**90.41**
Online GMM-HMM	88.60	84.53
Online ID-HMM	91.13	91.72
Online GID-HMM	89.03	84.33
Online IBL-HMM	**95.10**	**92.69**

Table 10: Average recognition rates for different used HMMs in the context of video anomaly detection UCSD, ped1 and ped2 datasets

4.5.2 Anomaly detection: Airport security line-up

This application permits identifying people going in the wrong direction in an airport security line-up. The videos are treated as sequences extracted from the anomalous Behavior data set [253]. The latter has been gathered from a surveillance camera hung up to the ceiling and filming vertically downwards. One part of the data set is clear from any anomalies and hence used for the training step, while the other is used for testing purposes. Figure 24 shows some frames from the data set.

Anomalies displayed in this data set are of a larger scale compared to the previous

Figure 24: Frames from Anomalous Behavior airport wrong direction with highlighted anomalies

application, we then choose to increase the cuboid size to prevent as many false positive cuboids. Here we choose 80×80 pixels. We use AUC-ROC curve [43] as a performance assessment measure. What's interesting, is that in this binary classification context, a model has to predict whether the frame is an anomaly or not. The AUC curve measures the models' performance depending on various thresholds. The highest AUC score will help us determine the best model. The AUC-ROC curve is plotted with True Positive Rate (TPR) and False Positive Rate (FPR). We thought it would also be interesting to allow some interest in evaluating the Equal Error Rate (EER) as a performance assessment. EER is an optimized value where a false-positive intersects with a false negative. The better a model is, the lower its EER score. Results are displayed in figure 25.

$$TPR = \frac{TruePositive}{TruePositive + FalseNegative} \tag{113}$$

$$FPR = \frac{FalsePositive}{TrueNegative + FalsePositive} \tag{114}$$

Performances of different tested methods displayed in table 11, show the significant role played by the online learning method in improving the detection performance

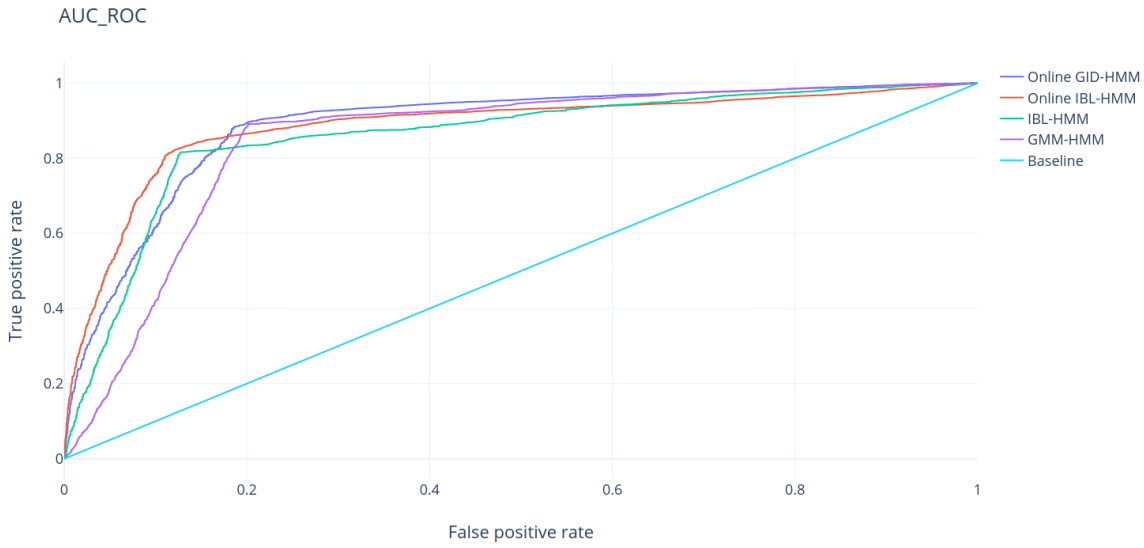


Figure 25: AUC-ROC curve comparison of the proposed Online IBL-HMM with other methods for Anomalous Behavior dataset

of anomalous events.

Method	Online	Offline
GMM-HMM	86.13	79.02
ID-HMM	89.64	80.11
GID-HMM	91.17	86.98
IBL-HMM	**94.83**	**92.06**

Table 11: Average recognition rates for different used HMMs in the context of video anomaly detection Anomalous Behavior data set both online and offline

4.5.3 Abnormal Crowd Behavior: Escape scene

This experiment aims to capture abnormal crowd behavior in three different scenes in the video sequence of unusual crowd events captured synthetically by the University of Minnesota (UMN) [188]. The data set is composed of videos of 11 different scenarios of an escape event in 3 different indoor and outdoor scenes: Lawn, Indoor and Plaza. Each video

is composed of an initial part of normal behavior followed by sequences of abnormal behavior where people run from the center of the scene to simulate an escaping event. All footage is recorded at a frame rate of 30 frames per second at a resolution of 640 $\times$ 480 using a static camera. Figure 26 shows sample frames of these scenes. Here, the process for identifying the likely patterns is performed in a similar way as in [171], where we use the bag of words [135] method to identify the events and normal videos for training LDA [32]. For computational simplicity, the resolution of the particle grid is kept at 25% of the number of pixels. We partition our frames into blocs of C clips. Then, from each clip C_j, W visual words are extracted. We randomly pick visual words of size $5 \times 5 \times 10$ and code a book of size S using K-means clustering. In this case, we extract $W = 30$ visual words from a block of 10 frames. Thus a final codebook contains $C = 10$ clips. To evaluate our model, 50 different frames of each scene are selected.

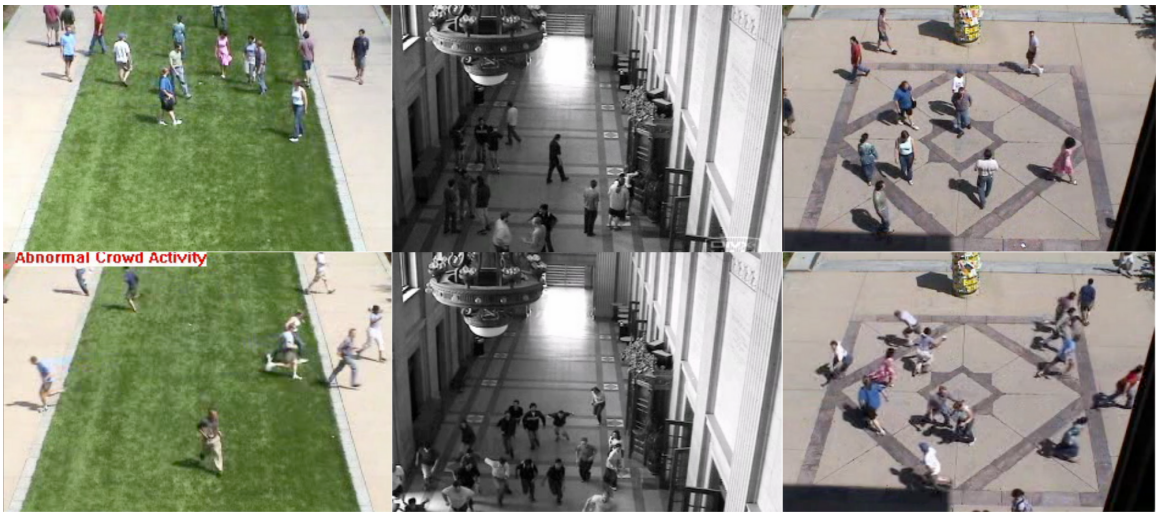


Figure 26: Frames from the UMN data set with normal (upper row) and abnormal escape scenes (bottom row) from three different indoor and outdoor scenes

Table 12 shows the average accuracy comparison of several tested methods namely

online-based and offline-based HMMs implemented for the sake of this particular comparison. We specifically want to focus on the role played by online HMMs compared to offline models but in detecting escape scenes, we also want to focus on the role played by the IBL as a distribution to improve the average recognition accuracy of anomalous scenes. Overall, the proposed method achieves the best accuracy with an average of 89.12%, which is higher than the average accuracy of 83.53% where we did not use the online-based model. We also observe that both online and offline IBL-HMM perform better compared to other methods.

Method	**Online**	**Offline**
GMM-HMM	71.13	69.80
ID-HMM	76.08	73.42
GID-HMM	83.40	78.55
IBL-HMM	**89.12**	**83.53**

Table 12: Average recognition rates for different used HMMs in the context of a crowd escape scene detection on the UMN data set, both online and offline

For further performance evaluation, we have presented the ROC curves in figure 27 for the different used models and can thus observe that our method achieves a better ratio and the number of false positives is significantly lower.

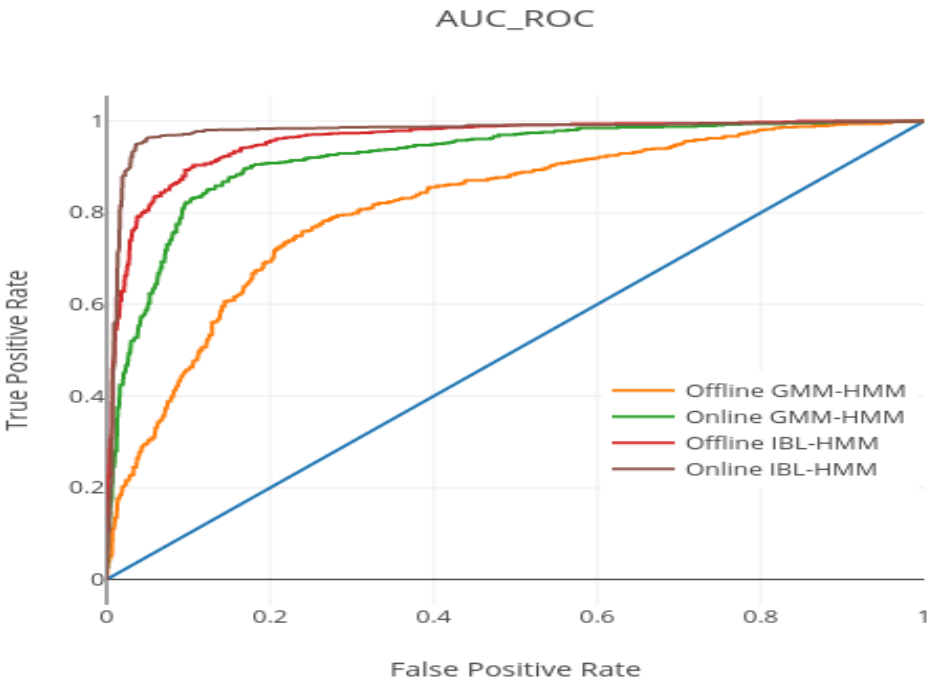


Figure 27: AUC-ROC curve for each of the tested models on the UMN dataset

One of the main takeaways is the crucial role that online learning plays in reducing false positive detection of anomalous behavior especially in binary contexts where only two scenarios such as "normal" or "escape scene" are possible. Clearly, in the mentioned situations we aim for the least false positive detection rate possible to avoid false alerts and thus reduce unnecessary alarming situations.

4.6 Conclusion

There are a multitude of techniques that researchers are adopting to address the challenge of abundant and massive data modeling. Online learning methods are one of the most powerful tools to handle big streams of data such as videos in a real-time context. Using

HMMs is also a suitable way to deal with dynamic data such as videos, but the biggest challenge remains in finding the most powerful distribution to faithfully model specific types of data such as positive vectors. Further, the interest in adopting IBL mixtures for modeling our data arose from the limitations encountered when other distributions such as Gaussian mixtures and inverted Dirichlet were adopted. In fact, IBL mixtures provided a smaller number of parameters compared to the generalized inverted Dirichlet, not to mention that it showed its effectiveness when dealing with positive vector modeling in contrast to the rest of the tested distributions. In this paper, we proposed a model in which all the aforementioned problems are addressed simultaneously in the case of human activities modeling and anomalies detection. The developed approach applies online learning of parameters within the HMM framework. Experimental results involving challenging real-life applications such as anomaly detection in a human crowd context showed that the proposed approach is highly promising. We have demonstrated that the proposed method is highly effective at discriminating between scenes of normal and abnormal behavior, and that our approach operates in real-time. Future works are intended to be done in the near future extending this work to different flexible distributions and considering a hybrid Generative-Discriminative model using Support Vector Machines kernels to improve classification capabilities and to further reduce error rates.

Chapter 5

Hybrid Generative Discriminative Approach with Hidden Markov Models and Support Vector Machines

5.1 Indoor Activity Recognition Using a Hybrid Generative-Discriminative Approach with Hidden Markov Models and Support Vector Machines

Human activity recognition is used for many practical applications such as context modeling in smart cities, surveillance and assisted living. In this chapter, we apply a hybrid generative-discriminative approach using Fisher kernels with inverted Dirichlet-based and inverted Beta-Liouville-based hidden Markov models (HMMs) to improve the recognition performance. We propose a method that combines HMMs as a generative approach, with the discriminative approach of Support Vector Machine (SVM). This strategy allows us to deal with Spatio-temporal motion data, and at the same time use the special focus on

the classification task that SVM could provide us. Experiments on the challenging activity recognition benchmark UCF101, demonstrate an effective improvement of the recognition performance compared to the standard generative and Gaussian-based HMM approaches.

5.1.1 Introduction

Human activity recognition has recently gained prominent interest as an active area of computer vision research. Indeed, being able to accurately identify different performed activities allows for further understanding of people's lifestyle and their needs over a period of time. Ranging from assisted living [61, 56], elderly fall detection [252], to smart homes technologies [205], applications of activity recognition (AR) are numerous and equally critical.

AR in computer vision focuses mainly on extracting information from pre-segmented video sequences, which is not necessarily the case in real-life scenarios where videos are not segmented beforehand. In realistic scenarios, the challenge lies in localizing an action of potential interest in time in a video. While most recent works focused on powerful pre-processing techniques to extract motion-related features within images and videos [96, 76, 62, 27], there has also been a rising need to understand more about the context of these multimedia and invest in reliable model-based techniques [129, 177, 92]. In fact, a better performance suggests that more global spatial and temporal information could be necessary for activity detection. Therefore, HMM has become one of the most widely used models thanks to its maturity and high efficiency in handling spatio-temporal aspects of human behaviour while moving or performing certain actions [9, 227].

Two main approaches are used in machine learning to perform recognition tasks: generative techniques that model the underlying distributions of classes, and discriminative techniques that give a sole focus on learning the class boundaries [211]. Both techniques have been widely used in computer vision to effectively recognize action patterns [178].

In a broader sense, generative models, such as HMM, are considered to be one of the machine learning methods that aspire to empower machines with the fundamental capacity to examine objects, events or observations and speculate on the upcoming aftermath. Therefore, they tend to require less training data than discriminative models. In particular, if the task to be performed is classification, SVM can clearly distinguish the differences between categories and can thus outperform generative models especially if a large number of training examples are available. SVM is extensively used due to their great capacity to generalize, often resulting in better performance than traditional classification techniques [130]. They have proven that they can yield good results in several recognition tasks [247, 38, 7].

As a discriminative approach, the main process of SVM is to find surfaces that better separate the different data classes. The main idea is to use a kernel that allows efficient discrimination in non-linearly separable input feature spaces. One of the key best practices when using SVM is to adopt the convenient kernel function which has to be suitable for the classified data and the objective task. Conventional kernels include linear, polynomial and radial basis function kernels[25]. Applying these kernels is not always possible, especially when it comes to classifying objects represented by sequences of different lengths [239]. Therefore, the mentioned kernels may not be a wise choice to model our action data. Consequently, a hybrid generative-discriminative method is adopted to allow the conversion of data into fixed-length and hence provide additional performance to the model. In this work, we propose the use of Fisher Kernels (FK) generated with inverted Dirichlet-based and inverted Beta-Liouville-based HMMs (IDHMM and IBLHMM respectively) to model the temporal variations.

The main contributions of this paper are the following: First, we apply for the first time two non-Gaussian HMMs, i.e. inverted Dirichlet-based HMM and the inverted Beta-Liouville-based HMM on the challenging Human Action benchmark UCF101 by the University of Central Florida. Second, we derive a hybrid generative-discriminative approach for both

of the aforementioned HMMs with FK for SVM-based modeling of positive vectors. This novel approach is also tested on the UCF101 recorded activities, as an unprecedented attempt of using Fisher Vectors-based hybrid generative-discriminative models to handle this challenging dataset. The remainder of the paper is organized as follows, Section 2 discusses the proposed model. Section 3 presents the performed experiments and obtained results. We finally conclude the paper in section 4.

5.1.2 Hybrid Generative-Discriminative approach with Fisher Kernels

In this section, we present the proposed approach. To illustrate our model, we are first listing various HMM notations and enumerating the upcoming used work script. We then recall the main process behind the forward-backward algorithm. Lastly, we perform a complete derivation of the FK-based model.

5.1.2.1 Hidden Markov Models

We consider a HMM with continuous emissions and K hidden states. We put a set of hidden states $H = \{h_1, ..., h_T\}; h_j \in [1, K]$.

The transition probabilities matrix: $B = \{b_{ij} = P(h_t = j|h_{t-1} = j')\}$ and the emission probabilities matrix: $C = \{c_{ij} = P(m_t = i|h_t = j)\}; i \in [1, M]$ where M is the number of mixture components associated with state j. We define the initial probability: π_j which is the probability to start the observation sequence from the state j.

We denote an HMM as: $\Delta = \{B, C, \varphi, \pi\}$ where φ is the set of mixture parameters depending on the chosen type of mixture. In this work, we focus on inverted Dirichlet and inverted Beta-Liouville distributions. Let a D-dimensional positive vector $\vec{X} = (X_1, X_2, ..., X_D)$ follow an Inverted Dirichlet (ID) distribution, the joint function is given by Tiao & Cuttman

[233] as follows:

$$\mathcal{ID}(\vec{X}|\vec{\alpha}) = \frac{\Gamma(|\vec{\alpha}|)}{\prod_{d=1}^{D+1}\Gamma(\alpha_d)}\prod_{d=1}^{D}X_d^{\alpha_d-1}\left(1+\sum_{d=1}^{D}X_d\right)^{-|\vec{\alpha}|} \tag{115}$$

where $X_d > 0, d = 1, 2, ..., D, \vec{\alpha} = (\alpha_1, ..., \alpha_{D+1})$ is the vector of parameters and $|\vec{\alpha}| = \sum_{d=1}^{D+1}\alpha_d, \alpha_d > 0, d = 1, 2, ..., D+1$.

For the inverted Beta-Liouville, we suppose a D-dimension vector $\vec{X} = (X_1, \ldots, X_D)$ is drawn from an inverted Beta-Liouville distribution [94], then we have

$$\begin{aligned} p(\vec{X}|\alpha_d, \ldots, \alpha_d, \alpha, \beta, \lambda) = \\ \frac{\Gamma\left(\sum_{d=1}^{D}\alpha_d\right)\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\prod_{d=1}^{D}\frac{X_d^{\alpha_l-1}}{\Gamma\left(\alpha_d\right)} \\ \times\lambda^{\beta}\left(\sum_{d=1}^{D}X_d\right)^{\alpha-\sum_{d=1}^{D}\alpha_d}\left(\lambda+\sum_{d=1}^{D}X_d\right)^{-(\alpha+\beta)} \end{aligned} \tag{116}$$

where $X_d > 0$ for $d = 1, \ldots, D$, $\alpha > 0, \beta > 0$ and $\lambda > 0$. $\Gamma(x) = \int_0^{+\infty} t^{x-1}e^{-t}dt$ is the Gamma function. In fact, the IBL distribution can be viewed as a generalized form of the inverted Dirichlet distribution that involves multiple symmetric and asymmetric modes.

5.1.2.2 Inference on hidden states: Forward-Backward Algorithm

The forward algorithm computes the probability of being in state h_j up to time t for the partial observation sequence produced by the model Δ. We consider a forward variable $\gamma_t(i) = P(X_1, X_2, \ldots, X_t, , i_t = h_i|\Delta)$. There is a recursive relationship that is used to compute the former probability. We can resolve for $\gamma_t(i)$ recursively as follows:

1. Initialization:

$$\gamma_t(i) = \pi_i \varphi_i(X_1) \qquad 1 \leq i \leq K \tag{117}$$

2. Recursion:

$$\begin{gathered} \gamma_{t+1}(j) = \left[\sum_{i=1}^{K} \gamma_t(i) b_{ij} \right] \varphi_j(X_{t+1}) \\ for \quad 1 \leq t \leq T-1, \;\; 1 \leq j \leq K \end{gathered} \tag{118}$$

3. Termination:

$$P(X|\Delta) = \sum_{i=1}^{K} \gamma_T(i) \tag{119}$$

The backward variable, which is the probability of the partial observation sequence $X_{t+1}, X_{t+2}, \ldots, X_T$ given the current state is denoted by $\delta_t(i)$ and can similarly be determined as follows:

1. Initialization:

$$\delta_t(i) = 1, \; 1 \leq i \leq K \tag{120}$$

2. Recursion:

$$\begin{gathered} \delta_t(i) = \sum_{j=1}^{K} b_{ij} \varphi_j(X_{t+1} \delta_{t+1}(j)) \\ for \quad t = T-1, T-2, \ldots, 1 \quad 1 \leq i \leq K \end{gathered} \tag{121}$$

3. Termination:

$$\begin{aligned} P(X|\Delta) &= \sum_{i=1}^{K} \gamma_i(T) \\ &= \sum_{i=1}^{K} \pi_i b_i(X_1) \varphi_i(1) \\ &= \sum_{i=1}^{K} \sum_{j=1}^{K} \gamma_t(i) b_{ij} \varphi_j(X_{t+1}) \delta_{t+1}(j) \end{aligned} \tag{122}$$

5.1.2.3 Fisher Kernels

Non-linear SVM serves our discrimination needs in the context of realistic recognition tasks. The strategy is to use a Kernel method to avoid calculation cost and memory consumption problems that might arise from performing inner product calculation of high-dimensional feature vectors. It will allow us to implicitly project objects to high-dimensional space by using a kernel function $\kappa(x_\zeta, x_\eta) = \langle \phi(x_\zeta), \phi(x_\eta) \rangle$ and solving the problem with observations x_ζ and x_η represented as Bag of Features (BoF) or Bag of Visual Words (BoVW) [119, 249] in general with ϕ being a projection function and $\langle \cdot, \cdot \rangle$ meaning the inner product. Here we choose Fisher Kernel as the kernel function. This choice is motivated by FK being a general way of fusing generative and discriminative approaches for classification. FK is formulated as

$$FK(X_\zeta, X_\eta) = \langle FS(X_\zeta, \Delta), FS(X_\eta, \Delta) \rangle \tag{123}$$

where X_ζ and X_η are two observations, Δ is the parameters set of a generative model defined by $P(X|\Delta)$ and $FS(X_\zeta, \Delta)$ is the Fisher score.

$$FS(X, \Delta) = \nabla_\Delta \log P(X|\Delta) \tag{124}$$

Given a particular HMM:

$$\begin{aligned} L(X|\Delta) &= \log P(X|\Delta) \\ &= \log \sum_{i=1}^{K} \gamma_T(i) \\ &= \log \sum_{i=1}^{K} \pi_i \varphi_i(X_1) \delta_1(i) \end{aligned} \tag{125}$$

The derivatives for the inverted Dirichlet-based HMM can be defined as follows:

$$\nabla_{\Delta} L(X|\Delta) = \left[\frac{\partial L(X|\Delta)}{\partial \pi_i}, \frac{\partial L(X|\Delta)}{\partial b_{ij}}, \frac{\partial L(X|\Delta)}{\partial \alpha_{id}}, \frac{\partial L(X|\Delta)}{\partial \alpha_i}\right] \tag{126}$$

Also, the derivatives for the inverted Beta-Liouville-based HMM can be defined as follows:

$$\nabla_{\Delta} L(X|\Delta) = \left[\frac{\partial L(X|\Delta)}{\partial \pi_i}, \frac{\partial L(X|\Delta)}{\partial b_{ij}}, \frac{\partial L(X|\Delta)}{\partial \alpha_{id}}, \frac{\partial L(X|\Delta)}{\partial \alpha_i} \frac{\partial L(X|\Delta)}{\partial \beta_i}, \frac{\partial L(X|\Delta)}{\partial \lambda_i}\right] \tag{127}$$

Each derivative with respect to the parameter is calculated using Eq.(155) in the following manner. Common entities derivatives can be expressed as:

$$\frac{\partial L(X|\Delta)}{\partial \pi_i} = \frac{\varphi_i(X_1)\delta_1(i)}{\sum_{i=1}^{K} \pi_i \varphi_i(X_1)\delta_1(i)} \tag{128}$$

$$\begin{aligned} \frac{\partial L(X \mid \Delta)}{\partial b_{ij}} &= \frac{1}{P(X \mid \Delta)} \sum_{k=1}^{K} \frac{\partial \gamma_T(k)}{\partial b_{ij}} \\ &= \frac{1}{P(X \mid \Delta)} \sum_{k=1}^{K} \left(\frac{\partial}{\partial b_{ij}} \sum_{l=1}^{K} \gamma_{T-1}(l) b_{lk} \varphi_k\left(X_T\right) \right) \\ &= \frac{1}{P(X \mid \Delta)} \sum_{k=1}^{K} \sum_{l=1}^{K} \frac{\partial \gamma_{T-1}(l)}{\partial b_{ij}} b_{lk} \varphi_k\left(X_T\right) \\ &+ \partial \gamma_{T-1}(i) \varphi_{ij}\left(X_T\right) \end{aligned} \tag{129}$$

Inverted Dirichlet related derivatives can be expressed as follows:

$$\frac{\partial L(X \mid \Delta)}{\partial \alpha_{id}(ID)} = \frac{1}{P(X \mid \Delta)} \left(\sum_{j=1}^{K} \sum_{k=1}^{K} \frac{\partial \gamma_{T-1}(k)}{\partial \alpha_{id}} b_{kj} \varphi_j\left(X_T\right) + \sum_{k=1}^{K} \partial \gamma_{T-1}(k) b_{ki} \frac{\partial \varphi_i\left(X_T\right)}{\partial \alpha_{id}} \right) \tag{130}$$

$$\frac{\partial L(X \mid \Delta)}{\partial \alpha_i(ID)} = \frac{1}{P(X \mid \Delta)} \left(\sum_{j=1}^{K} \sum_{k=1}^{K} \frac{\partial \gamma_{T-1}(k)}{\partial \alpha_i} b_{kj} \varphi_j \left(X_T \right) + \sum_{k=1}^{K} \partial \gamma_{T-1}(k) b_{ki} \frac{\partial \varphi_i \left(X_T \right)}{\partial \alpha_i} \right) \tag{131}$$

$$\frac{\partial \varphi_i \left(X_t \right)}{\partial \alpha_{id}(ID)} = \Psi \left(\sum_{d=1}^{D} \alpha_{id} \right) - \Psi \left(\alpha_{id} \right) + \log \left(X_d \right) - \log \left(1 + \sum_{d=1}^{D} X_d \right) \tag{132}$$

$$\frac{\partial \varphi_i \left(X_t \right)}{\partial \alpha_i(ID)} = \Psi \left(\sum_{d=1}^{D} \alpha_i \right) - \Psi \left(\alpha_i \right) - \log \left(1 + \sum_{d=1}^{D} X_d \right) \tag{133}$$

Figure 28: Frames from the used UCF101 subset with 10 different activities respectively from left to right: Mopping Floor(A1), Brushing Teeth(A2), Mixing Batter(A3), Writing On Board(A4), Shaving Beard(A5), Pizza Tossing(A6), Jump Rope(A7), Blow Dry Hair(A8), Blowing Candles(A9), and Pull ups (A10)

Inverted Beta-Liouville related derivatives can be expressed as follows:

$$\begin{aligned} \frac{\partial L(X \mid \Delta)}{\partial \alpha_{id}(IBL)} = & \frac{1}{P(X \mid \Delta)} \left(\sum_{j=1}^{K} \sum_{k=1}^{K} \frac{\partial \gamma_{T-1}(k)}{\partial \alpha_{id}} b_{kj} \varphi_j \left(X_T \right) \right. \\ & \left. + \sum_{k=1}^{K} \partial \gamma_{T-1}(k) b_{ki} \frac{\partial \varphi_i \left(X_T \right)}{\partial \alpha_{id}} \right) \end{aligned} \tag{134}$$

$$\frac{\partial L(X \mid \Delta)}{\partial \alpha_i \textit{(IBL)}} = \frac{1}{P(X \mid \Delta)} \left(\sum_{j=1}^{K} \sum_{k=1}^{K} \frac{\partial \gamma_{T-1}(k)}{\partial \alpha_i} b_{kj} \varphi_j \left(X_T \right) + \sum_{k=1}^{K} \partial \gamma_{T-1}(k) b_{ki} \frac{\partial \varphi_i \left(X_T \right)}{\partial \alpha_i} \right) \tag{135}$$

$$\frac{\partial L(X \mid \Delta)}{\partial \beta_i} = \frac{1}{P(X \mid \Delta)} \left(\sum_{j=1}^{K} \sum_{k=1}^{K} \frac{\partial \gamma_{T-1}(k)}{\partial \beta_i} b_{kj} \varphi_j \left(X_T \right) + \sum_{k=1}^{K} \partial \gamma_{T-1}(k) b_{ki} \frac{\partial \varphi_i \left(X_T \right)}{\partial \beta_i} \right) \tag{136}$$

$$\frac{\partial L(X \mid \Delta)}{\partial \lambda_i} = \frac{1}{P(X \mid \Delta)} \left(\sum_{j=1}^{K} \sum_{k=1}^{K} \frac{\partial \gamma_{T-1}(k)}{\partial \lambda_i} b_{kj} \varphi_j \left(X_T \right) + \sum_{k=1}^{K} \partial \gamma_{T-1}(k) b_{ki} \frac{\partial \varphi_i \left(X_T \right)}{\partial \lambda_i} \right) \tag{137}$$

$$\frac{\partial \varphi_i \left(X_t \right)}{\partial \alpha_i \textit{(IBL)}} = \Psi(\alpha_i + \beta_i) - \Psi(\alpha_i) + \log \sum_{d=1}^{D} X_d - \log(\lambda_i + \sum_{d=1}^{D} X_d) \tag{138}$$

$$\frac{\partial \varphi_i \left(X_t \right)}{\partial \alpha_{id} \textit{(IBL)}} = \Psi\left(\sum_{d=1}^{D} \alpha_{id} \right) - \Psi(\alpha_{id}) + \log X_d - \log \sum_{d=1}^{D} X_d \tag{139}$$

$$\frac{\partial \varphi_i \left(X_t \right)}{\partial \beta_i \textit{(IBL)}} = \Psi(\alpha_i + \beta_i) - \Psi(\beta_i) + \log \lambda_i - \log(\lambda_i + \sum_{d=1}^{D} X_d) \tag{140}$$

$$\frac{\partial \varphi_i \left(X_t \right)}{\partial \lambda_i \textit{(IBL)}} = \frac{\beta_i}{\lambda_i} - \frac{\alpha_i + \beta_i}{\lambda_i + \sum_{d=1}^{D} X_d} \tag{141}$$

with $\Psi(.)$ being the digamma function.

5.1.3 Experiments

We choose to test our model on the notorious UCF101 dataset[1][228]. The task is to recognize different actions of daily life (ADLs) in an indoor setting.

UCF101 is an action recognition dataset of realistic action videos, collected from YouTube, having 101 action categories. This data set is an extension of the UCF50 dataset which has 50 action categories. UCF101 is the largest and one of the most challenging action datasets in terms of complexity and scale, with a broad variety of actions with large variations in camera motion and cluttered backgrounds. It consists of 13,320 videos of a maximum of 150 frames per video with the resolution of 320×240, for 101 human actions divided into five types: Human-Human Interaction, Playing Musical Instruments, Body-Motion Only and Sports. In this work, we choose to evaluate the proposed method on a chosen subset of indoor activities.

The used subset contains 10 different activities: Mopping Floor (A1), Brushing Teeth (A2), Mixing Batter (A3), Writing On Board (A4), Shaving Beard(A5), Pizza Tossing (A6), Jump Rope (A7), Blow Dry Hair (A8), Blowing Candles (A9), and Pull-ups (A10). Subjects filmed in the dataset have different appearances, genders and ethnicity. Video clips are 1-71s long on average. Sample frames from the adopted activities are displayed in figure 28.

The frame size is 320×240 and the frame rate is 25 frames/s. We extract cuboids from each sequence by considering 100-150 frames of the actions and fixing the size of the bounding box to 200×200. We first extract features on a frame basis (at a rate of one frame/second) considering a combination of both low-level and high-level cues. For low-level cues, we perform quantization of the motion vectors into 8 different directions. For high-level cues we apply an enhanced pose estimator as in [210], however, we only detect the location of 9 body parts. We then construct a descriptor for each body part, modeled by

[1]The UCF101 dataset is publicly available at: https://www.crcv.ucf.edu/data/UCF101.php

an 8 bin histogram with optical flows assigned to the corresponding body part. As a final step, we concatenate all histograms to create a 72 bin histogram for each frame. We use the extracted features as training and testing data.

For each activity, a separate HMM is trained using the extracted features. In the testing step, we calculate the likelihood of each testing video sequence and class labels are assigned according to the maximum likelihood calculated. The main goal here is to use the generated outcome to train our SVM for an enhanced recognition rate. In the performed experiments, we set the number of states $K = 2$ for all trained HMMs with mixture components $M = 2$ for both used distributions (ID and IBL). In total, four model combinations are tested for comparison ends ID-HMM, Hybrid ID-HMM (HyID-HMM), IBLHMM and Hybrid IBL-HMM (HyIBL-HMM).

The obtained results displayed in table 13 show that both hybrid IDHMM and IBLHMM-based methods demonstrate relatively similar performance, nonetheless, the difference between them and the generative approach is conspicuous. We notice indeed that the IBLHMM outperforms its IDHMM analogue in the generative approach.

Table 13: Average recognition accuracies for different used activity recognition models

	Generative approach	Hybrid SVM-HMM
ID-based HMM	84.99%	**91.72%**
IBL-based HMM	89.04%	**96.91%**

This increasing recognition capacity was expected and is once more validated when it comes to positive vector modeling. Using IBL as an emission probability distribution, improved the modeling accuracy considering that it contains the inverted Dirichlet distribution as a special case, and hence generates additional flexibility. Furthermore, results achieved by applying the hybrid generative-discriminative approach demonstrate the striking increase in terms of the modeling accuracy and further validate the improved performance that SVM provided to the generative technique. Figures 29 and 30 present the

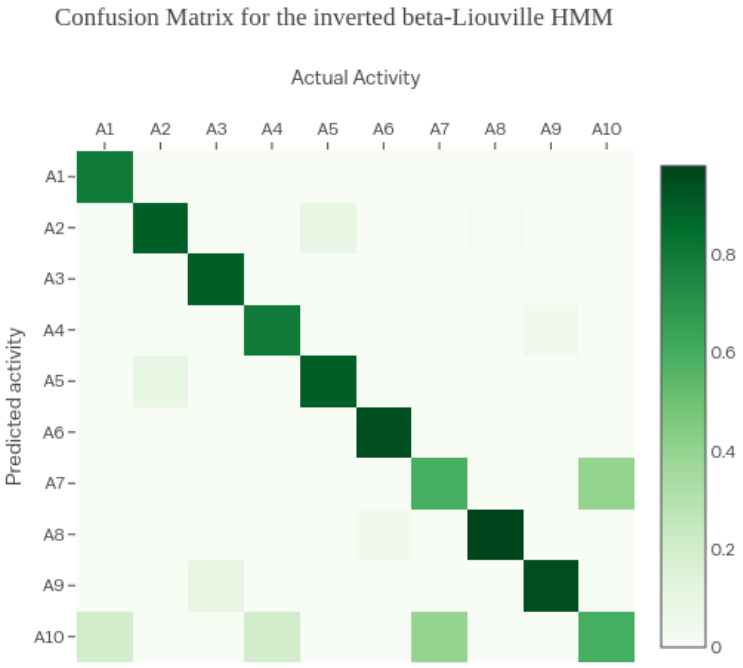


Figure 29: Confusion matrix for the inverted Beta-Liouville HMM with UCF101 subset

confusion matrices of the tested dataset. They describe recognition rates of SVM-HMM and generative HMMs for each activity. There are some similarities in the appearance of Jump Rope and Pull-ups. This results in a lower recognition rate for the generative approach which emphasizes the similarity of intra-class, whereas quite the opposite happened for the hybrid approach with the role played by SVM in emphasizing the difference of inter-class. This proves that our hybrid approach is effective to improve the performance of our activity recognition model.

5.1.4 Conclusion

In this work, we presented a hybrid generative-discriminative approach to automatically identify indoor human activities in video sequences using a combination of HMMs as a generative approach, along with the discriminative SVM. The main motivation behind this choice is to be able to enhance the model's capacity by taking advantage of the powerful

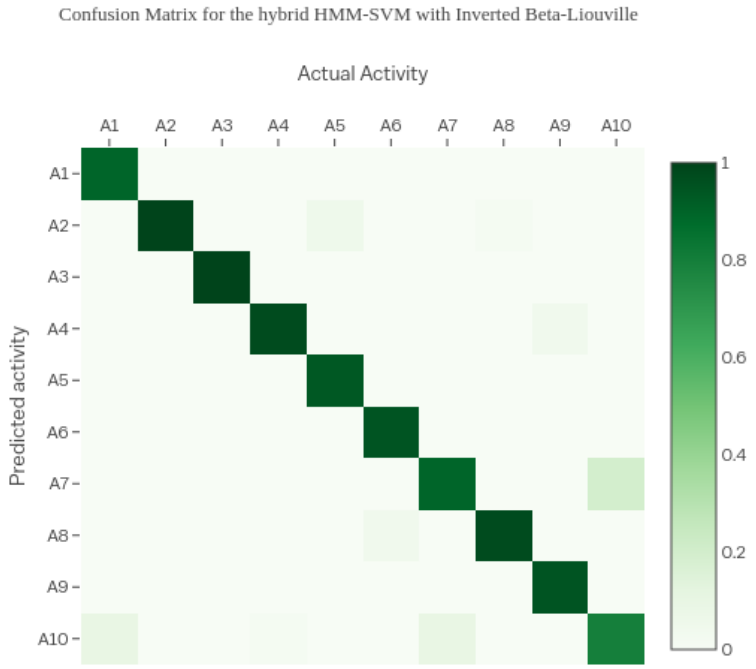


Figure 30: Confusion matrix for the hybrid inverted Beta-Liouville HMM-SVM with UCF101 subset

classification role that SVM plays without neglecting the Spatio-temporal aspects of motion data. We also gave a special focus on modeling positive vectors by using non-Gaussian inverted Dirichlet and inverted Beta-Louiville distributions as emission probabilities for our HMM. We carried out what we believe to be the first attempt of applying IDHMM and IBLHMM both in generative and hybrid modes on the challenging UCF101 benchmark. A comprehensive solution for activity detection and recognition was introduced, where large videos are initially split into video subsequences containing potentially interesting activities, which are then processed in for accurate activity recognition. According to the results obtained from the conducted experiments, we proved that the proposed approach obtained highly accurate recognition rates compared to both generative IDHMM and IBLHMM. Future works are intended to be done in the near future extending this work to different computer vision applications with the use of the Generalized inverted-Dirichlet distribution as emission probabilities to deal with the strictly positive covariance structure imposed

by the inverted Dirichlet form.

5.2 Sentiment Analysis from User Reviews Using a Hybrid Generative-Discriminative HMM-SVM Approach

Sentiment analysis aims to empower automated methods with the capacity to recognize sentiments, opinions and emotions in text. This recognition capacity is now highly demanded to process and extract proper knowledge from the exponentially-growing volume of user-generated data. Applications such as analyzing online products reviews on e-commerce marketplaces, opinion mining from social networks and support chat-bots optimization are putting into practice various methods to perform this complex natural language processing task. In this work, we apply a hybrid generative-discriminative approach using Fisher kernels with generalized inverted Dirichlet-based hidden Markov models to improve the recognition performance in the context of textual analysis. We propose a method that combines HMMs as a generative approach, with the discriminative approach of Support Vector Machine. This strategy allows us to deal with sequential information of the text, and at the same time use the special focus on the classification task that SVM could provide us. Experiments on two challenging user reviews datasets i.e Amazon for products reviews and IMDb movies reviews, demonstrate an effective improvement of the recognition performance compared to the standard generative and Gaussian-based HMM approaches.

5.2.1 Introduction

Opinion mining received massive interest in recent years, particularly with the important role that reviews and shared experiences over e-commerce and marketplaces platforms play in shaping purchase intentions. User-generated data is increasing drastically, especially when it comes to reviews and feedback shared over the internet [51]. This huge volume of data calls for automated methods to process and extract proper knowledge from it [160]. Analyzing users' opinions from different perspectives can considerably help not only the customer to buy or adopt the best product available in the market, but also the merchant, to better understand what are the good or bad features related to their products and determine their effect on the buyers' opinion and feeling regarding the product [241]. These reviews are for the most part available in a text format in an unstructured way and naturally need to be modeled appropriately in order to provide useful insights to both customer and seller. Therefore, recognizing sentiments and attitudes in textual data can provide a better understanding of trends and tendencies related to products [175, 155].

Sentiment analysis, also known as opinion mining, analyzes people's opinions as well as their emotions towards a product, an event or an organization [159]. It has been widely investigated in different research works and approached through different methodologies such as lexicon-based approaches as well as hybrid approaches [57]. Nevertheless, there has been rarely a solid explainability or knowledge behind decisions resulting from these methods, and the latter were oftentimes handled as black-box methods. Challenges in sentiment analysis as a natural language processing application are numerous. In fact, analyzing text reviews implies dealing with text sequences that are usually limited in length, have many misspellings and shortened forms of words [255]. As a result, we have an immense vocabulary size and vectors representing each review are highly sparse.

Two main approaches are used in machine learning to perform recognition tasks: generative

techniques that model the underlying distributions of classes, and discriminative techniques that give a sole focus on learning the class boundaries [211]. Both techniques have been widely used in sentiment analysis to effectively recognize divergent users' attitudes [197, 212, 40].

Hidden Markov Models (HMMs) represent a powerful tool to properly model sequential information within textual data. Their generative aspects constitute a quite potent way to handle sentiment recognition and they tend to require less training data than discriminative models. In the case where the task to be performed is classification, Support Vector Machines (SVM) can clearly distinguish the differences between categories and can thus outperform generative models especially if a large number of training examples are available. SVM is extensively used due to its great capacity to generalize, often resulting in better performance than traditional classification techniques [213].

As a discriminative approach, the main functioning of SVM is to find surfaces that better separate the different data classes using a kernel that allows efficient discrimination in non-linearly separable input feature spaces. Hence the importance of adopting the convenient kernel function which needs to be suitable for the classified data and the objective task. Standard kernels include linear polynomial and radial basis function kernels [25]. Adopting these kernels is not always possible, especially when it comes to classifying objects represented by sequences of different lengths [239]. Consequently, the mentioned kernels may not be a good fit to model our text data. Therefore, a hybrid generative-discriminative approach is adopted to permit the conversion of data into fixed-length and hence provide additional performance to the model.

In this work, we introduce a novel implementation of a hybrid generative-discriminative model and examine its performance on real-life benchmark datasets. We propose the use of Fisher Kernels (FK) generated with Generalized inverted Dirichlet-based HMMs (GIDHMM) to model textual data. Moreover, the use of GID (Generalized Inverted

Dirichlet) to model emission probabilities is backed by the several interesting mathematical properties that this distribution has to offer. These properties allow for a representation of GID samples in a transformed space where features are independent and follow inverted Beta distributions. Adopting this distribution allows us to take advantage of conditional independence among features. This interesting strength is used in this paper to develop a statistical model that essentially handles positive vectors.

In light of the existing methods in sentiment analysis, the main contributions of this paper are the following: First, we apply for the first time a non-Gaussian HMM, i.e. generalized inverted Dirichlet-based HMM on the challenging product reviews benchmark by Amazon and the IMDb movie reviews dataset. Second, we derive a hybrid generative-discriminative approach of our HMM-based framework with FK for SVM-based modeling of positive vectors. This novel approach is also tested on the aforementioned datasets, as an unprecedented attempt of using Fisher Vectors-based hybrid generative-discriminative models to handle textual data analysis. The remainder of the paper is organized as follows, Section 2 presents background topics on sentiment analysis and examines related works. Section 3 discusses the proposed model. Section 4 presents the performed experiments and obtained results. We finally conclude the paper in section 5.

5.2.2 Related work

Sentiment analysis has been the focus of numerous research works, where it has been approached in different levels namely document, sentence and aspect level [100]. While the document level focuses on classifying the whole opinion document into either a positive or negative sentiment, sentence-level looks at determining whether the sentence expresses the nature of opinion (negative, positive, neutral). On the other hand, the aspect-level analysis provides a detail-oriented approach to handle the broad aspect. Thus,

it focuses on determining whether or not a part of the text is opinion-oriented towards a certain aspect. It can present a positive polarity towards one aspect and a negative polarity towards another. Classifying the text as positive or negative depends on the chosen aspect and applied knowledge [162, 161].
It is noteworthy to mention that expressions associated with sentiment are mainly the words or features that express the sentiment of the text, such as adjectives or adverbs. Furthermore, when tackling sentiment analysis, there are mainly three types of machine learning approaches, i.e. supervised, unsupervised and semi-supervised learning and they are respectively used in cases where data is labelled, unlabeled and partially labelled [107, 121].

In HMM-based sentiment analysis, models analyze the input textual data and formulate clusters. After that HMMs are utilized to perform the categorization by considering the clusters as hidden states. Every model analyzes a given instance in order to specify its sentimental polarity. In comparison to related works in the literature, HMM-based methods possess higher interpretability and can model the changing aspects of sentiment information. Multiple sentiment analysis applications adopt HMMs as the main model. In [204] Rabiner proposes a method of predicting sentiments from voice. Also, in [134] authors use HMMs to detect sentiments by considering the label information as positive, negative or neutral. Knowledge about the words' position and hidden states is available and injected into the model. While this approach has shown effective results, it clearly assumes knowledge of the labels and thus requires a significant human effort.

In our work, we do not require knowledge about the states labels and we propose another alternative where we estimate the similarity between the pattern of input text and that of sentences expressing either a positive or negative sentiment, plus we make use of SVM

to increase the model's performance when it comes to the classification accuracy. We detail our method in the next section.

5.2.3 Hybrid Generative-Discriminative approach with Fisher Kernels

When it comes to our adopted approach, a single HMM is trained for every class in the data depending on the context aspect. The resulting likelihoods will be further classified by the SVM classifier to identify the sentiment.

In this section, we present the proposed approach. To illustrate our model, we are first listing various HMM notations and enumerating the upcoming used work script. We then recall the main process behind the forward-backward algorithm. Lastly, we perform a complete derivation of the FK-based model.

5.2.3.1 Hidden Markov Models

A Hidden Markov Model is a statistical model that can be used to describe real-world processes with observable output signals. HMMs are defined as an underlying stochastic process formed by a Markov chain that is not observable (hidden). For each hidden state, a stochastic model creates observable output signals or observations, based on which hidden states can be estimated [203]. We consider a HMM with continuous emissions and K hidden states. We put a set of hidden states $H = \{h_1, ..., h_T\}; h_j \in [1, K]$.

The transition probabilities matrix: $B = \{b_{ij} = P(h_t = j|h_{t-1} = j')\}$ and the emission probabilities matrix: $C = \{c_{ij} = P(m_t = i|h_t = j)\}; i \in [1, M]$ where M is the number of mixture components associated with state j. We define the initial probability: π_j which is the probability to start the observation sequence from the state j.

We denote an HMM as: $\Delta = \{B, C, \varphi, \pi\}$ where φ is the set of mixture parameters depending on the chosen type of mixture.

In this work, we focus on the generalized inverted Dirichlet distribution. Let $\overrightarrow{X}$ a D-dimensional positive vector following a GID distribution. The joint density function is given by Lingappaiah [158] as:

$$p(\overrightarrow{X}|\overrightarrow{\alpha},\overrightarrow{\beta}) = \prod_{d=1}^{D} \frac{\Gamma(\alpha_d+\beta_d)}{\Gamma(\alpha_d)\Gamma(\beta_d)} \frac{X_d^{\alpha_d-1}}{\left(1+\sum_{l=1}^{d} X_d\right)^{\eta_d}} \tag{142}$$

where $\overrightarrow{\alpha} = [\alpha_1, ..., \alpha_D]$, $\overrightarrow{\beta} = [\beta_1, ..., \beta_D]$. η is defined such that $\eta_d = \alpha_d + \beta_d - \beta_{d+1}$ for $d = 0, ..., D$ with $\beta_{D+1} = 0$.

The GID estimation is made simple thanks to an essential propriety, that is if there exists a vector $\overrightarrow{X}$ that follows a GID distribution, then we can come up with another vector $\overrightarrow{W}_n = [\overrightarrow{W}_{n1}, ..., \overrightarrow{W}_{nD}]$ where each element follows an inverted Beta (IB) distribution following the transformation:

$$W_{nd} = f(X_{nd}) = \begin{cases} X_{nd}, & \text{d=1} \\ \frac{X_{nd}}{1+X_{n1}+,...,+X_{nd-1}}, & \text{d=2, ..., D} \end{cases} \tag{143}$$

Then, the multivariate extension of the 2-parameters inverted Beta distribution is given by:

$$p_{IBeta}(W_{nd}|\alpha_{jd},\beta_{jd}) = \frac{\Gamma(\alpha_{jd}+\beta_{jd})}{\Gamma(\alpha_{jd})\Gamma(\beta_{jd})} \frac{W_{nd}^{\alpha_{jd}-1}}{(1+W_{jd})^{(\alpha_{jd}+\beta_{jd})}} \tag{144}$$

The mean of IB is given by:

$$E(W_d) = \frac{\alpha_d}{\beta_d - 1} \tag{145}$$

The variance of IB is given by:

$$Var(W_d) = \frac{\alpha_d(\alpha_d + \beta_d - 1)}{(\beta_d - 2)(\beta_d - 1)^2} \tag{146}$$

5.2.3.2 Inference on hidden states: Forward-Backward Algorithm

The forward algorithm computes the probability of being in state h_j up to time t for the partial observation sequence produced by the model Δ. We consider a forward variable $\gamma_t(i) = P(X_1, X_2, \ldots, X_t, , i_t = h_i|\Delta)$. There is a recursive relationship that is used to compute the former probability. We can resolve for $\gamma_t(i)$ recursively as follows:

1. Initialization:

$$\gamma_t(i) = \pi_i \varphi_i(X_1) \qquad 1 \leq i \leq K \tag{147}$$

2. Recursion:

$$\begin{gathered} \gamma_{t+1}(j) = \left[\sum_{i=1}^{K} \gamma_t(i) b_{ij} \right] \varphi_j(X_{t+1}) \\ for \quad 1 \leq t \leq T-1, \;\; 1 \leq j \leq K \end{gathered} \tag{148}$$

3. Termination:

$$P(X|\Delta) = \sum_{i=1}^{K} \gamma_T(i) \tag{149}$$

The backward variable, which is the probability of the partial observation sequence $X_{t+1}, X_{t+2}, \ldots, X_T$ given the current state is denoted by $\delta_t(i)$ and can similarly be determined as follows:

1. Initialization:

$$\delta_t(i) = 1, \; 1 \leq i \leq K \tag{150}$$

2. Recursion:

$$\delta_t(i) = \sum_{j=1}^{K} b_{ij}\varphi_j(X_{t+1}\delta_{t+1}(j)) \quad for \quad t = T-1, T-2, \ldots, 1 \quad 1 \leq i \leq K \tag{151}$$

3. Termination:

$$\begin{aligned} P(X|\Delta) &= \sum_{i=1}^{K} \gamma_i(T) \\ &= \sum_{i=1}^{K} \pi_i b_i(X_1)\varphi_i(1) \\ &= \sum_{i=1}^{K}\sum_{j=1}^{K} \gamma_t(i) b_{ij}\varphi_j(X_{t+1})\delta_{t+1}(j) \end{aligned} \tag{152}$$

5.2.3.3 Fisher Kernels

Non-linear SVM serves our discrimination needs in the context of realistic recognition tasks. The strategy is to use a Kernel method to avoid calculation cost and memory consumption problems that might arise from performing inner product calculation of high-dimensional feature vectors. It will allow us to implicitly project objects to high-dimensional space by using a kernel function $\kappa(x_\zeta, x_\eta) = \langle \phi(x_\zeta), \phi(x_\eta) \rangle$ and solving the problem with observations x_ζ and x_η represented as Bag of Features (BoF) or Bag of Words (BoW) [119, 249] in general with ϕ being a projection function and $\langle \cdot, \cdot \rangle$ meaning the inner product. Here we choose Fisher Kernel as the kernel function. This choice is motivated by FK being a general way of fusing generative and discriminative approaches for classification. FK is formulated as

$$FK(X_\zeta, X_\eta) = \langle FS(X_\zeta, \Delta), FS(X_\eta, \Delta) \rangle \tag{153}$$

where X_ζ and X_η are two observations, Δ is the parameters set of a generative model

defined by $P(X|\Delta)$ and $FS(X_\zeta, \Delta)$ is the Fisher score.

$$FS(X, \Delta) = \nabla_\Delta \log P(X|\Delta) \tag{154}$$

Given a particular HMM:

$$\begin{aligned} L(X|\Delta) &= \log P(X|\Delta) \\ &= \log \sum_{i=1}^{K} \gamma_T(i) \\ &= \log \sum_{i=1}^{K} \pi_i \varphi_i(X_1)\delta_1(i) \end{aligned} \tag{155}$$

The derivatives for GID-based HMM can be defined as follows:

$$\nabla_\Delta L(X|\Delta) = \left[\frac{\partial L(X|\Delta)}{\partial \pi_i}, \frac{\partial L(X|\Delta)}{\partial b_{ij}}, \frac{\partial L(X|\Delta)}{\partial \alpha_{id}}, \frac{\partial L(X|\Delta)}{\partial \beta_{id}}\right] \tag{156}$$

$$\frac{\partial L(X|\Delta)}{\partial \pi_i} = \frac{\varphi_i(X_1)\delta_1(i)}{\sum_{i=1}^{K} \pi_i \varphi_i(X_1)\delta_1(i)} \tag{157}$$

$$\begin{aligned} \frac{\partial L(X \mid \Delta)}{\partial b_{ij}} &= \frac{1}{P(X \mid \Delta)} \sum_{k=1}^{K} \frac{\partial \gamma_T(k)}{\partial b_{ij}} \\ &= \frac{1}{P(X \mid \Delta)} \sum_{k=1}^{K} \left(\frac{\partial}{\partial b_{ij}} \sum_{l=1}^{K} \gamma_{T-1}(l) b_{lk} \varphi_k\left(X_T\right) \right) \\ &= \frac{1}{P(X \mid \Delta)} \sum_{k=1}^{K} \sum_{l=1}^{K} \frac{\partial \gamma_{T-1}(l)}{\partial b_{ij}} b_{lk} \varphi_k\left(X_T\right) + \partial \gamma_{T-1}(i) \varphi_{ij}\left(X_T\right) \end{aligned} \tag{158}$$

$$\frac{\partial L(X \mid \Delta)}{\partial \alpha_{id}} = \frac{1}{P(X \mid \Delta)} \left(\sum_{j=1}^{K} \sum_{k=1}^{K} \frac{\partial \gamma_{T-1}(k)}{\partial \alpha_{id}} b_{kj} \varphi_j \left(X_T \right) + \sum_{k=1}^{K} \partial \gamma_{T-1}(k) b_{ki} \frac{\partial \varphi_i \left(X_T \right)}{\partial \alpha_{id}} \right) \tag{159}$$

$$\frac{\partial L(X \mid \Delta)}{\partial \beta_{id}} = \frac{1}{P(X \mid \Delta)} \left(\sum_{j=1}^{K} \sum_{k=1}^{K} \frac{\partial \gamma_{T-1}(k)}{\partial \beta_{id}} b_{kj} \varphi_j \left(X_T \right) + \sum_{k=1}^{K} \partial \gamma_{T-1}(k) b_{ki} \frac{\partial \varphi_i \left(X_T \right)}{\partial \beta_{id}} \right) \tag{160}$$

$$\frac{\partial \varphi_i \left(X_t \right)}{\partial \alpha_{id}} = \Psi \left(\alpha_{id} + \beta_{id} \right) - \Psi \left(\alpha_{id} \right) + \log \left(\frac{X_d}{1 + X_d} \right) \tag{161}$$

$$\frac{\partial \varphi_i \left(X_t \right)}{\partial \beta_{id}} = \Psi \left(\alpha_{id} + \beta_{id} \right) - \Psi \left(\beta_{id} \right) + \log \left(\frac{1}{1 + X_d} \right) \tag{162}$$

5.2.4 Experiments

5.2.4.1 Problem Modeling

The main motive behind the use of HMMs in sentiment analysis is the strong analogy behind the process of understanding a sentiment from a text in real-life and the predictive aspect of HMMs. In fact, an opinion consists of a number of words that together represent what the person is trying to express. To understand it, a person would first proceed by reading the words sequentially from left to right, knowing that each word would normally be related to the previous one in a certain way to create a meaningful sentence.

Accordingly, words forming an emotion are modeled as observations in a HMM, while the emotion is the hidden state, which needs to be unveiled. In this work, we propose to tackle this problem of sentiment analysis in a hybrid way; where the HMM states can be modeled

approximately by considering a hidden variable given by patterns that are independent of the class of text. An abstraction of this process is illustrated in figure 31. We first perform word clustering to indicate a certain word pattern, in a way that all negative and positive connotations are clustered separately. After that, we make use of our SVM to further classify the output into negative and positive classes. This treatment requires a dictionary constructed for use by sentiment analysis models.

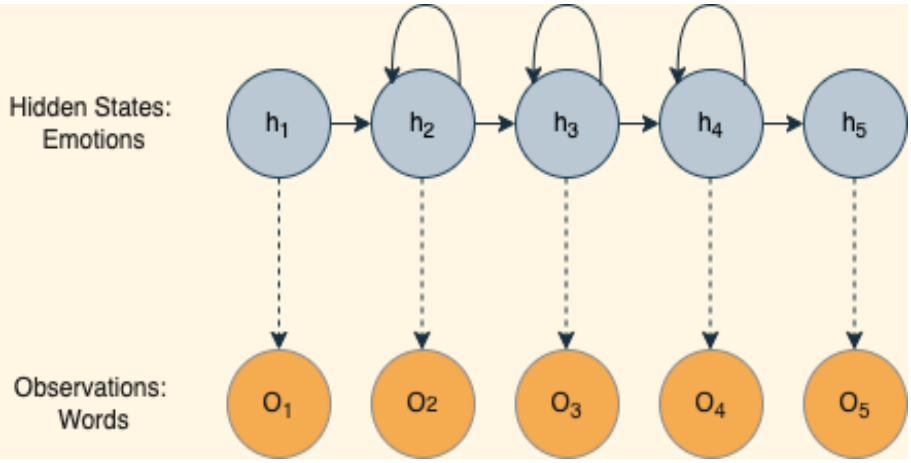


Figure 31: Problem modeling through hidden state-observation HMM

5.2.4.2 Datasets

We choose to experiment on the Amazon[2] reviews dataset from the Stanford Network Analysis Project (SNAP), which spans 18 years period of product reviews [170]. Amazon dataset is a popular corpus of product reviews collected from the Amazon marketplace, which includes ratings and plain text reviews for a multitude of product niches. In this work, we choose to work on the Electronics niche and randomly pick a mix of 30,000 sample reviews from training and testing sets with a vocabulary size of 72.208 unique words. As a word segmentation approach, we use the Part-of-Speech tagger designed by the Stanford NLP group applying default settings. We have also removed numerals,

[2]Publicly available at: https://snap.stanford.edu/data/web-Amazon.html

auxiliary words and verbs, punctuation and stop words as a part of the pre-processing [196]. An output vector is then generated corresponding to the input word. Each review is modeled by a text vector that is obtained after adding up all the word vectors and dividing by the number of words.

We also test our work on the IMDb dataset [165], which is mainly developed for the task of binary sentiment classification of movie reviews. It consists of an equal number of positive and negative reviews. The dataset is evenly divided between training and test sets with 25,000 reviews each. We choose to work on both subsets uniformly and we hence deal with 50,000 samples from each group with 76,340 unique words in total. A similar pre-processing to the one adopted with the Amazon dataset is then applied.

Experiments are carried out in a total of 10 independent runs, each time starting with a different initial observation. The resulting log-likelihoods and accuracies are averaged on these 10 independent runs. It is worth mentioning that through the training process our aim is to maximize the log-likelihood of the data and we target by performing these experiments a higher recognition accuracy each time. Therefore we will not focus on assessing the time complexity in this work. We also use the totality of the dataset for training and testing and do not opt for splitting the data into train and test segments for both datasets.

5.2.4.3 Results

A HMM is trained for each class, in this case, positive and negative. A test text is sent to each model and the probabilities of occurrence are computed. Each HMM returns a probability and the model with the highest probability of occurrence will indicate the class of this text. We choose the number of hidden states to be $K = 2$ for both experiments. In our discussion, we focus on the effect of using a hybrid model as opposed to a HMM-based

model. We also shed some light on the usefulness of using the GID distribution as emission probabilities. Results on both datasets are presented in figure 32.

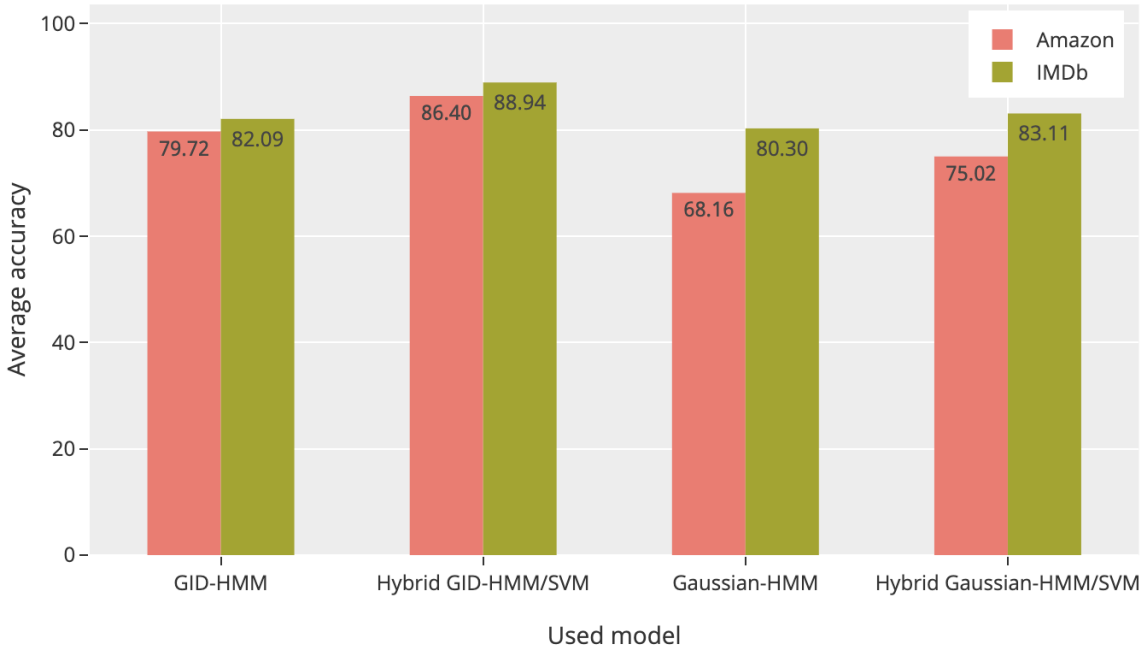


Figure 32: Average accuracies for sentiment recognition on the Amazon and IMDb datasets with each of the tested models

We notice that, on both datasets, Hybrid HMM-SVM models perform remarkably better than Generative-only models, be it GID-based or Gaussian-based. The hybrid GID-based HMM/SVM model achieved average accuracies of 86.40% and 88.94% on the Amazon and IMDb datasets respectively, while we only yielded 79.72% and 82.09% using the GID-HMM Generative-only model. Most importantly, we notice that GID-based models achieved the highest accuracy on both hybrid and generative approaches compared to the Gaussian-based models. This increasing recognition capacity was expected and is once more validated when it comes to positive vector modeling. Using GID as an emission probability distribution, clearly improved the modeling accuracy. Conclusively, results

achieved by applying the hybrid generative-discriminative approach demonstrate the striking increase in terms of the modeling accuracy and further validate the improved performance that SVM provided to the generative technique.

5.2.5 Conclusion

In this work, we presented a hybrid generative-discriminative approach to automatically identify sentiments expressed in user reviews online, using a combination of HMMs as a generative approach, along with the discriminative SVM. The main motivation behind this choice is to be able to enhance the model's capacity by taking advantage of the powerful classification role that SVM plays without neglecting the sequential aspect of text data. We also gave a special focus on modeling positive vectors by using non-Gaussian Generalized Inverted Dirichlet distributions as emission probabilities for our HMM. The interest in adopting the GID for modeling our data arose from the limitations encountered when inverted Dirichlet was adopted, in particular its restraining strictly positive covariance. We carried out what we believe to be the first attempt of applying GIDHMM both in generative and hybrid modes on the challenging Amazon product reviews and IMDb reviews. A comprehensive solution for sentiment detection was introduced, where we allowed the automatic recognition of positive and negative emotions, in textual data. According to the results obtained from the conducted experiments, we proved that the proposed approach obtained highly accurate recognition rates compared to both generative GIDHMM and Gaussian-based HMM. Future works are intended to be done in the near future extending this work to different Natural Language Process applications, such as product recommendation and understanding user intent.

Chapter 6

Conclusion

In this thesis, we studied HMMs and developed a number of their extensions. We focused on one major difficulty that has yet to be addressed when investigating HMMs, namely the use of appropriate distributions as emission probabilities when modeling positive vectors. In reality, despite their widespread use, HMMs are rarely used in a fashion that prioritizes their ability to handle positive vectors. This omission can result in poorly performing HMMs in real-life applications such as image and video categorization or unusual event detection. Conjointly, three other essential challenges that became increasingly critical as we shift towards a modern and data-driven regime, have been brought up: (i) embedding a feature selection method into the framework of HMMs, (ii) adopting an online environment when learning HMM and mixtures parameters and (iii) adopting a hybrid generative-discriminative set up to explore what value can discriminative models add to our generative ones. The above-mentioned challenges have been tackled in a novel manner by proposing unprecedented methods and implementations putting to experiment powerful but rarely utilized distributions such as the Inverted Dirichlet, the Generalized Inverted Dirichlet, and the Inverted Beta-Liouville.

In Chapter 2 we developed a new extension to HMMs by adapting them to handle positive vectors and showcase their capabilities compared to the traditionally GM-based

HMMs. We used Inverted Dirichlet mixtures to bring some light to the powerful modeling capabilities that this distribution has when dealing with non-Gaussian data. Thus we proved that our method is very effective in dynamic texture categorization, recognizing facial expressions from pictures and estimating occupancy in an office setting implemented in the context of smart building development. At that particular point of our research journey, we figured that a generalized ID form could help us deal with the strictly positive covariance structure imposed by the inverted Dirichlet form. We also presumed that to further enhance the accuracy of the model training in recognition applications, we could integrate a feature selection approach qualified to discard irrelevant but compromising features and thus improve the classification task.

In Chapter 3, we addressed problems such as the restraining strictly positive covariance that the Inverted Dirichlet distribution has and adopting a feature selection technique to overcome the need for external knowledge when dealing with features. Therefore, we proposed a framework that applies feature selection to a GID-based HMM. Parameters were learned via a MAP method adding a huge advantage in raising both accuracies of parameter estimates and feature saliencies. Experimental results involving challenging real-life applications such as facial expressions recognition and natural outdoor scene recognition showed that the proposed approach is highly promising.

Chapter 4 was devoted to tackle the challenge of abundant and massive data modeling. We adopted IBL mixtures as emission probabilities for our HMM. The need to utilize this distribution came from the limitations encountered when other distributions such as Gaussian mixtures and inverted Dirichlet were adopted. In fact, IBL mixtures provided a smaller number of parameters compared to the generalized inverted Dirichlet, not to mention that it showed its effectiveness when dealing with positive vector modeling in contrast to the rest of the tested distributions. This work came along with an online learning set up that takes into account the eventual availability of new training data during the learning process.

Our model showed that it could easily incorporate new data without necessarily having to retrain the model.

Finally, yet importantly, in Chapter 5, we proposed a hybrid generative-discriminative approach which we applied to two different real-life applications namely indoor activities recognition and sentiment analysis from text data in a customer experience setting. The motive behind merging those two approaches is the quest to enhance models' capacities by taking advantage of the powerful classification role that SVM plays without neglecting the sequential aspect of data. Always with the goal of modeling data with the appropriate distribution, we tested this hybrid framework using ID, GID and IBL mixtures as emission probabilities. The proposed methods showed high classification capacities in the previously mentioned applications.

This work is the culmination of several research routes pursued with the purpose of fully exposing and exploiting the potential of HMMs. When employing HMMs and not receiving nearly the intended results due to restrictive distributions, the work we completed has long been alluded to as the insightful prospective future work, in research papers and dissertations. Our methods can be used in a variety of academic and industrial settings, and they can assist bridge the semantic gap between system levels and human understanding.

The learning of the associated infinite-form mixture-based HMMs could be the focus of future research. This can contribute to improving the modeling capacity by examining an infinite possible number of classes when we have a stream of new data that can form new classes that were not considered at the start. We're also considering moving in another different route in the future. In fact, deep learning approaches may be one of the most promising directions for modeling very large data sets. Indeed, deep learning is one of the most widely studied methods today, with promising findings in feature representation, classification and categorization. Therefore future work on developing a deep structured generative model based on the proposed methods as a powerful generalization to multiple

layers could be considered.

Bibliography

[1] S. Adams. *Simultaneous feature selection and parameter estimation for hidden Markov models*. PhD thesis, Dissertation, University of Virginia, 2015.

[2] S. Adams and P. A. Beling. A survey of feature selection methods for gaussian mixture models and hidden markov models. *Artificial Intelligence Review*, 52(3):1739–1779, 2019.

[3] S. Adams, P. A. Beling, and R. Cogill. Feature selection for hidden markov models and hidden semi-markov models. *IEEE Access*, 4:1642–1657, 2016.

[4] J. K. Aggarwal and Q. Cai. Human motion analysis: A review. *Computer vision and image understanding*, 73(3):428–440, 1999.

[5] B. Ai, Z. Fan, and R. X. Gao. Occupancy estimation for smart buildings by an auto-regressive hidden markov model. In *2014 American Control Conference*, pages 2234–2239. IEEE, 2014.

[6] M. Al Mashrgy, T. Bdiri, and N. Bouguila. Robust simultaneous positive data clustering and unsupervised feature selection using generalized inverted dirichlet mixture models. *Knowledge-Based Systems*, 59:182–195, 2014.

[7] F. Alalyan, N. Zamzami, and N. Bouguila. A hybrid approach based on svm and bernoulli mixture model for binary vectors classification. In *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 1155–1160. IEEE, 2020.

[8] S. Ali. *Hidden Markov Models and their Extensions for Proportional Sequential Data*. PhD thesis, Concordia University, 2021.

[9] S. Ali and N. Bouguila. Multimodal action recognition using variational-based beta-liouville hidden markov models. *IET Image Processing*, 14(17):4785–4794, 2021.

[10] R. M. Altman. Mixed hidden markov models: an extension of the hidden markov model to the longitudinal data setting. *Journal of the American Statistical Association*, 102(477):201–210, 2007.

[11] M. Amayri, Q.-D. Ngo, and S. Ploix. Estimating occupancy from measurement and knowledge with bayesian networks. In *2016 International Conference on Computational Science and Computational Intelligence (CSCI)*, pages 508–513. IEEE, 2016.

[12] M. Amayri, Q.-D. Ngo, S. Ploix, et al. Bayesian network and hidden markov model for estimating occupancy from measurements and knowledge. In *2017 9th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, volume 2, pages 690–695. IEEE, 2017.

[13] Z. Ambadar, J. W. Schooler, and J. F. Cohn. Deciphering the enigmatic face: The importance of facial dynamics in interpreting subtle facial expressions. *Psychological Science*, 16(5):403–410, 2005. PMID: 15869701.

[14] E. L. Andrade, o. J. Blunsden, and R. B. Fisher. Performance analysis of event detection models in crowded scenes. In *2006 IET International Conference on Visual Information Engineering*, pages 427–432, 2006.

[15] E. L. Andrade, S. Blunsden, and R. B. Fisher. Hidden markov models for optical flow analysis in crowds. In *18th international conference on pattern recognition (ICPR'06)*, volume 1, pages 460–463. IEEE, 2006.

[16] E. L. Andrade, S. Blunsden, and R. B. Fisher. Modelling crowd scenes for event detection. In *18th international conference on pattern recognition (ICPR'06)*, volume 1, pages 175–178. IEEE, 2006.

[17] F. I. Bashir, A. A. Khokhar, and D. Schonfeld. Object trajectory-based activity classification and recognition using hidden markov models. *IEEE Transactions on Image Processing*, 16(7):1912–1919, July 2007.

[18] L. Batista, E. Granger, and R. Sabourin. Dynamic selection of generative–discriminative ensembles for off-line signature verification. *Pattern Recognition*, 45(4):1326–1340, 2012.

[19] L. E. Baum and T. Petrie. Statistical inference for probabilistic functions of finite state markov chains. *The annals of mathematical statistics*, 37(6):1554–1563, 1966.

[20] L. E. Baum and T. Petrie. Statistical inference for probabilistic functions of finite state markov chains. *Ann. Math. Statist.*, 37(6):1554–1563, 12 1966.

[21] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains. *The annals of mathematical statistics*, 41(1):164–171, 1970.

[22] T. Bdiri and N. Bouguila. An infinite mixture of inverted dirichlet distributions. In *International Conference on Neural Information Processing*, pages 71–78. Springer, 2011.

[23] T. Bdiri and N. Bouguila. Learning inverted dirichlet mixtures for positive data clustering. In *International Workshop on Rough Sets, Fuzzy Sets, Data Mining, and Granular-Soft Computing*, pages 265–272. Springer, 2011.

[24] T. Bdiri and N. Bouguila. Positive vectors clustering using inverted dirichlet finite mixture models. *Expert Systems with Applications*, 39(2):1869–1882, 2012.

[25] T. Bdiri and N. Bouguila. Bayesian learning of inverted dirichlet mixtures for svm kernels generation. *Neural Computing and Applications*, 23(5):1443–1458, 2013.

[26] T. Bdiri, N. Bouguila, and D. Ziou. Variational bayesian inference for infinite generalized inverted dirichlet mixtures with feature selection and its application to clustering. *Applied Intelligence*, 44(3):507–525, 2016.

[27] S. R. Beleza and K. Fukui. Slow feature subspace for action recognition. In *International Conference on Pattern Recognition*, pages 702–716. Springer, 2021.

[28] M. Bertini, A. Del Bimbo, and L. Seidenari. Multi-scale and real-time non-parametric approach for anomaly detection and localization. *Computer Vision and Image Understanding*, 116(3):320–329, 2012.

[29] M. Bicego, U. Castellani, and V. Murino. A hidden markov model approach for appearance-based 3d object recognition. *Pattern Recognition Letters*, 26(16):2588–2599, 2005.

[30] J. A. Bilmes et al. A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. *International Computer Science Institute*, 4(510):126, 1998.

[31] C. M. Bishop. *Pattern recognition and machine learning*. springer, 2006.

[32] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[33] A. L. Blum and P. Langley. Selection of relevant features and examples in machine learning. *Artificial intelligence*, 97(1-2):245–271, 1997.

[34] A. Bosch, A. Zisserman, and X. Muñoz. Scene classification via plsa. In A. Leonardis, H. Bischof, and A. Pinz, editors, *Computer Vision – ECCV 2006*, pages 517–530, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[35] L. Bottou et al. Online learning and stochastic approximations. *On-line learning in neural networks*, 17(9):142, 1998.

[36] N. Bouguila. Infinite liouville mixture models with application to text and texture categorization. *Pattern Recognition Letters*, 33(2):103–110, 2012.

[37] N. Bouguila and W. ElGuebaly. Discrete data clustering using finite mixture models. *Pattern Recognition*, 42(1):33–42, 2009.

[38] N. Bouguila and W. Fan. *Mixture models and applications*. Springer, 2020.

[39] N. Bouguila and D. Ziou. High-dimensional unsupervised selection and estimation of a finite generalized dirichlet mixture model based on minimum message length. *IEEE transactions on pattern analysis and machine intelligence*, 29(10):1716–1731, 2007.

[40] W. Bourequat and H. Mourad. Sentiment analysis approach for analyzing iphone release using support vector machine. *International Journal of Advances in Data and Information Systems*, 2(1):36–44, 2021.

[41] S. Bourouis, R. Alroobaea, S. Rubaiee, M. Andejany, F. M. Almansour, and N. Bouguila. Markov chain monte carlo-based bayesian inference for learning finite and infinite inverted beta-liouville mixture models. *IEEE Access*, 9:71170–71183, 2021.

[42] C. Boutsidis, M. W. Mahoney, and P. Drineas. Unsupervised feature selection for the k-means clustering problem. In *Proceedings of the 22nd International Conference on Neural Information Processing Systems*, NIPS'09, page 153–161, Red Hook, NY, USA, 2009. Curran Associates Inc.

[43] A. P. Bradley. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern recognition*, 30(7):1145–1159, 1997.

[44] J. Brooks, S. Kumar, S. Goyal, R. Subramany, and P. Barooah. Energy-efficient control of under-actuated hvac zones in commercial buildings. *Energy and Buildings*, 93:160–168, 2015.

[45] L. M. Candanedo and V. Feldheim. Accurate occupancy detection of an office room from light, temperature, humidity and co2 measurements using statistical learning models. *Energy and Buildings*, 112:28–39, 2016.

[46] O. Cappé. Online em algorithm for hidden markov models. *Journal of Computational and Graphical Statistics*, 20(3):728–749, 2011.

[47] R. A. Cárdenas-Ovando, E. A. Fernández-Figueroa, H. A. Rueda-Zárate, J. Noguez, and C. Rangel-Escareño. A feature selection strategy for gene expression time series experiments with hidden markov models. *Plos one*, 14(10):e0223183, 2019.

[48] L. Carnevali, F. Santoni, and E. Vicario. Learning marked markov modulated poisson processes for online predictive analysis of attack scenarios. In *2019 IEEE 30th International Symposium on Software Reliability Engineering (ISSRE)*, pages 195–205. IEEE, 2019.

[49] T. Caughey. Classical normal modes in damped linear dynamic systems. *Journal of Applied Mechanics*, 27(2):269–271, 1960.

[50] S. Chackravarthy, S. Schmitt, and L. Yang. Intelligent crime anomaly detection in smart cities using deep learning. In *2018 IEEE 4th International Conference on Collaboration and Internet Computing (CIC)*, pages 399–404. IEEE, 2018.

[51] K. Chakraborty, S. Bhattacharyya, and R. Bag. A survey of sentiment analysis from social media data. *IEEE Transactions on Computational Social Systems*, 7(2):450–464, 2020.

[52] I. Channoufi, S. Bourouis, N. Bouguila, and K. Hamrouni. Color image segmentation with bounded generalized gaussian mixture model and feature selection. In *2018 4th International Conference on Advanced Technologies for Signal and Image Processing (ATSIP)*, pages 1–6. IEEE, 2018.

[53] L. Chen, D. Barber, and J.-M. Odobez. Dynamical dirichlet mixture model. Technical report, IDIAP, 2007.

[54] M. Cholewa and P. Głomb. Estimation of the number of states for gesture recognition with hidden markov models based on the number of critical points in time sequence. *Pattern Recognition Letters*, 34(5):574–579, 2013.

[55] R. J. Connor and J. E. Mosimann. Concepts of independence for proportions with a generalization of the dirichlet distribution. *Journal of the American Statistical Association*, 64(325):194–206, 1969.

[56] A. Coronato and G. De Pietro. Situation awareness in applications of ambient assisted living for cognitive impaired people. *Mobile Networks and Applications*, 18(3):444–453, 2013.

[57] K. Cortis and B. Davis. Over a decade of social opinion mining. *arXiv e-prints*, pages arXiv–2012, 2020.

[58] Dacheng Tao, Xiaoou Tang, Xuelong Li, and Xindong Wu. Asymmetric bagging and random subspace for support vector machines-based relevance feedback in image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(7):1088–1099, 2006.

[59] K. J. Dana, B. Van Ginneken, S. K. Nayar, and J. J. Koenderink. Reflectance and texture of real-world surfaces. *ACM Transactions On Graphics (TOG)*, 18(1):1–34, 1999.

[60] R. Datta, W. Ge, J. Li, and J. Z. Wang. Toward bridging the annotation-retrieval gap in image search by a generative modeling approach. In *Proceedings of the 14th ACM international conference on Multimedia*, pages 977–986. ACM, 2006.

[61] E. Demir, E. Köseoğlu, R. Sokullu, and B. Şeker. Smart home assistant for ambient assisted living of elderly people with dementia. *Procedia computer science*, 113:609–614, 2017.

[62] U. Demir, Y. S. Rawat, and M. Shah. Tinyvirat: Low-resolution video action recognition. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 7387–7394, 2021.

[63] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977.

[64] L. Denoyer, H. Zaragoza, and P. Gallinari. HMM-based passage models for document classification and ranking. In *ECIR'01 - 23rd European Colloquium on Information Retrieval Research*, pages 126–135, Darmstadt, Germany, 2001.

[65] M. A. A. Dewan, M. Murshed, and F. Lin. Engagement detection in online learning: a review. *Smart Learning Environments*, 6(1):1, 2019.

[66] P. M. Djuric and J.-H. Chun. An mcmc sampling approach to estimation of nonstationary hidden markov models. *IEEE Transactions on Signal Processing*, 50(5):1113–1123, 2002.

[67] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *2005 IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, pages 65–72, 2005.

[68] B. Dong, B. Andrews, K. P. Lam, M. Höynck, R. Zhang, Y.-S. Chiou, and D. Benitez. An information technology enabled sustainability test-bed (itest) for occupancy detection through an environmental sensing network. *Energy and Buildings*, 42(7):1038–1046, 2010.

[69] G. Doretto, A. Chiuso, Y. N. Wu, and S. Soatto. Dynamic textures. *International Journal of Computer Vision*, 51(2):91–109, 2003.

[70] A. Drosou, D. Ioannidis, K. Moustakas, and D. Tzovaras. Spatiotemporal analysis of human activities for biometric authentication. *Computer Vision and Image Understanding*, 116(3):411–421, 2012.

[71] R. O. Duda, P. E. Hart, and D. G. Stork. Pattern classification second edition john wiley & sons. *New York*, 58:16, 2001.

[72] J. G. Dy and C. E. Brodley. Feature selection for unsupervised learning. *Journal of machine learning research*, 5(Aug):845–889, 2004.

[73] A. Ebadat, G. Bottegal, D. Varagnolo, B. Wahlberg, and K. H. Johansson. Regularized deconvolution-based approaches for estimating room occupancies. *IEEE Transactions on Automation Science and Engineering*, 12(4):1157–1168, 2015.

[74] C. Eckart and G. Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, 1936.

[75] G. Edwards, A. Lanitis, C. Taylor, and T. Cootes. Statistical models of face images — improving specificity. *Image and Vision Computing*, 16(3):203 – 211, 1998.

[76] O. Elharrouss, N. Almaadeed, S. Al-Maadeed, A. Bouridane, and A. Beghdadi. A combined multiple action recognition and summarization for surveillance video sequences. *Applied Intelligence*, 51(2):690–712, 2021.

[77] H. Eom, Y. Son, and S. Choi. Feature-selective ensemble learning-based long-term regional pv generation forecasting. *IEEE access*, 8:54620–54630, 2020.

[78] E. Epaillard and N. Bouguila. Hidden markov models based on generalized dirichlet mixtures for proportional data modeling. In *IAPR Workshop on Artificial Neural Networks in Pattern Recognition*, pages 71–82. Springer, 2014.

[79] E. Epaillard and N. Bouguila. Proportional data modeling with hidden markov models based on generalized dirichlet and beta-liouville mixtures applied to anomaly detection in public areas. *Pattern Recognition*, 55:125–136, 2016.

[80] E. Epaillard and N. Bouguila. Variational bayesian learning of generalized dirichlet-based hidden markov models applied to unusual events detection. *IEEE transactions on neural networks and learning systems*, 30(4):1034–1047, 2018.

[81] E. Epaillard, N. Bouguila, and D. Ziou. Classifying textures with only 10 visual-words using hidden markov models with dirichlet mixtures. In *Adaptive and Intelligent Systems*, pages 20–28. Springer, 2014.

[82] V. L. Erickson, M. Á. Carreira-Perpiñán, and A. E. Cerpa. Observe: Occupancy-based system for efficient reduction of hvac energy. In *Proceedings of the 10th ACM/IEEE International Conference on Information Processing in Sensor Networks*, pages 258–269. IEEE, 2011.

[83] N. Esfandian, F. Razzazi, and A. Behrad. A clustering based feature selection method in spectro-temporal domain for speech recognition. *Engineering Applications of Artificial Intelligence*, 25(6):1194–1202, 2012.

[84] C. P. Ezenkwu, U. I. Akpan, and B. U.-A. Stephen. A class-specific metaheuristic technique for explainable relevant feature selection. *Machine Learning with Applications*, 6:100142, 2021.

[85] W. Fan and N. Bouguila. Learning finite beta-liouville mixture models via variational bayes for proportional data clustering. In *Twenty-Third International Joint Conference on Artificial Intelligence*, 2013.

[86] W. Fan and N. Bouguila. Online learning of a dirichlet process mixture of beta-liouville distributions via variational inference. *IEEE Transactions on Neural Networks and Learning Systems*, 24(11):1850–1862, 2013.

[87] W. Fan and N. Bouguila. Dynamic textures clustering using a hierarchical pitman-yor process mixture of dirichlet distributions. In *2015 IEEE International Conference on Image Processing (ICIP)*, pages 296–300. IEEE, 2015.

[88] W. Fan and N. Bouguila. Topic novelty detection using infinite variational inverted dirichlet mixture models. In *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*, pages 70–75, 2015.

[89] W. Fan and N. Bouguila. An accelerated variational framework for face expression recognition. *2018 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom)*, pages 1–5, 2018.

[90] W. Fan and N. Bouguila. Simultaneous clustering and feature selection via non-parametric pitman–yor process mixture models. *International Journal of Machine Learning and Cybernetics*, 10(10):2753–2766, 2019.

[91] W. Fan, H. Sallay, N. Bouguila, and S. Bourouis. A hierarchical dirichlet process mixture of generalized dirichlet distributions for feature selection. *Computers & Electrical Engineering*, 43:48–65, 2015.

[92] W. Fan, R. Wang, and N. Bouguila. Simultaneous positive sequential vectors modeling and unsupervised feature selection via continuous hidden markov models. *Pattern Recognition*, 119:108073, 2021.

[93] W. Fan, L. Yang, N. Bouguila, and Y. Chen. Sequentially spherical data modeling with hidden markov models and its application to fmri data analysis. *Knowledge-Based Systems*, 206:106341, 2020.

[94] K.-T. Fang, S. Kotz, and K. W. Ng. *Symmetric multivariate and related distributions*. Chapman and Hall/CRC, 2018.

[95] T. A. Farmer, M. Brown, and M. K. Tanenhaus. Prediction, explanation, and the role of generative models in language processing. *Behavioral and Brain Sciences*, 36(3):211–212, 2013.

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

[97] A. Fathima and K. Vaidehi. Review on facial expression recognition system using machine learning techniques. In *Advances in Decision Sciences, Image Processing, Security and Computer Vision*, pages 608–618. Springer, 2020.

[98] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pages 178–178. IEEE, 2004.

[99] L. Fei-Fei, A. Iyer, C. Koch, and P. Perona. What do we perceive in a glance of a real-world scene? *Journal of vision*, 7(1):10–10, 2007.

[100] R. Feldman. Techniques and applications for sentiment analysis. *Communications of the ACM*, 56(4):82–89, 2013.

[101] J. Ferguson. pp. 143–179, variable duration models for speech. In *Proc. of the Symposium on the applications of hidden Markov models to text and speech, JD Ferguson, Ed. Princeton: IDA-CRD*, 1980.

[102] M. Figueiredo and A. Jain. Unsupervised learning of finite mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3):381–396, 2002.

[103] E. Fons, P. Dawson, J. Yau, X. jun Zeng, and J. Keane. A novel dynamic asset allocation system using Feature Saliency Hidden Markov models for smart beta investing. Papers 1902.10849, arXiv.org, Feb. 2019.

[104] M. Frydenberg. The chain graph markov property. *Scandinavian Journal of Statistics*, pages 333–353, 1990.

[105] T. Fuse and K. Kamiya. Statistical anomaly detection in human dynamics monitoring using a hierarchical dirichlet process hidden markov model. *IEEE Transactions on Intelligent Transportation Systems*, 18(11):3083–3092, 2017.

[106] M. Gales, S. Young, et al. The application of hidden markov models in speech recognition. *Foundations and Trends® in Signal Processing*, 1(3):195–304, 2008.

[107] G. Gautam and D. Yadav. Sentiment analysis of twitter data using machine learning approaches and semantic analysis. In *2014 Seventh International Conference on Contemporary Computing (IC3)*, pages 437–442, 2014.

[108] J.-L. Gauvain and C.-H. Lee. Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains. *IEEE transactions on speech and audio processing*, 2(2):291–298, 1994.

[109] E. A. Gehan. A generalized wilcoxon test for comparing arbitrarily singly-censored samples. *Biometrika*, 52(1-2):203–224, 1965.

[110] Z. Ghahramani. An introduction to hidden markov models and bayesian networks. In *Hidden Markov models: applications in computer vision*, pages 9–41. World Scientific, 2001.

[111] Z. Ghahramani and M. I. Jordan. Factorial hidden markov models. In *Advances in Neural Information Processing Systems*, pages 472–478, 1996.

[112] Z. Ghahramani and M. I. Jordan. Factorial hidden markov models. *Machine learning*, 29(2):245–273, 1997.

[113] M. Graña, M. Termenon, A. Savio, A. Gonzalez-Pinto, J. Echeveste, J. Pérez, and A. Besga. Computer aided diagnosis system for alzheimer disease using brain diffusion tensor imaging features selected by pearson's correlation. *Neuroscience letters*, 502(3):225–229, 2011.

[114] R. D. Gupta and D. S. P. Richards. Multivariate liouville distributions, iii. *Journal of Multivariate Analysis*, 43(1):29–57, 1992.

[115] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *J. Mach. Learn. Res.*, 3(null):1157–1182, Mar. 2003.

[116] Z. Hajirahimi and M. Khashei. Hybrid structures in time series modeling and forecasting: A review. *Engineering Applications of Artificial Intelligence*, 86:83–106, 2019.

[117] L. P. Hansen. Large sample properties of generalized method of moments estimators. *Econometrica: Journal of the Econometric Society*, pages 1029–1054, 1982.

[118] M. B. Harms, A. Martin, and G. L. Wallace. Facial emotion recognition in autism spectrum disorders: a review of behavioral and neuroimaging studies. *Neuropsychology review*, 20(3):290–322, 2010.

[119] Z. S. Harris. Distributional structure. *Word*, 10(2-3):146–162, 1954.

[120] J. A. Hartigan and M. A. Wong. Algorithm as 136: A k-means clustering algorithm. *Journal of the royal statistical society. series c (applied statistics)*, 28(1):100–108, 1979.

[121] T. Hastie, R. Tibshirani, and J. Friedman. Unsupervised learning. In *The elements of statistical learning*, pages 485–585. Springer, 2009.

[122] S. Hegde, K. Achary, and S. Shetty. Feature selection using fisher's ratio technique for automatic speech recognition. *arXiv preprint arXiv:1505.03239*, 2015.

[123] T. Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '99, page 50–57, New York, NY, USA, 1999. Association for Computing Machinery.

[124] T. Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning*, 42:177–196, 2001.

[125] T. Hofmann. Probabilistic latent semantic analysis, 2013.

[126] J. H. Holland. Genetic algorithms. *Scientific american*, 267(1):66–73, 1992.

[127] C. Hu, W. Fan, J.-X. Du, and N. Bouguila. A novel statistical approach for clustering positive data based on finite inverted beta-liouville mixture models. *Neurocomputing*, 333:110–123, 2019.

[128] Y. Huang, K. B. Englehart, B. Hudgins, and A. D. Chan. A gaussian mixture model based classification scheme for myoelectric control of powered upper limb prostheses. *IEEE Transactions on Biomedical Engineering*, 52(11):1801–1811, 2005.

[129] S. U. Innocenti, F. Becattini, F. Pernici, and A. Del Bimbo. Temporal binary representation for event-based action recognition. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 10426–10432, 2021.

[130] T. S. Jaakkola, D. Haussler, et al. Exploiting generative models in discriminative classifiers. *Advances in neural information processing systems*, pages 487–493, 1999.

[131] T. Jebara and A. P. Pentland. *Discriminative, generative and imitative learning*. PhD thesis, PhD thesis, Media laboratory, MIT, 2001.

[132] L. Ji, Y. Ren, G. Liu, and X. Pu. Training-based gradient lbp feature models for multiresolution texture classification. *IEEE Transactions on Cybernetics*, 48(9):2683–2696, 2017.

[133] F. Jiang, Y. Wu, and A. K. Katsaggelos. Abnormal event detection from surveillance video by dynamic hierarchical clustering. In *2007 IEEE International Conference on Image Processing*, volume 5, pages V–145. IEEE, 2007.

[134] W. Jin, H. H. Ho, and R. K. Srihari. Opinionminer: A novel machine learning system for web opinion mining and extraction. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, page 1195–1204, New York, NY, USA, 2009. Association for Computing Machinery.

[135] T. Joachims. Text categorization with support vector machines: Learning with many relevant features. In *European conference on machine learning*, pages 137–142. Springer, 1998.

[136] B. H. Juang and L. R. Rabiner. Hidden markov models for speech recognition. *Technometrics*, 33(3):251–272, 1991.

[137] B. Julesz. Textons, the elements of texture perception, and their interactions. *Nature*, 290(5802):91, 1981.

[138] E. J. Justino, A. El Yacoubi, F. Bortolozzi, and R. Sabourin. An off-line signature verification system using hmm and graphometric features. In *Proc. of the 4th International Workshop on Document Analysis Systems*, pages 211–222. Citeseer, 2000.

[139] T. Kailath. The divergence and bhattacharyya distance measures in signal selection. *IEEE transactions on communication technology*, 15(1):52–60, 1967.

[140] M. Kang, J. Ahn, and K. Lee. Opinion mining using ensemble text hidden markov models for text classification. *Expert Systems with Applications*, 94:218–227, 2018.

[141] S.-R. Ke, H. L. U. Thuc, Y.-J. Lee, J.-N. Hwang, J.-H. Yoo, and K.-H. Choi. A review on video-based human activity recognition. *Computers*, 2(2):88–131, 2013.

[142] N. A. Khan, S. A. Waheeb, A. Riaz, and X. Shang. A three-stage teacher, student neural networks and sequential feed forward selection-based feature selection approach for the classification of autism spectrum disorder. *Brain sciences*, 10(10):754, 2020.

[143] K. H. Kim, K. Park, H. Kim, B. Jo, S. H. Ahn, C. Kim, M. Kim, T. H. Kim, S. B. Lee, D. Shin, et al. Facial expression monitoring system for predicting patient's sudden movement during radiotherapy using deep learning. *Journal of Applied Clinical Medical Physics*, 2020.

[144] J. Kittler, P. Pudil, and P. Somol. Advances in statistical feature selection. In *Proceedings of the Second International Conference on Advances in Pattern Recognition*, ICAPR '01, page 425–434, Berlin, Heidelberg, 2001. Springer-Verlag.

[145] M. H. C. Law, M. A. T. Figueiredo, and A. K. Jain. Simultaneous feature selection and clustering using mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1154–1166, Sep. 2004.

[146] S. Lazebnik, C. Schmid, and J. Ponce. A sparse texture representation using local affine regions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8):1265–1278, 2005.

[147] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 2169–2178. IEEE, 2006.

[148] S. Le Corff and G. Fort. Online expectation maximization based algorithms for inference in hidden markov models. *Electronic Journal of Statistics*, 7:763–792, 2013.

[149] C. Lee and G. G. Lee. Information gain and divergence-based feature selection for machine learning-based text categorization. *Information processing & management*, 42(1):155–165, 2006.

[150] H. Lee, D. Lee, and H.-J. Lee. A predictive initialization of hidden state parameters in a hidden markov model for hand gesture recognition. In *2018 IEEE International Conference on Consumer Electronics-Asia (ICCE-Asia)*, pages 206–212. IEEE, 2018.

[151] W.-C. Lee and D. Yoon. A study on facial expression and first impression through machine learning. In *2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, pages 298–301. IEEE, 2019.

[152] T. Leung and J. Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *International journal of computer vision*, 43(1):29–44, 2001.

[153] S. E. Levinson, L. R. Rabiner, and M. M. Sondhi. An introduction to the application of the theory of probabilistic functions of a markov process to automatic speech recognition. *Bell System Technical Journal*, 62(4):1035–1074, 1983.

[154] J. Li, A. Najmi, and R. M. Gray. Image classification by a two-dimensional hidden markov model. *IEEE transactions on signal processing*, 48(2):517–533, 2000.

[155] R. Liang and J.-q. Wang. A linguistic intuitionistic cloud decision support model with sentiment analysis for product selection in e-commerce. *International Journal of Fuzzy Systems*, 21(3):963–977, 2019.

[156] M. J. Lindstrom and D. M. Bates. Newton—raphson and em algorithms for linear mixed-effects models for repeated-measures data. *Journal of the American Statistical Association*, 83(404):1014–1022, 1988.

[157] M. J. Lindstrom and D. M. Bates. Newton—raphson and em algorithms for linear mixed-effects models for repeated-measures data. *Journal of the American Statistical Association*, 83(404):1014–1022, 1988.

[158] G. Lingappaiah. On the generalised inverted dirichlet distribution. *Demostratio Mathematica*, 9(3):423–433, 1976.

[159] B. Liu. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167, 2012.

[160] B. Liu. *Sentiment analysis: Mining opinions, sentiments, and emotions*. Cambridge university press, 2020.

[161] M. Liu, F. Zhou, K. Chen, and Y. Zhao. Co-attention networks based on aspect and context for aspect-level sentiment analysis. *Knowledge-Based Systems*, 217:106810, 2021.

[162] N. Liu and B. Shen. Rememnn: A novel memory neural network for powerful interaction in aspect-based sentiment analysis. *Neurocomputing*, 395:66–77, 2020.

[163] P. Liu, S.-K. Nguang, and A. Partridge. Occupancy inference using pyroelectric infrared sensors through hidden markov models. *IEEE Sensors Journal*, 16(4):1062–1068, 2016.

[164] D. G. Lowe. Object recognition from local scale-invariant features. In *iccv*, page 1150. Ieee, 1999.

[165] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.

[166] K. Mahantesh, V. M. Aradhya, and S. Niranjan. A study of subspace mixture models with different classifiers for very large object classification. In *2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 540–544. IEEE, 2014.

[167] S. Maldonado and R. Weber. A wrapper method for feature selection using support vector machines. *Information Sciences*, 179(13):2208–2217, 2009.

[168] S. Mao, D. Tao, G. Zhang, P. Ching, and T. Lee. Revisiting hidden markov models for speech emotion recognition. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6715–6719. IEEE, 2019.

[169] R. E. Mayer. Searching for the role of emotions in e-learning. *Learning and Instruction*, 70:101213, 2020.

[170] J. McAuley and J. Leskovec. Hidden factors and hidden topics: Understanding rating dimensions with review text. In *Proceedings of the 7th ACM Conference on Recommender Systems*, RecSys '13, page 165–172, New York, NY, USA, 2013. Association for Computing Machinery.

[171] R. Mehran, A. Oyama, and M. Shah. Abnormal crowd behavior detection using social force model. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 935–942. IEEE, 2009.

[172] M. Momenzadeh, M. Sehhati, and H. Rabbani. A novel feature selection method for microarray data classification based on hidden markov model. *Journal of Biomedical Informatics*, 95:103213, 2019.

[173] G. Mongillo and S. Deneve. Online learning with hidden markov models. *Neural computation*, 20(7):1706–1716, 2008.

[174] J. Montero and L. Sucar. Feature selection for visual gesture recognition using hidden markov models. In *Proceedings of the Fifth Mexican International Conference in Computer Science, 2004. ENC 2004.*, pages 196–203. IEEE, 2004.

[175] J. Moreno-Garcia and J. Rosado. Using syntactic analysis to enhance aspect based sentiment analysis. In J. Medina, M. Ojeda-Aciego, J. L. Verdegay, D. A. Pelta, I. P. Cabrera, B. Bouchon-Meunier, and R. R. Yager, editors, *Information Processing and Management of Uncertainty in Knowledge-Based Systems. Theory and Foundations*, pages 671–682, Cham, 2018. Springer International Publishing.

[176] K. P. Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

[177] F. Najar, S. Bourouis, N. Bouguila, and S. Belghith. A fixed-point estimation algorithm for learning the multivariate ggmm: application to human action recognition. In *2018 IEEE Canadian Conference on Electrical & Computer Engineering (CCECE)*, pages 1–4. IEEE, 2018.

[178] R. Nasfi, M. Amayri, and N. Bouguila. A novel approach for modeling positive vectors with inverted dirichlet-based hidden markov models. *Knowledge-Based Systems*, 192:105335, 2020.

[179] R. Nasfi and N. Bouguila. Indoor activity recognition using a hybrid generative-discriminative approach with hidden markov models and support vector machines. In *2022 23rd IEEE International Conference on Industrial Technology (ICIT)*, 2022. Submitted for publication.

[180] R. Nasfi and N. Bouguila. A novel feature selection method using generalized inverted dirichlet-based hmms for image categorization. *International Journal of Machine Learning and Cybernetics*, pages 1–17, 2022.

[181] R. Nasfi and N. Bouguila. Online learning of inverted beta-liouville hmms for anomaly detection in crowd scenes. In *Hidden Markov Models and Applications*. Springer, 2022.

[182] R. Nasfi and N. Bouguila. Sentiment analysis from user reviews using a hybrid generative-discriminative hmm-svm approach. In *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*. Springer, 2022. Submitted for publication.

[183] R. Nasfi and M. Soui. Extraction of interesting adaptation rules. *Procedia Computer Science*, 34:607–612, 2014.

[184] A. Nefian and M. Hayes. Face recognition using an embedded hmm. In *IEEE Conference on Audio and Video-based Biometric Person Authentication*, pages 19–24, 1999.

[185] V.-L. Nguyen, N.-S. Vu, and P.-H. Gosselin. A scattering transform combination with local binary pattern for texture classification. In *2016 14th International Workshop on Content-Based Multimedia Indexing (CBMI)*, pages 1–4. IEEE, 2016.

[186] P. E. Nikravesh. *Computer-aided analysis of mechanical systems*. Prentice-Hall, Inc., 1988.

[187] T. L. Nwe, S. W. Foo, and L. C. De Silva. Speech emotion recognition using hidden markov models. *Speech communication*, 41(4):603–623, 2003.

[188] U. of Minnesota. Unusual crowd activity dataset of university of minnesota, available from: http://mha.cs.umn.edu/movies/crowd-activity-all.avi.

[189] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002.

[190] T. Ojala, M. Pietikäinen, and T. Mäenpää. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24:971–987, 2002.

[191] F. Oldewurtel, D. Sturzenegger, and M. Morari. Importance of occupancy information for building climate control. *Applied energy*, 101:521–532, 2013.

[192] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International journal of computer vision*, 42(3):145–175, 2001.

[193] L. S. Oliveira, E. Justino, C. Freitas, and R. Sabourin. The graphology applied to signature verification. In *12th Conference of the International Graphonomics Society*, pages 286–290, 2005.

[194] T. Otsuka and J. Ohya. Recognition of facial expressions using hmm with continuous output probabilities. In *Proceedings 5th IEEE International Workshop on Robot and Human Communication. RO-MAN'96 TSUKUBA*, pages 323–328. IEEE, 1996.

[195] J. Pan, C. C. Ferrer, K. McGuinness, N. E. O'Connor, J. Torres, E. Sayrol, and X. Giro-i Nieto. Salgan: Visual saliency prediction with generative adversarial networks. *arXiv preprint arXiv:1701.01081*, 2017.

[196] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[197] I. Perikos, S. Kardakis, M. Paraskevas, and I. Hatzilygeroudis. Hidden markov models for sentiment analysis in social media. In *2019 IEEE International Conference on Big Data, Cloud Computing, Data Science Engineering (BCD)*, pages 130–135, 2019.

[198] R. Péteri, S. Fazekas, and M. J. Huiskes. Dyntex: A comprehensive database of dynamic textures. *Pattern Recognition Letters*, 31(12):1627–1632, 2010.

[199] B. R. Povlow and S. M. Dunn. Texture classification using noncausal hidden markov models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(10):1010–1014, 1995.

[200] A. Punzo and A. Maruotti. Clustering multivariate longitudinal observations: The contaminated gaussian hidden markov model. *Journal of Computational and Graphical Statistics*, 25(4):1097–1098, 2016.

[201] Y. Qiao and L. Weng. Hidden markov model based dynamic texture classification. *IEEE Signal Processing Letters*, 22(4):509–512, 2015.

[202] Y.-L. Qiao and Z.-Y. Xing. Dynamic texture classification using multivariate hidden markov model. *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences*, 101(1):302–305, 2018.

[203] L. Rabiner and B. Juang. An introduction to hidden markov models. *ieee assp magazine*, 3(1):4–16, 1986.

[204] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[205] P. Raei and A. Bouchachia. A literature review on the design of smart homes for people with dementia using a user-centred design approach. In *Proceedings of the 30th International BCS Human Computer Interaction Conference 30*, pages 1–8, 2016.

[206] M. Rahmaninia and P. Moradi. Osfsmi: online stream feature selection method based on mutual information. *Applied Soft Computing*, 68:733–746, 2018.

[207] A. Ravichandran, R. Chaudhry, and R. Vidal. Categorizing dynamic textures using a bag of dynamical systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(2):342–353, 2013.

[208] P. C. Ribeiro, R. Audigier, and Q. C. Pham. Rimoc, a feature to discriminate unstructured motions: Application to violence detection for video-surveillance. *Computer vision and image understanding*, 144:121–143, 2016.

[209] C. Robert and G. Casella. *Monte Carlo statistical methods*. Springer Science & Business Media, 2013.

[210] N. Rostamzadeh, G. Zen, I. Mironică, J. Uijlings, and N. Sebe. Daily living activities recognition via efficient high and low level cues combination and fisher kernel representation. In *International Conference on Image Analysis and Processing*, pages 431–441. Springer, 2013.

[211] Y. D. Rubinstein, T. Hastie, et al. Discriminative vs informative learning. In *KDD*, volume 5, pages 49–53, 1997.

[212] S. Rustamov, E. Mustafayev, and M. A. Clements. An application of hidden markov models in subjectivity analysis. In *2013 7th International Conference on Application of Information and Communication Technologies*, pages 1–4. IEEE, 2013.

[213] M. R. Saleh, M. T. Martín-Valdivia, A. Montejo-Ráez, and L. Ureña-López. Experiments with svm to classify opinions in different domains. *Expert Systems with Applications*, 38(12):14799–14804, 2011.

[214] A. Samara, L. Galway, R. Bond, and H. Wang. Affective state detection via facial expression analysis within a human–computer interaction context. *Journal of Ambient Intelligence and Humanized Computing*, 10(6):2175–2184, 2019.

[215] F. Samaria and F. Fallside. *Automated face identification using hidden markov models*. Olivetti Research Limited, 1993.

[216] F. Samaria and F. Fallside. *Face identification and feature extraction using hidden markov models*. Citeseer, 1993.

[217] F. Samaria and S. Young. Hmm-based architecture for face identification. *Image and vision computing*, 12(8):537–543, 1994.

[218] F. S. Samaria and A. C. Harter. Parameterisation of a stochastic model for human face identification. In *Proceedings of 1994 IEEE Workshop on Applications of Computer Vision*, pages 138–142. IEEE, 1994.

[219] R. Sanford, S. Gorji, L. G. Hafemann, B. Pourbabaee, and M. Javan. Group activity detection from trajectory and video data in soccer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 898–899, 2020.

[220] B. Schuller, G. Rigoll, and M. Lang. Hidden markov model-based speech emotion recognition. In *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03).*, volume 2, pages II–1. IEEE, 2003.

[221] T. Sebastian, V. Jeyaseelan, L. Jeyaseelan, S. Anandan, S. George, and S. I. Bangdiwala. Decoding and modelling of time series count data using poisson hidden markov model and markov ordinal logistic regression models. *Statistical methods in medical research*, page 0962280218766964, 2018.

[222] M. T. Shahria, F. T. Progga, S. Ahmed, and A. Arisha. Application of neural networks for detection of sexual harassment in workspace. In *2021 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT)*, pages 1–4. IEEE, 2021.

[223] C. Shan, S. Gong, and P. W. McOwan. Facial expression recognition based on local binary patterns: A comprehensive study. *Image Vision Comput.*, 27:803–816, 2009.

[224] R. Shang, J. Song, L. Jiao, and Y. Li. Double feature selection algorithm based on low-rank sparse non-negative matrix factorization. *International Journal of Machine Learning and Cybernetics*, pages 1–18, 2020.

[225] C. E. Shannon. A mathematical theory of communication. *Bell system technical journal*, 27(3):379–423, 1948.

[226] R. S. Sidhu and M. Sharad. Smart surveillance system for detecting interpersonal crime. In *2016 International Conference on Communication and Signal Processing (ICCSP)*, pages 2003–2007. IEEE, 2016.

[227] R. Singh, A. K. S. Kushwaha, and R. Srivastava. Multi-view recognition system for human activity based on multiple features for video surveillance system. *Multimedia Tools and Applications*, 78(12):17165–17196, 2019.

[228] K. Soomro, A. R. Zamir, and M. Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

[229] J. Stiller and G. Radons. Online estimation of hidden markov models. *IEEE Signal Processing Letters*, 6(8):213–215, 1999.

[230] Y. Sun and A. N. Akansu. Facial expression recognition with regional hidden markov models. *Electronics Letters*, 50(9):671 –673, 2014.

[231] P.-N. Tan. *Introduction to data mining*. Pearson Education India, 2018.

[232] X. Tian, D. Tao, and Y. Rui. Sparse transfer learning for interactive video search reranking. *ACM Trans. Multimedia Comput. Commun. Appl.*, 8(3), Aug. 2012.

[233] G. G. Tiao and I. Cuttman. The inverted dirichlet distribution with applications. *Journal of the American Statistical Association*, 60(311):793–805, 1965.

[234] P. Tirdad, N. Bouguila, and D. Ziou. Variational learning of finite inverted dirichlet mixture models and applications. In *Artificial Intelligence Applications in Information and Communication Technologies*, pages 119–145. Springer, 2015.

[235] H. Uğuz. A two-stage feature selection method for text categorization by using information gain, principal component analysis and genetic algorithm. *Knowledge-Based Systems*, 24(7):1024–1032, 2011.

[236] M. A. ul Haq, M. Y. Hassan, H. Abdullah, H. A. Rahman, M. P. Abdullah, F. Hussin, and D. M. Said. A review on lighting control technologies in commercial buildings, their performance and affecting factors. *Renewable and Sustainable Energy Reviews*, 33:268–279, 2014.

[237] S. Uziel, T. Elste, W. Kattanek, D. Hollosi, S. Gerlach, and S. Goetze. Networked embedded acoustic processing system for smart building applications. In *2013 Conference on Design and Architectures for Signal and Image Processing*, pages 349–350. IEEE, 2013.

[238] M. Varma and R. Garg. Locally invariant fractal features for statistical texture classification. In *2007 IEEE 11th international conference on computer vision*, pages 1–8. IEEE, 2007.

[239] C. Wang, X. Zhao, Z. Wu, and Y. Liu. Motion pattern analysis in crowded scenes based on hybrid generative-discriminative feature maps. In *2013 IEEE International Conference on Image Processing*, pages 2837–2841. IEEE, 2013.

[240] J. Wang, X. Chen, and W. Gao. Online selecting discriminative tracking features using particle filter. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 1037–1042. IEEE, 2005.

[241] Y. Wang and A. Pal. Detecting emotions in social media: A constrained optimization approach. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, page 996–1002. AAAI Press, 2015.

[242] T.-T. Wong. Generalized dirichlet distribution in bayesian analysis. *Applied Mathematics and Computation*, 97(2-3):165–181, 1998.

[243] C. Wu, S. Wang, and Q. Ji. Multi-instance hidden markov model for facial expression recognition. *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, 1:1–6, 2015.

[244] T. Xiang and S. Gong. Activity based surveillance video content modelling. *Pattern Recognition*, 41(7):2309 – 2326, 2008.

[245] T. Xiang and S. Gong. Video behavior profiling for anomaly detection. *IEEE transactions on pattern analysis and machine intelligence*, 30(5):893–908, 2008.

[246] L. Xie, P. Xu, S.-F. Chang, A. Divakaran, and H. Sun. Structure analysis of soccer video with domain knowledge and hidden markov models. *Pattern Recognition Letters*, 25(7):767–775, 2004.

[247] W. Xu, Y. Pang, Y. Yang, and Y. Liu. Human activity recognition based on convolutional neural network. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 165–170. IEEE, 2018.

[248] Y. Xu, H. Ji, and C. Fermüller. Viewpoint invariant texture description using fractal analysis. *International Journal of Computer Vision*, 83(1):85–100, 2009.

[249] J. Yang, Y.-G. Jiang, A. G. Hauptmann, and C.-W. Ngo. Evaluating bag-of-visual-words representations in scene classification. In *Proceedings of the international workshop on Workshop on multimedia information retrieval*, pages 197–206. ACM, 2007.

[250] D. Yu and L. Deng. Hidden markov models and the variants. In *Automatic Speech Recognition*, pages 23–54. Springer, 2015.

[251] Y. Yu, H. Zhu, L. Wang, and W. Pedrycz. Dense crowd counting based on adaptive scene division. *International Journal of Machine Learning and Cybernetics*, 12(4):931–942, 2021.

[252] Y. Yun and I. Y.-H. Gu. Visual information-based activity recognition and fall detection for assisted living and ehealthcare. In *Ambient Assisted Living and Enhanced Living Environments*, pages 395–425. Elsevier, 2017.

[253] A. Zaharescu and R. Wildes. Anomalous behaviour detection using spatiotemporal oriented energies, subset inclusion histogram comparison and event-driven processing. In *European Conference on Computer Vision*, pages 563–576. Springer, 2010.

[254] N. Zamzami and N. Bouguila. Deriving probabilistic svm kernels from exponential family approximations to multivariate distributions for count data. In *Mixture Models and Applications*, pages 125–153. Springer, 2020.

[255] N. Zamzami and N. Bouguila. High-dimensional count data clustering based on an exponential approximation to the multinomial beta-liouville distribution. *Information Sciences*, 524:116–135, 2020.

[256] Z. Zeng, M. Pantic, G. I. Roisman, and T. S. Huang. A survey of affect recognition methods: Audio, visual, and spontaneous expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(1):39–58, 2009.

[257] B. Zhang. Regression clustering. In *Third IEEE International Conference on Data Mining*, pages 451–458. IEEE, 2003.

[258] J. Zhang, M. Marszałek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *International journal of computer vision*, 73(2):213–238, 2007.

[259] W. Zhang and Z. Yin. Eeg feature selection for emotion recognition based on cross-subject recursive feature elimination. In *2020 39th Chinese Control Conference (CCC)*, pages 6256–6261. IEEE, 2020.

[260] Y. Zheng, B. Jeon, L. Sun, J. Zhang, and H. Zhang. Student's t-hidden markov model for unsupervised learning using localized feature selection. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(10):2586–2598, Oct 2018.

[261] J. Zhou and X. Zhang. An ica mixture hidden markov model for video content analysis. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(11):1576–1586, Nov 2008.

[262] X. Zhou, X. Huang, and Y. Wang. Real-time facial expression recognition in the interactive game based on embedded hidden markov model. *Proceedings. International Conference on Computer Graphics, Imaging and Visualization, 2004. CGIV 2004.*, pages 144–148, 2004.

[263] H. Zhu, Z. He, and H. Leung. Simultaneous feature and model selection for continuous hidden markov models. *IEEE Signal Processing Letters*, 19(5):279–282, 2012.